

Documentation

OTRS 3.3 - Руководство Администратора

Build Date:

2013-04-30

OTRS 3.3 - Руководство Администратора

Авторские права © 2003-2013 OTRS AG

René Bakker, Stefan Bedorf, Michiel Beijen, Shawn Beasley, Hauke Böttcher, Jens Bothe, Udo Bretz, Martin Edenhofer, Carlos Javier García, Martin Gruner, Manuel Hecht, Christopher Kuhn, André Mindermann, Marc Nilius, Elva María Novoa, Henning Oswald, Martha Elia Pascual, Thomas Raith, Carlos Fernando Rodríguez, Stefan Rother, Burchard Steinbild, Michael Thießmeier, Daniel Zamorano.

Эта работа защищена авторским правом OTRS AG.

Это руководство может быть скопировано целиком или частично, при условии, что это уведомление об авторских правах будет включено в каждую копию.

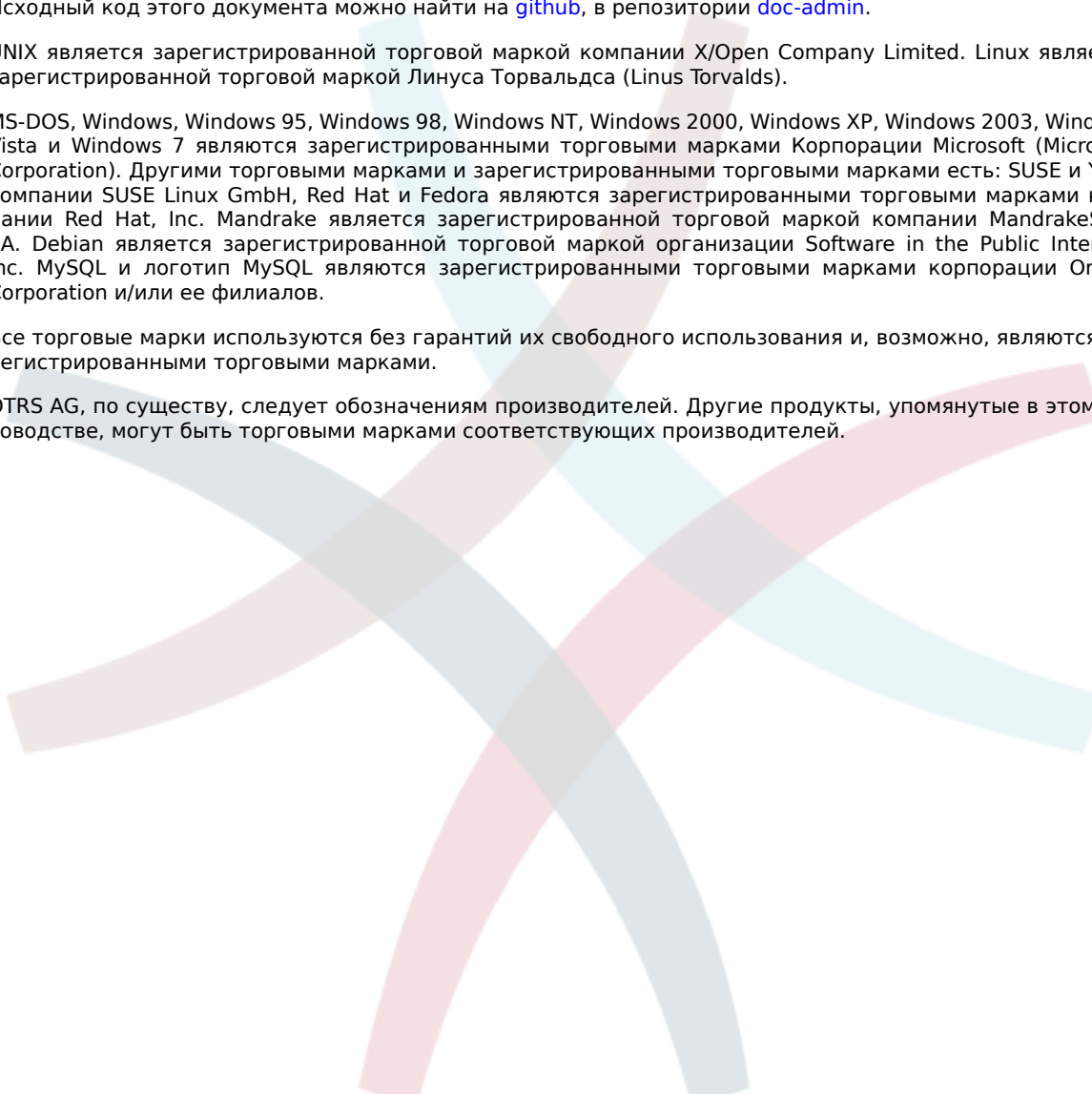
Исходный код этого документа можно найти на [github](#), в репозитории [doc-admin](#).

UNIX является зарегистрированной торговой маркой компании X/Open Company Limited. Linux является зарегистрированной торговой маркой Линуса Торвальдса (Linus Torvalds).

MS-DOS, Windows, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows 2003, Windows Vista и Windows 7 являются зарегистрированными торговыми марками Корпорации Microsoft (Microsoft Corporation). Другими торговыми марками и зарегистрированными торговыми марками есть: SUSE и YaST компании SUSE Linux GmbH, Red Hat и Fedora являются зарегистрированными торговыми марками компании Red Hat, Inc. Mandrake является зарегистрированной торговой маркой компании MandrakeSoft, SA. Debian является зарегистрированной торговой маркой организации Software in the Public Interest, Inc. MySQL и логотип MySQL являются зарегистрированными торговыми марками корпорации Oracle Corporation и/или ее филиалов.

Все торговые марки используются без гарантий их свободного использования и, возможно, являются зарегистрированными торговыми марками.

OTRS AG, по существу, следует обозначениям производителей. Другие продукты, упомянутые в этом руководстве, могут быть торговыми марками соответствующих производителей.



Содержание

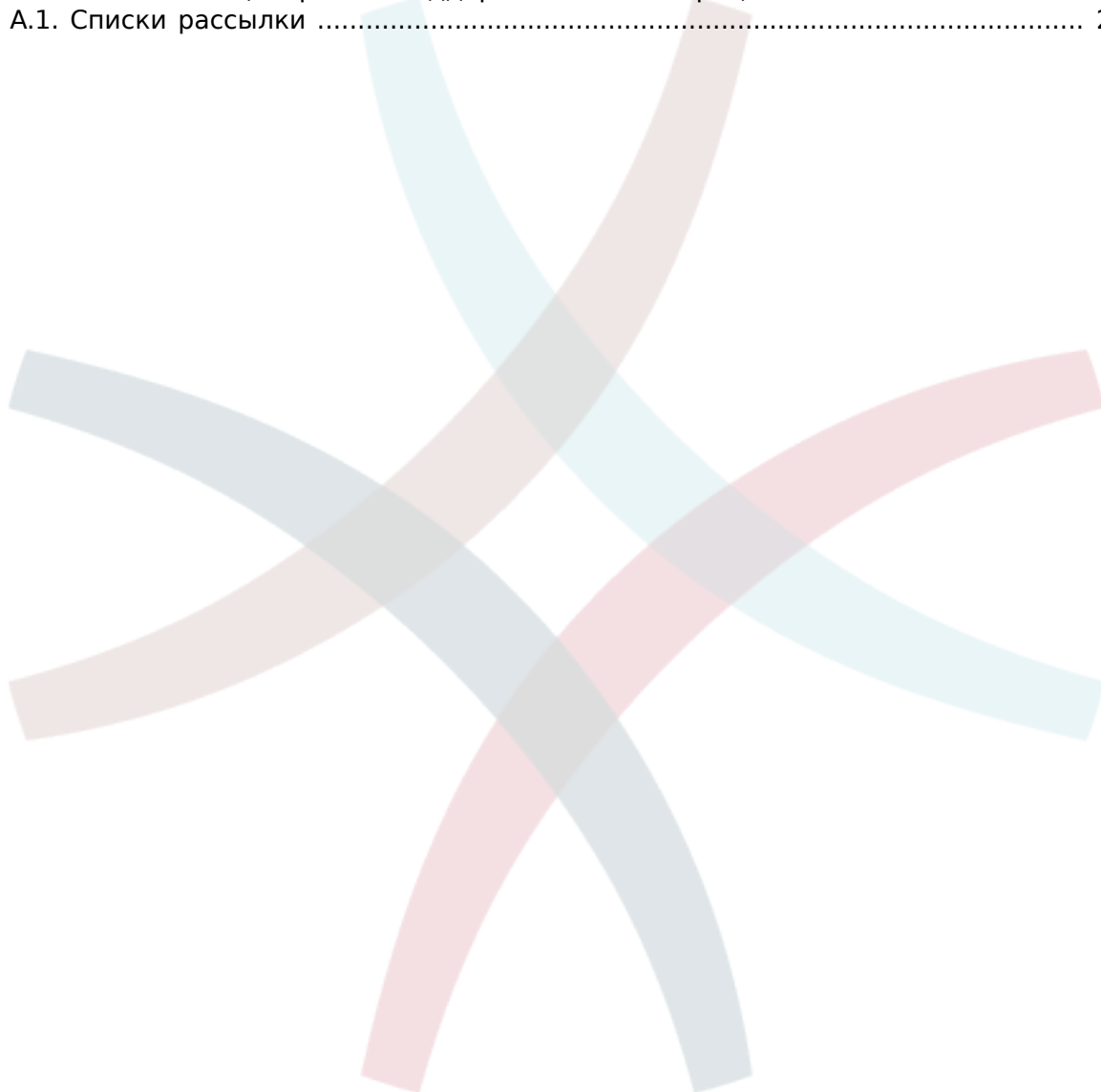
Предисловие	ix
1. Введение	1
1. Системы Обработки Заявок - Основы	1
1.1. Что такое система обработки заявок и зачем она нужна?	1
1.2. Что такое заявка?	2
2. OTRS Help Desk	2
2.1. Основы	3
2.2. Особенности и функциональные возможности	3
2.3. Требования к аппаратному и программному обеспечению	12
2.4. Сообщество	13
2.5. Профессиональная поддержка OTRS	14
2. Установка	15
1. Самый простой способ - установка из готовых бинарных пакетов	15
1.1. Установка из rpm-пакетов на сервер под управлением Suse Linux	15
1.2. Установка OTRS в операционной системе Red Hat Enterprise Linux или CentOS	18
1.3. Установка OTRS на системы Debian или Ubuntu	22
1.4. Установка OTRS на операционной системе Microsoft Windows	23
2. Установка из исходных кодов (Linux, Unix)	25
2.1. Подготовка к установке из исходных кодов	25
2.2. Установка Perl-модулей	26
2.3. Настройка веб-сервера Apache	27
2.4. Настройка базы данных	27
2.5. Настрой планировщика задач (cron jobs) для OTRS	29
2.6. Планировщик OTRS	32
3. Использование веб установщика	32
4. Обновление OTRS-фреймворка	36
5. Обновление с помощью Windows Installer	41
5.1. Выполнение промежуточных/patch level обновлений на Windows	41
5.2. Выполнение обновлений "малого" уровня на Windows	41
6. Дополнительные приложения	41
6.1. FAQ (Часто Задаваемые Вопросы)	41
3. Первые шаги	42
1. Веб-интерфейс для агентов	42
2. Пользовательский веб-интерфейс	42
3. Общедоступный веб-интерфейс	43
4. Первое применение	43
5. Веб-интерфейс - обзор	44
6. Панель Управления/Дайджест	46
7. Что такое очередь?	50
8. Что такое просмотр очередей?	51
9. Пользовательские настройки	52
4. Администрирование	54
1. Панель управления OTRS	54
1.1. Основы	54
1.2. Агенты, Группы и Роли	54
1.3. Клиенты и Группы Клиентов	63
1.4. Очереди	66
1.5. Приветствия, подписи, вложения и шаблоны	68
1.6. Авто ответы	77
1.7. Адреса электронной почты	79
1.8. Уведомления	80
1.9. SMIME	82

1.10. PGP	83
1.11. Состояния	83
1.12. SysConfig	84
1.13. Использование почтовый ящиков	85
1.14. Фильтрация входящих почтовых сообщений	86
1.15. Выполнение автоматизированных заданий с помощью GenericAgent	89
1.16. Электронная почта администратора	91
1.17. Управление сеансами	91
1.18. Системный журнал	92
1.19. SQL запросы	93
1.20. Менеджер пакетов	94
1.21. Веб-сервисы	94
1.22. Динамические Поля	95
2. Настройка Системы	96
2.1. Конфигурационные файлы OTRS	96
2.2. Настройка системы через графический веб-интерфейс	97
3. Резервное копирование системы	98
3.1. Резервное копирование	98
3.2. Восстановление	99
4. Настройки электронной почты	99
4.1. Отправка / получение электронной почты	99
4.2. Безопасность электронной почты с PGP	105
4.3. Безопасность электронной почты с S/MIME	107
5. Использование внешних хранилищ данных	110
5.1. Данные клиентов	110
5.2. Пользовательский бэк-энд	111
5.3. Хранилища (бэк-енды) для аутентификации Агентов и Клиен- тов	118
5.4. Настройка самостоятельной регистрации клиента	122
6. Настройки Заявки	125
6.1. Состояния заявок	125
6.2. Приоритеты заявок	128
6.3. Ответственность за Заявку & Наблюдение за Заявкой	129
7. Функции связанные с временем	131
7.1. Настройка периода рабочего времени, праздников и часовых поя- сов	131
7.2. Автоматические Разблокировки	132
8. Настройка вывода в формате PDF	133
9. Stats-модуль	133
9.1. Обработка модуля агентом	134
9.2. Администрирование модуля статистики OTRS администрато- ром.	143
9.3. Администрирование модуля статистики системным администрато- ром	143
10. Динамические Поля	146
10.1. Введение	146
10.2. Настройка Системы	146
11. Generic Interface	165
11.1. Слои интерфейса Generic Interface	165
11.2. Поток связи Generic Interface	167
11.3. Веб-сервисы	170
11.4. Графический интерфейс веб-сервиса	170
11.5. Интерфейс командной строки Веб-Сервиса	187
11.6. Конфигурация Веб-сервиса	189
11.7. Коннекторы (Connectors)	196
12. Планировщик OTRS	206
12.1. Графический интерфейс планировщика	207

12.2. Интерфейс командной строки Планировщика	208
5. Настройка	213
1. Списки Контроля Доступа (ACLs)	213
1.1. Введение	213
1.2. Примеры	213
1.3. Ссылка	216
2. Управление Процессами	219
2.1. Введение	219
2.2. Пример процесса	219
2.3. Выполнение примера	219
2.4. Process configuration reference	243
3. Создание своих собственных тем (шаблонов)	262
4. Локализация интерфейса OTRS	263
6. Настройка Производительности	264
1. OTRS	264
1.1. TicketIndexModule	264
1.2. TicketStorageModule	264
1.3. Архивирование Заявок	265
2. База данных	266
2.1. MySQL	266
2.2. PostgreSQL	266
3. Веб-сервер	266
3.1. Предустановленное соединение с базой данных	266
3.2. Предварительно загруженные модули - startup.pl	267
3.3. Перезагрузка Perl-модулей во время обновления с диска	267
3.4. Выбор Правильной Стратегии	267
3.5. mod_gzip/mod_deflate	267
A. Дополнительные ресурсы	268
1. OTRS Group вебсайт	268
2. Списки рассылки	268
3. Трекинг ошибок	269
4. Коммерческая поддержка	269
B. Configuration Options Reference	270
1. DynamicFields	270
2. Framework	275
3. GenericInterface	405
4. ProcessManagement	417
5. Scheduler	425
6. Ticket	428
C. Благодарности	659
D. GNU Free Documentation License	661
0. PREAMBLE	661
1. APPLICABILITY AND DEFINITIONS	661
2. VERBATIM COPYING	662
3. COPYING IN QUANTITY	662
4. MODIFICATIONS	663
5. COMBINING DOCUMENTS	664
6. COLLECTIONS OF DOCUMENTS	664
7. AGGREGATION WITH INDEPENDENT WORKS	665
8. TRANSLATION	665
9. TERMINATION	665
10. FUTURE REVISIONS OF THIS LICENSE	665
How to use this License for your documents	666

Список таблиц

2.1. Описание некоторых сценариев планировщика задач cron job.	29
4.1. Предопределенные группы, доступны для новой (только что) установленной системы OTRS	56
4.2. Права, связанные с OTRS-группами	58
4.3. Дополнительные группы прав	59
4.4. События для Автоматических ответов	78
4.5. Функции различных X-OTRS-заголовков	86
4.6. В систему будут добавлены следующие поля:	148
4.7. Список init сценариев и поддерживаемых Операционных систем	209
A.1. Списки рассылки	268



Список примеров

4.1. Сортирование спама в определенную очередь	89
4.2. .fetchmailrc	102
4.3. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::Match	102
4.4. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::CMD	103
4.5. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition	103
4.6. Настройка клиентского хранилища базы данных (DB)	111
4.7. Хранение Заявок Компании в базе данных DB	114
4.8. Настройка LDAP в качестве клиентского бэк-енда	114
4.9. Использование заявок Компании с LDAP-бэкэндом	116
4.10. Использование больше чем одного пользовательского хранилища дан- ных с OTRS	116
4.11. Проверка подлинности агентов путем использования Базы Данных (DB) в качестве хранилища информации.	119
4.12. Проверка подлинности агентов при использовании LDAP в качестве хра- нилища данных	119
4.13. Аутентификация Агентов с помощью HTTPBasic	120
4.14. Аутентификация (проверка подлинности) агентов с использованием Radius-сервера в качестве хранилища информации	121
4.15. Аутентификация Клиента в Базе Данных	121
4.16. Аутентификация пользователей с помощью LDAP-бэкэнда	121
4.17. Аутентификация клиентов с помощью HTTPBasic	122
4.18. Аутентификация клиентов с использованием Radius	122
4.19. Определение значения серии - один из элементов	141
4.20. Определение значения серии - один из элементов	141
4.21. Активировать Field1 на странице Новая Заявка на основе телефонного звонка.	158
4.22. Сделать Field1 на странице просмотра Заявки созданной на базе теле- фонного звонка New Phone Ticket Screen as mandatory.	158
4.23. Активировать несколько полей на странице просмотра Новых Телефон- ных Заявко.	159
4.24. Отключить некоторые поля на странице Новой Заявки, созданной на ос- нове телефонного звонка.	160
4.25. Активировать поле Field1 на странице просмотра подробной информа- ции о Заявке.	161
4.26. Активировать поле Field1 на странице просмотра обобщенной информа- ции о Заявке.	162
4.27. Активировать Field1 в событии TicketCreate.	163
4.28. Активировать поле Field1 на странице Пользовательские Настройки.	164
4.29. Пример для запуска OTRS-Планировщика в linux	209
4.30. Пример для Запуска OTRS-Планировщика	210
4.31. Пример принудительной остановки Планировщика OTRS	210
4.32. Пример регистрации Планировщика OTRS как сервиса Widows	211
4.33. Пример запуска Планировщика OTRS в Windows	211
4.34. Пример принудительной остановки Планировщика OTRS в Windows	212
5.1. Списки прав доступа (ACL) позволяют перемещать в очереди только за- явки с приоритетом 5.	213
5.2. ACL позволяют перемещать в другие очереди только заявки с приорите- том 5, уже имеющиеся в базе данных.	214
5.3. Списки прав доступа (ACL) делают невозможным закрытие заявок в оче- реди raw и скрывают кнопку "закрыть".	214
5.4. ACL удаляет навсегда возможность установить состояние "closed successful".	215

5.5. ACL только отображает Hardware-сервисы для заявок, которые были созданы в очередях, начинающихся с "HW".	215
5.6. Ссылка отображает все возможные важные настройки ACL	216



Предисловие

Эта книга предназначена для использования OTRS-администраторами. Он также послужит хорошим справочником для новичков OTRS.

В следующих главах описывается установка, настройка и администрирование программы OTRS. В первой трети текста описываются ключевые функциональные возможности программного обеспечения, в то время как остальной текст является ссылкой на полный набор настраиваемых параметров.

Эта книга и дальше находится в стадии разработки и совершенствуется с выходом новых версий продукта. Чтобы сделать это справочное руководство более качественным, полным и точным нам нужна обратная связь с вами. Пожалуйста, напишите нам, если заметите отсутствие некоторых разделов в этой книге, если некоторые вещи объясняются недостаточно хорошо или, если просто видите орфографические, грамматические ошибки или опечатки. Мы будем признательны за любой вид обратной связи, которую можно осуществить посредством нашей системы отслеживания ошибок, которая находится по адресу <http://bugs.otrs.org>. Заранее благодарим вас за помощь!



Глава 1. Введение

1. Системы Обработки Заявок - Основы

В этой главе дается краткий обзор идеи заявок в целом, и систем обработки заявок, в частности. Краткий пример иллюстрирует преимущества использования системы обработки заявок.

1.1. Что такое система обработки заявок и зачем она нужна?

Следующий пример показывает, что собой являет система обработки заявок и как можно сэкономить много времени и денег, используя подобную систему в своей компании.

Давайте представим, что Максим занимается производством видеомагнитофонов. Максим получает много писем от клиентов, которые нуждаются в помощи по устройству. Спустя несколько дней, у него нету свободного времени чтобы оперативно ответить на все письма клиентов или хотя бы просмотреть их. Некоторые клиенты не хотят долго ждать и пишут второе письмо с тем же вопросом. Все письма, содержащие просьбы о поддержке хранятся в одном почтовом ящике. Запросы не сортируются и Максим отвечает на письма пользуясь обычной программой для работы с электронной почтой.

Поскольку Максим не успевает быстро дать ответ на все письма, его инженеры: Андрей и Александр берутся помочь Максиму. Для работы с почтой Андрей и Александр используют один и тот же почтовый ящик, который хранит все письма клиентов. Они ничего не подозревают о том, что Максим получил два одинаковых письма с вопросом о помощи от отчаявшегося клиента. Временами они оба отвечают на один и тот же запрос, и клиент получает два различных ответа. Более того, Максим не имеет представления о содержимом этих ответов. Также он не знает подробностей большинства проблем его клиентов и о том как они решаются, какие проблемы возникают чаще всего, а также, сколько времени и денег он должен потратить на поддержку клиентов.

На собрании коллега рассказывает Максиму о системах обработки заявок и как они могут решить его проблемы с поддержкой клиентов. После поиска информации в Интернете, Максим принимает решение установить OTRS - Открытую Систему Обработки Заявок, далее просто OTRS, на компьютере, который будет доступен из веб как для клиентов компании так и для ее сотрудников. С этого момента, запросы клиентов больше не приходят на частный почтовый ящик Максима, теперь они направляются на почтовый ящик, который используется для OTRS. OTRS подключена к этому почтовому ящику и автоматически сохраняет все запросы в своей базе данных. Система генерирует авто-ответ для каждого нового запроса и отправляет его клиенту, так что теперь клиент может быть уверен в том, что его запрос поступил в обработку и ответ будет дан в ближайшее время. Для каждого запроса OTRS создает прямую ссылку - номер заявки. Теперь клиенты довольны, они получают быстрый ответ на свою заявку и им уже нет никакого смысла писать повторно письмо с тем же вопросом. Максим, Андрей и Александр могут спокойно отвечать на поступившие запросы используя обычный веб-браузер для работы с OTRS. Также, следует отметить тот факт, что ни одно сообщение клиента не будет отредактировано дважды, поскольку система автоматически блокирует заявку, для которой создается ответ.

Давайте представим что г-н Гришко делает запрос в компанию Максима и его сообщение обрабатывается системой OTRS. Александр дает краткий ответ на его вопрос. Но у г-на Гришко есть дополнительный вопрос и на следующий день он отвечает на письмо Александра. Поскольку у Александра и своих дел хватает, на письмо г-на Гришко отвечает Максим. Способность OTRS хранить историю позволяет Максиму просмотреть всю цепочку переписки по этому запросу и написать более подробный ответ г-ну Гришко. Г-н Гришко, в свою очередь, даже не догадывается, что в процесс общения были вовлечены два разных человека, он доволен советами относительно решения своей проблемы, которые пришли в последнем сообщении от Максима.

Конечно, это всего лишь краткий обзор функциональных возможностей и особенностей систем обработки заявок. Но в случае, если в вашу компанию приходит множество запросов от клиентов через письма электронной почты и телефонные звонки и они должны обрабатываться в разное время, - система обработки заявок поможет существенно сэкономить время и деньги, позволит структурировать всю систему поддержки клиентов, сделать более прозрачным процесс обмена информацией между клиентами и компаниями, что неотменно приведет к повышению эффективности как для клиентов так и для обслуживающего персонала и позитивно скажется на финансовом положении компании.

1.2. Что такое заявка?

Заявка очень похожа на медицинскую карточку пациента больницы. Медицинская карточка создается, когда пациент посещает больницу впервые, все важные данные о пациенте, такие как: личные данные, информация о состоянии его здоровья, медицинском осмотре записываются в медицинскую карточку. С каждым новым визитом пациента в больницу, каждый из лечащих врачей добавляет в медицинскую карточку информацию о состоянии пациента, истории болезни и, используемые для лечения лекарственные препараты. Таким образом другие лечащие врачи и медсестры могут видеть, подробную картину состояния пациента. Когда пациент выздоровел и выписался из больницы, медицинская карточка закрывается и вся информация передается в архив.

Системы обработки заявок, такие как OTRS, обрабатывают заявки как обыкновенные email. Когда клиент посылает запрос, система создает новую заявку, что сравнимо с открытием новой медицинской карточки пациента больницы. Ответ в этой новой заявке можно сравнить с записью лечащего врача в медицинской карточке, свидетельствующей о изменении состояния здоровья пациента. Заявка считается закрытой, когда клиенту отправлен ответ или если заявка закрыта системой автоматически. Если клиент отправляет ответ на уже закрытую заявку, она будет заново открыта системой и в нее будет добавлена новая информация. Чтобы сохранить целостность всех данных, хранящихся в системе, каждая заявка хранится и архивируется вместе со всей информацией. Поскольку заявки обрабатываются как обыкновенные сообщения электронной почты - вместе с ними также будут сохранены все вложения и контекстные аннотации. Конечно же, такая полезная информация как примечания, даты, данные о нанятых работниках, рабочее время необходимое для работы с заявкой и т.д также сохраняется системой. Все заявки могут быть отсортированы в соответствующем порядке, также можно с легкостью производить поиск всей необходимой информации.

2. OTRS Help Desk

В этой главе описываются характерные особенности и функциональные возможности Открытой Системы Обработки Заявок (OTRS). Здесь вы найдете полезную информацию о том, какое оборудование и программное обеспечение необходимо для функционирования OTRS. Также, в этой главе рассказывается о том, как получить ком-

мерческую поддержку системы OTRS, на случай, если вы в ней нуждаетесь, а также, каким образом можно связаться с сообществом.

2.1. Основы

OTRS Help Desk (OTRS) веб приложение, которое устанавливается на веб-сервер и используется в веб-браузере.

OTRS разделен на несколько компонентов. Главным компонентом всей системы является OTRS-фреймворк, в котором содержатся все основные функции приложения и система обработки заявок. Веб-интерфейс OTRS позволяет устанавливать дополнительные приложения, такие как, например, OTRS::ITSM, а также инструментальные средства для мониторинга информации о состоянии системы, базу знаний (FAQ) и многие другие.

2.2. Особенности и функциональные возможности

OTRS имеет большой набор функций. Следующий список дает обзор основных особенностей, включенных в OTRS фреймворк.

2.2.1. Пользовательский интерфейс

- OTRS поставляется с современным веб-интерфейсом разделенным для агентов и пользователей
- Он прекрасно работает в любом современном браузере, включая мобильные платформы и retina-дисплеи.
- Используя темы и скины можно настроить веб-интерфейс по своему вкусу.
- Многофункциональная, настраиваемая панель управления для агентов с персональным, отдельным блоком для просмотра заявок и графической статистикой.
- Расширяемый генератор отчетов предоставляет различные статистические данные и настраиваемые параметры планирования отчетов.
- С помощью ProcessManagement можно создавать собственные, основанные на заявках экраны и процессы (поток работ по заявке - workflows).
- OTRS имеет встроенную систему прав, функциональность которой можно с легкостью расширить с помощью списков контроля доступа (ACL).
- Поддержка более 30 языков и разных часовых поясов.

2.2.2. Почтовый интерфейс

- Поддержка MIME с вложениями.
- Автоматическое преобразование HTML в простой текст (повышает безопасность "чувствительного" содержания и значительно ускоряет поиск).
- Входящая почта может быть отфильтрована с использованием сложных правил, например для извлечения спам - сообщений или для распределения заявок по очередям.
- Поддержка PGP и S/MIME стандартов для управления ключами/сертификатами и обработки электронной почты.

- Автоматические ответы, настраиваемые для каждой очереди.
- Уведомления агентов по электронной почте о новых, последующих или разблокированных заявках.
- Имеется возможность задать свой собственный идентификатор заявки для распознавания связанных с ней объектов, например, Звонок№, Заявка№ или Запрос№. В системе имеется несколько способов генерации номеров (основанный на дате, на базе случайных чисел и т.д.), также, вы можете включить свой собственный. Связанные объекты также могут распознаваться с использованием In-Reference-To заголовков или внешних номеров заявок.

2.2.3. Заявки

- OTRS использует понятие заявки для сбора информации из всех внешних и внутренних источников коммуникации. Эти заявки помещаются в очереди.
- Доступно много различных способов просмотра заявок в системе (на основе Очередей, Статуса, Эскалации...) используя различные уровни детализации (малый/средний/предварительный просмотр).
- Все изменения заявки регистрируются в Истории заявки.
- Изменения в заявках могут быть выполнены многими разными способами, такими как ответ на заявку, перенаправление, пересылка, перемещение в другую очередь, изменение ее атрибутов (состояние, приоритет и пр...), блокировка и ввод затраченного на нее времени. Можно изменять множество заявок одновременно (массовое действие).
- Перевод в ожидание и установка параметров эскалации / Управление SLA - соглашениями об уровне сервиса предоставляют возможность планирования обработки заявок во времени и устанавливать ограничения для них.
- Заявки могут быть привязаны к другими заявкам или объектам, например, к записям из FAQ.
- "GenericAgent" позволяет совершать автоматические и синхронные действия над заявками.
- OTRS поставляется с мощной поисковой системой, которая позволяет выполнять сложный и полнотекстовый поиск по заявкам.

2.2.4. Система

- OTRS работает во многих операционных системах (Linux, Solaris, AIX, FreeBSD, OpenBSD, Mac OS 10.x, Microsoft Windows) и поддерживает работу с разнообразными СУБД для основного сервера баз данных OTRS (MySQL, PostgreSQL, Oracle, MSSQL).
- Ядро системы может быть расширено с помощью установки дополнительных пакетов OTRS. Существует множество свободно распространяемых пакетов (таких как FAQ, OTRS::ITSM и другие), а также FeatureAddon пакеты для клиентов, заключивших сервисные контракты с OTRS group.
- Для хранения данных о клиентах предусмотрена возможность интеграции внешних источников данных, используя AD, eDirectory или OpenLDAP. Клиенты могут аутентифицироваться с помощью внутренней БД, LDAP, HTTPAuth или Radius.
- Используя GenericInterface, можно легко подключить OTRS к другим веб-сервисам. Простые веб-сервисы могут быть интегрированы без использования про-

граммирования, сложных сценариев с различными расширениями. OTRS Ticket Connector позволяет создание, обновление и поиск заявок через веб-сервисы через стороннее ПО.

Рассмотрим изменения в последних версиях OTRS.

2.2.5. Новые возможности OTRS 3.3

2.2.5.1. Продуктивность

- Теперь можно использовать фильтры по допустимым атрибутам заявки в виджетах Дайджеста и стандартных обзорах заявок. Можно настраивать перечень и порядок отображения колонок в обзорах.
- В обзорах заявок в среднем формате и формате предпросмотра можно использовать сортировку.
- В приборной панели добавлен виджет календарь, который может отображать заявки как события.
- Добавлен новый виджет в Дайджест для отображения в табличной форме количества заявок в разрезе состояний и очередей.
- Агенты могут отмечать важные статьи.
- Новый способ выбора значений (для очередей, сервисов и пр.) с помощью виджета, с отображением списков в виде дерева, делает работу более быстрой и удобной.
- Добавлена поддержка поиска по относительным значениям дат (например, более, чем 1 месяц назад) в динамических полях типа Date и Date/Time.
- Теперь есть возможность задания шаблонов (ранее, "стандартных ответов") также при создании новых заявок и их пересылке.
- Список доступных процессов теперь можно фильтровать с помощью ACL.
- Добавлен возможность инициировать процесс/процессную заявку из интерфейса клиента.
- Во многих местах теперь, значение текстового поля более не обрезается на определенном количестве символов ("Queue1..."). Вместо этого экран расширяется и можно воспользоваться прокруткой по горизонтали. Это позволяет увидеть больше информации сразу.
- OTRS теперь поддерживает Retina-дисплеи. Изображения были адаптированы под высокое разрешение и большинство иконок были заменены символами из веб-шрифта FontAwesome
- Добавлена новая возможность "управляемый дайджест". Она дает возможность отображать статистические диаграммы в Дайджесте. Обращаем внимание, что IE8 не поддерживает эту возможность.

2.2.5.2. Работа с внешними системами

- OTRS ныне может использовать несколько баз данных для компаний клиентов, благодаря Cyrille @ belnet-ict.
- OTRS теперь может автоматически сохранять данные клиента в динамических полях заявки для постоянного хранения в заявке. Это может быть полезно в отчетах, если, например, в отчетном периоде, клиент сменил компанию.

- OTRS теперь способен корректно присоединять входящие почтовые сообщения к существующим заявкам, основанным на номерах из внешних систем.
- OTRS способен теперь забирать почту также и через POP3/TLS соединения.

2.2.5.3. Установка и Администрирование

- Теперь Веб-Установщик может устанавливать OTRS с такими базами данных как PostgreSQL, Oracle и SQL Server, а также MySQL.
- Теперь OTRS имеет полную поддержку MySQL 5.6.
- Задания Планировщика теперь могут выполняться на основе заданных событиях по заявке.
- Новый графический редактор ACL значительно упрощает редактирование списков ACL.
- Фильтры Postmaster теперь могут использовать условия с отрицанием, спасибо Renée Bäcker.
- Фильтры Postmaster теперь могут определять относительную дату ожидания и Владельца / Ответственного для новых заявок, основанных на данных из входящих email.
- Пароли для Клиента и Агента теперь могут быть зашифрованы с помощью стойкого алгоритма bcrypt, который лучше SHA.
- Сейчас многие значки теперь используют иконочный шрифт, который позволяет проще создавать различные обложки с различными основными цветами. Это также улучшает общую производительность за счет малого количества загружаемых файлов (картинок).

2.2.6. Новые возможности OTRS 3.2

2.2.6.1. Больше ориентирован на клиента

- Новый Центр Информации о Клиентах предоставляет превосходный "дайджестоподобный" обзор клиентов (компаний). Вы можете видеть
 - Эскалированные, ожидающие напоминания, новые и открытые заявки компании клиента.
 - Клиенты (контакты) принадлежащие этой компании, с их индивидуальными счетчиками заявок и их обозначениями для создания новых заявок для них.
 - Общий обзор статусов заявок компании клиента.
- Новая возможность "Переключиться на клиента" дает возможность агенту, с определенными правами, зайти на портал клиента с его/клиента правами.

2.2.6.2. Более настраиваемый

2.2.6.2.1. Управление Процессами

- Новый функционал управление процессами дает возможность использования процессов в OTRS.

2.2.6.2.2. Улучшен пользовательский веб-интерфейс

- Веб - интерфейс клиента теперь полностью поддерживает использование AJAX и ACL.

- Он теперь требует наличия поддержки JavaScript и несовместим с Internet Explorer 6 и более ранними версиями.
- Для интерфейса клиента вы теперь можете задать тип заявки по умолчанию для новых заявок. Дополнительно, вы также имеете возможность отключить ввод типа заявки и использовать значение его по умолчанию для всех заявок создаваемых через интерфейс клиента.

2.2.6.2.3. Улучшен веб-интерфейс Агента

- Агенты теперь могут осуществлять поиск заявок на основании времени эскалации.
- Новая возможность показывать динамические поля по умолчанию при поиске заявок.
- Оптимизация использования площади экрана при работе с заявками во всплывающих окнах во избежание использования прокрутки. Для каждого экрана заявки, размер окна текстового редактора теперь может быть настроен отдельно.
- Теперь появилась возможность перемещать заявки в другую очередь в окнах действий с заявкой (TicketNote, TicketClose и т.д.) после включения этой опции в SysConfig. По умолчанию, она выключена.
- Если результатом поиска является единственная заявка, теперь вы сразу попадете на экран просмотра заявки, минуя список найденных.
- Новая возможность скрывать заданные типы сообщений/заметок на экранах просмотра заявок, что может быть полезным для более удобного обзора экрана.
- Новый виджет для Дайджеста отображающий список агентов "вне офиса".
- Новый CKEditor 4 делает работу с форматированным текстом (письма в HTML) проще и стабильнее.

2.2.6.2.4. Улучшения в Администрировании

- Уведомления о событиях теперь можно отправлять только для определенных типов отправителей сообщений/заметок.
- Модуль отчетов OTRS теперь понимает термин "Неделя" в дополнение к дням, месяцам и годам. Это дает возможность создавать отчеты, например, по заявкам "созданным за последнюю неделю" или "заявки созданные в очередях за неделю".
- Теперь можно помещать кастомизированные DTL (шаблоны) файлы в Custom/Kernel/Output/HTML, и они будут перекрывать системные DTL файлы как это уже работает для Perl файлов.
- В AdminSMIME теперь можно отображать содержимое сертификата в человеко-читаемом виде.
- SysConfig теперь поддерживает настройку типов Date и DateTime.

2.2.6.3. Улучшена масштабируемость

2.2.6.3.1. Улучшенная функция Архивирование Заявок

- При архивации заявок, информация, прочитанная агентом и сообщения могут быть удалены, также как и информация о подписке агентов на заявки. Эта опция вклю-

чена по умолчанию и позволяет уменьшить количество информации в базе данных в больших системах с множеством заявок и агентов.

- Добавлен новый скрипт для удаления этих данных из существующих архивных заявок.
- Архивные заявки теперь всегда отображаются как "прочтенные" агентом.

2.2.6.3.2. Улучшена производительность

- Управление сеансами стало в 10 раз быстрее, особенно это касается работы с активными пользователями.
- Теперь можно задать лимит количества одновременно работающих агентов и/или клиентов во избежание перегрузки серверов.
- Значительное снижение количества выполненных инструкций базы данных в объемах заявок и шаблонов поиска заявок в интерфейсах агента и клиента.
 - Это позволит снизить загрузку сервера баз данных, особенно в больших системах. В ряде случаев OTRS стал заметно более "отзывчивым" (если система тормозила из-за загрузки сервера БД или задержек).
- Улучшена производительность для LDAP синхронизации пользователей.
- Увеличена скорость кэширования для многих кэш-файлов.

2.2.6.4. Улучшена совместимость

2.2.6.4.1. FAQ коннектор для GenericInterface

- Появилась возможность доступа к данным модуля FAQ (база знаний OTRS) через веб сервисы (GenericInterface). Это может быть полезно, например, для включения статей FAQ в ваши корпоративные веб сайты.

2.2.7. Новые возможности OTRS 3.1

2.2.7.1. GENERIC INTERFACE - набор средств для работы с веб сервисами.

- GI гибкий фреймворк, который позволяет веб-сервисам OTRS взаимодействовать с приложениями сторонних производителей.
- OTRS может работать как поставщик (сервер, который получает запросы от удаленной стороны) или же как клиент (отправлять запросы).
- Просты подключения к веб-сервисам могут быть созданы путем конфигурации Generic Interface, не требуя дополнительного программирования.
- Сложные сценарии могут быть реализованы путем подключения пользовательских расширений OTRS, которые добавляют perl-код на различных архитектурных уровнях.
- *Connectors/Коннекторы* представляют собой часть OTRS для веб-сервисов Generic Interface. Например, коннектор заявок представляет функцию создания/обновления заявок, таким образом они могут быть использованы в веб-сервисе в зависимости от используемого транспортного протокола.
- Процесс демона планировщика поддерживает асинхронное управление событиями. Это полезно для асинхронного запуска запроса веб-сервиса от OTRS к дру-

гой системе, после того как запрос агента был выполнен (например, когда заявка была создана). Иначе, он может блокировать ответ, что приведет к увеличению времени ожидания ответа агентом.

С помощью Generic Interface можно легко настроить новые веб-сервисы с использованием существующих в системе OTRS модулей, без дополнительного программирования. Они могут быть настроены для создания новых веб-сервисов. При создании нового коннектора веб-сервиса, администратор должен добавить:

- Новый веб-сервис в GUI администратора
- Основные мета-данные (транспортные (SOAP), URL и т.д.) и
- Существующие операции (часть коннектора) и задать для каждой операции правила, как должны быть сопоставлены данные (входящие и исходящие)

Отладчик Generic Interface поможет OTRS-администратору проверить как поступают запросы и как они обрабатываются в рамках различных слоев.

2.2.7.1.1. Текущие возможности

- Сетевые транспортные протоколы: SOAP/HTTP. Поддержка протоколов REST and JSON будет добавлена в будущем, в зависимости от спроса потребителей.
- Настраиваемое отображение данных в GUI для преобразования key/value-значений входящих и исходящих данных.
- Графический отладчик для проверки конфигурации и потока информации от настроенных веб-сервисов.
- Ticket-коннектор позволяет использовать OTRS для обработки заявок как веб-сервис.

2.2.7.1.2. Будущие Особенности

- Дополнительные сетевые транспортные протоколы (REST, JSON).
- GI заменит iPhoneHandle как backend для мобильных приложений.
- Были добавлены дополнительные коннекторы для обеспечения работы многих функций OTRS с веб сервисами (например, для создания, обновления или удаления агентов, клиентов, сервисов или учетных элементов основных средств).

2.2.7.2. ДИНАМИЧЕСКИЕ ПОЛЯ

Динамические поля (DynamicFields) заменяют существующие поля FreeText и FreeTime с динамической структурой для заявок и статей. Они также позволяют создавать пользовательские формы в OTRS.

- Неограниченное количество полей можно настроить с помощью собственного графического пользовательского интерфейса администрирования.
- Поля могут иметь различные типы, которые могут быть использованы как для заявок так и для статей. По умолчанию доступны:
 - Text
 - Multiline text
 - Checkbox
 - Dropdown

- Multi-select
- Date
- Дата и время
- Новые пользовательские типы полей (например выпадающий список, который получает данные из внешнего источника) могут быть добавлены с небольшими усилиями, поскольку поля создаются по модульному принципу.
- В будущем, динамические поля смогут использоваться для объектов, отличных от заявок или в пользовательских модулях. Например, пользовательский модуль добавляющий объекты для управления "заказами/нарядами", сможет использовать динамические поля для присоединения данных к этим заказам.
- Скрипт обновления базы данных превратит поля FreeText и связанные с ними настройки в новую структуру.

2.2.7.3. УСОВЕРШЕНСТВОВАНИЯ ПО УПРАВЛЕНИЮ ЗАЯВКАМИ

2.2.7.3.1. Улучшено создание Заявок

- Когда создается новая заявка на основе email или телефонного звонка, множество email-адресов может быть задано как 'To:', 'CC:' or 'BCC:' .

2.2.7.3.2. Поддержка входящих телефонных звонков

- Теперь для созданных заявок могут быть зарегистрированы входящие телефонные звонки (до этого, можно было зарегистрировать только исходящие звонки).

2.2.7.3.3. Улучшены возможности предпросмотра заявок.

- Теперь в SysConfig можно исключать статьи отдельных типов (напримр, статьи от внутренних агентов), это будет предотвращать их появление в режиме предварительного просмотра.
- Определенный тип статьи может быть настроен таким образом, что будет отображать статьи этого типа полностью, если имеются соответствующие права для их просмотра.

2.2.7.3.4. Улучшена операция перемещения Заявок

- Страница, которая отображается после перемещения заявки теперь может быть настроена пользователем. Варианты просмотра: просмотр расширенной информации (LastScreenView) или же список заявок (LastScreenOverview).

2.2.7.3.5. Улучшена операция одновременной работы с несколькими элементами

- В новой версии функции массового действия, исходящие письма теперь могут быть посланы от нескольких заявок одновременно. Т.к. заявки могут относиться к разным очередям и эти очереди могут иметь разные шаблоны, приветствия и подписи, они не используются в письмах, формируемых при массовом действии.
- Дополнительные массовые действия позволяют настроить тип заявки для отдельных заявок.

2.2.7.3.6. Configurable Reject Sender Email Address

- Эта возможность позволяет подставить другой адрес почты вместо адреса администратора во избежание создания новых заявок через электронную почту. Она

используется во всех случаях, когда не допускается создание клиентами заявок через электронную почту.

2.2.7.4. АВТОМАТИЗАЦИЯ ПРОЦЕССОВ

2.2.7.4.1. Добавлена эскалация событий

- OTRS теперь создает события для каждого доступного типа эскалации (время ответа, обновления и решения). Это позволяет выполнять действия (такие как уведомления) до наступления времени эскалации, в момент наступления и окончания времени эскалации.

2.2.7.4.2. Улучшен механизм уведомлений

- Новый generic-agent-модуль уведомлений позволяет OTRS-администратору определить сообщения, которые будут показаны в веб-интерфейсе агентов, когда они войдут в систему.

2.2.7.4.3. Улучшен механизм расчета времени

- Все параметры времени теперь будут вычисляться только на базе данных сервера приложений, решая тем самым проблемы, возникавшие из-за разницы в данных часов между сервером приложений и сервером баз данных.

2.2.7.4.4. Улучшен GenericAgent

- Теперь GenericAgent может фильтровать заявки по времени изменения.
- Кроме того, GenericAgent может установить ответственность за заявку для схожих (соответствующих) заявок.

2.2.7.5. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС, ТЕКСТОВЫЙ РЕДАКТОР, КОДИРОВКА

2.2.7.5.1. Улучшилась производительность Пользовательского интерфейса

- Улучшена скорость рендеринга и отображения статей, большое спасибо Stelios Gikas <stelios.gikas@noris.net>!

2.2.7.5.2. Обновление текстового редактора Rich Text Editor

- Добавлена поддержка IOS5
- Блок цитат можно оставить с помощью клавиши ввода.
- Обновлен CKEditor 3.4 до CKEditor 3.6, про улучшения можно прочесть перейдя по ссылке [CKEditor 3.5](#) and [CKEditor 3.6](#).
- Улучшена поддержка браузера IE9.
- Диалоговые окна изменяемого размера.

2.2.7.5.3. Поддержка Unicode - Удалена поддержка кодировок, не относящихся к UTF-8

- UTF-8 теперь единственная внутренняя кодировка, разрешенная OTRS.
- Теперь все языковые файлы отформатированы в кодировке UTF-8, что значительно упрощает их обработку и дальнейшее усовершенствование механизма перевода.

2.2.7.6. ПОДДЕРЖКА ДРАЙВЕРОВ БАЗ ДАННЫХ

2.2.7.6.1. Улучшена совместимость с драйвером базы данных PostgreSQL

- Добавлена поддержка PostgreSQL 9.1
- Доступен новый драйвер для PostgreSQL 8.1 или более ранних версий.

2.2.7.6.2. Улучшена совместимость драйвера MS SQL

- Теперь драйвер MS SQL сохраняет бинарные данные используя тип данных VARBINARY а не устаревший тип TEXT, тип NVARCHAR также используется вместо устаревшего VARCHAR (это делается для улучшенной поддержки Unicode).

2.2.7.7. ИНТЕГРАЦИЯ С ПОЧТОЙ

2.2.7.7.1. Улучшилась обработка почты

- Теперь, при подключении к почтовым учетным записям IMAP можно обрабатывать сообщения не только из папки INBOX но и из другой определенной папки.
- Теперь OTRS также может подключаться к почтовым IMAP-серверам используя протокол Transport Layer Security (TLS), это очень полезная опция для современных ограниченных сред.

2.3. Требования к аппаратному и программному обеспечению

OTRS может быть установлен на многих операционных системах. OTRS работает не только на Linux и других UNIX-подобных ОС (таких как например, OpenBSD или FreeBSD), но также и на всех платформах Microsoft Windows. Для работы OTRS не требуется каких-нибудь специальных требований к оборудованию. Мы рекомендуем использовать машины, с процессором, как минимум 2 ГГц Xeon, 2 Гб Оперативной памяти и 160 Гб свободного пространства на жестком диске.

Если вы хотите использовать OTRS, вам потребуются некоторые другие компоненты программного обеспечения. Основные требования к программному обеспечению: веб-сервер и сервер баз данных, рабочая среда для выполнения Perl с некоторыми дополнительными модулями. Веб-сервер и Perl должны быть установлены на той же машине что и OTRS. База данных может быть установлена на локальной машине или же на другом хосте.

В качестве веб-сервера, мы рекомендуем использовать apache 2. Это связано с тем, что его модуль mod_perl позволяет существенным образом улучшить производительность OTRS. Если у вас нету возможности использовать Apache в качестве веб-сервера, OTRS может работать на любом другом веб-сервере, поддерживающем выполнение Perl-скриптов.

В качестве базы данных, можно использовать MySQL, PostgreSQL, Oracle или MSSQL. В случае использования MySQL в качестве сервера базы данных - есть возможность задать некоторые настройки посредством веб-интерфейса, при установке системы.

Для Perl, мы рекомендуем использовать по крайней мере версию 5.8.8. Также вам будут нужны дополнительные модули, которые могут быть установлены как при установке самого Perl и CPAN так и с помощью менеджера пакетов вашей операционной системы (rpm, yast, apt-get).

Требования к программному обеспечению

2.3.1. поддержка Perl

- Perl 5.10 или выше

2.3.2. Поддержка Веб-сервера

- Apache2 + mod_perl2 или выше (рекомендуется)
- Веб-сервер с поддержкой CGI-интерфейса (CGI-интерфейс не рекомендуется)
- Microsoft Internet Information Server (IIS) 6 или выше

2.3.3. Поддержка базы данных

- MySQL 5.0 или выше
- PostgreSQL 8.0 или выше (8.2 или выше, рекомендуется)
- Oracle 10g или выше
- Microsoft SQL Server 2005 или выше

Раздел руководства установка Perl-модулей более подробно описывает как можно настроить модули, необходимые для работы OTRS.

Если для установки OTRS используются бинарные пакеты, собранные специально под вашу операционную систему, то все необходимые модули языка Perl уже включены. Если нет, - используйте менеджер пакетов вашей операционной системы (например rpm, Windows-Инсталлятор) для установки всех необходимых модулей.

2.3.4. Поддержка веб-браузера

Для работы с OTRS необходимо использовать современных веб-браузер с включенной поддержкой JavaScript. Мы не поддерживаем следующие браузеры:

- Браузер Internet Explorer до 8 версии
- Браузер Firefox до 10 версии
- Браузер Safari до 5-й версии

Мы рекомендуем использовать самые последние версии браузеров, т.к. они обладают лучшей производительностью JavaScript и производительность рендеринга. Критическая разница в производительности на разных браузерах может возникать при больших объемах данных или в "тяжелых" системах. Мы всегда рады оказать помощь в таких случаях.

2.4. Сообщество

OTRS имеет большое сообщество пользователей. По ссылке списки рассылки пользователи и разработчики обсуждают OTRS, общаются на различные темы, связанные с системой обработки заявок. Также есть возможность обсуждать вопросы, касающиеся установки, настройки, использования, локализации и разработки приложения. Ошибки, замеченные в программном обеспечении отправляйте по ссылке система мониторинга ошибок . Ни одно сообщение не останется без внимания, все они гарантировано будут рассмотрены соответствующими разработчиками.

Домашняя страница OTRS-сообщества: <http://www.otrs.com/open-source/>.

2.5. Профессиональная поддержка OTRS

В случае потребности в помощи в конфигурировании или пользовательском изменении OTRS, а также обезопасить себя, не стесняйтесь обращаться к нам: мы предлагаем широкий набор профессиональных услуг, как то: глобальная поддержка предприятий, консультирование и проектирование, включающее дизайн системы, внедрение, доработку, поддержку приложений и полный сервис жизненного цикла.

Наш [Сервисный контракт](#) гарантирует помощь и профессиональную поддержку и, наконец, не в последнюю очередь бесплатный доступ к [OTRS Feature Add-ons/Дополнения к OTRS](#) - полезные дополнительные возможности для OTRS.

[The OTRS Group](#) предлагает специальные [программы тренингов](#) в разных странах. Вы можете принять участие в одном из наших публичных OTRS Administrator тренингах, проводимых регулярно, или заказать обучение на территории вашей компании, с учетом всех ее особенностей.



Глава 2. Установка

В этой главе описывается установка и базовая конфигурация главного OTRS-фреймворка. Здесь вы найдете информацию об установке OTRS из исходных кодов или из бинарных пакетов, например RPM, или с помощью Windows-инсталлятора.

В этой главе рассматриваются такие вопросы как: конфигурация веб-сервера и сервера базы данных, интерфейса между OTRS и базой данных, установка дополнительных Perl-модулей, установка соответствующих прав доступа для OTRS, настройка планировщика задач cron jobs для OTRS и основных параметров в конфигурационных файлах.

Следуйте подробным шагам этой главы чтобы установить OTRS на своем сервере. Потом можно использовать его веб-интерфейс чтобы войти в систему и производить администрирование.

1. Самый простой способ - установка из готовых бинарных пакетов

Самый простой и удобный способ установки OTRS - использовать готовые пакеты, которые можно найти и загрузить перейдя по ссылке <http://www.otrs.com>. В следующих разделах описывается установка OTRS из готовых или бинарных пакетов, специально собранных для операционных систем: SUSE, Red Hat и Microsoft Windows. В случае если нет возможности использовать готовые пакеты - установку придется производить вручную.

1.1. Установка из rpm-пакетов на сервер под управлением Suse Linux

В этом разделе описывается установка встроенных RPM пакетов на дистрибутив SUSE Linux. Мы протестировали все SLES и OpenSUSE релизы, доступные на данный момент. Перед началом установки перейдите по адресу <http://www.otrs.com/downloads> и убедитесь нет ли новых OTRS RPM пакетов, доступных для скачивания. Всегда старайтесь использовать только самые новые RPM пакеты.

1.1.1. Подготовка базы данных для OTRS

В качестве базы данных, можно использовать MySQL, PostgreSQL, Oracle or Microsoft SQL Server. Наиболее распространенной СУБД для использования с OTRS является MySQL. Эта глава описывает шаги необходимые для настройки MySQL на сервер с ОС SUSE. Конечно, вы можете установить СУБД на выделенных сервер баз данных, если это необходимо для масштабируемости или других целей.

Примечание

Если вы следуете описанному в этой главе на openSUSE 12.3 и еще не установили MySQL но вместо нее можно установить MariaDB , известный форк/ветку MySQL. Проблем не будет, все будет работать нормально (даже немного лучше в некоторых аспектах).

Установите MySQL с использованием следующей команды с правами пользователя root:


```
linux:~ # zypper install mysql perl-DBD-mysql
```

Будет установлена MySQL со стандартными настройками. Необходимо сделать некоторые изменения, для лучшей совместимости с OTRS. С помощью текстового редактора откройте файл `/etc/my.cnf` и измените строку с **max_allowed_packet** в нем, а также добавьте строку ниже следующим образом:

```
max_allowed_packet=20M
query_cache_size=32M
```

Теперь выполните **rcmysql restart** для рестарта сервера баз данных и включения сделанных изменений. Затем выполните **/usr/bin/mysql_secure_installation** и следуйте инструкциям на экране для установки пароля для пользователя root СУБД, удаления анонимного доступа и удаления тестовой базы данных. Наконец, исполните **chkconfig -a mysql** для проверки настройки автоматического запуска mysql одновременно со стартом сервера.

1.1.2. Установка OTRS

Установите OTRS с использованием **zypper** из командной строки. При этом будут выполнены связанные задачи, такие как установка веб-сервера Apache и ряд модулей Perl. Убедитесь, что OTRS RPM файл скопирован в текущий каталог.

```
otrs-sles:~ # zypper install otrs-3.3.*.rpm
....
Retrieving package otrs-3.3.3-01.noarch (1/26), 17.5 MiB (74.3 MiB unpacked)
Installing: otrs-3.3.3-01 [done]
Additional rpm output:
Check OTRS user ... otrs added.

Next steps:

[start database and Apache]
Make sure your database is running and execute 'rcapache2 restart'.

[install the OTRS database]
Use a webbrowser and open this link:
http://myserver.example.com/otrs/installer.pl

[OTRS services]
Start OTRS 'rcotrs start-force' (rcotrs {start|stop|status|restart|start-force|
    stop-force}).

((enjoy))

Your OTRS Team
http://otrs.org/

otrs-sles:~ #
```

Сценарий: Команда для установки OTRS.

Установка OTRS завершена. Чтобы новые изменения, касающиеся конфигурации OTRS, вступили в силу - перезагрузите веб-сервер используя команду из Сценария приведенного ниже. Также выполните **chkconfig** для уверенности в том, что OTRS автоматически стартует при перезагрузке сервера.

```
otrs-sles:~ # chkconfig -a apache2
```

```
apache2          0:off 1:off 2:off 3:on  4:off 5:on  6:off
otrs-sles:~ # rcapache2 start
Starting httpd2 (prefork) httpd2-prefork: Could not reliably determine the server's fully
qualified domain name, using 10.x.x.x for ServerName

done
otrs-sles:~ #
```

Скрипт: Запуск веб сервера.

1.1.3. Установка дополнительных perl модулей

OTRS требуются некоторые дополнительные модули, кроме тех, что устанавливаются с помощью RPM. Вы можете доустановить их вручную. Выяснить, какие дополнительные модули требуются, можно с помощью скрипта `bin/otrs.CheckModules.pl` размещенного в каталоге `/opt/otrs`. Некоторые модули требуются только в случае включения дополнительных опций, например для взаимодействия с серверами IMAP(S) или вывода в формате PDF. При использовании SLES необходимо подключить внешнее хранилище для конфигурации zypper для доступа к модулям, требующимся вашей системе. Выберите модуль для вашей версии ОС отсюда: <http://download.opensuse.org/repositories/devel:/languages:/perl/>. Добавьте репозиторий, например, для SLES 11 SP2:

```
zypper ar -f -n perl http://download.opensuse.org/repositories/devel:/
languages:/perl/SLE_11_SP2 Perl
```

В ОС openSUSE 12.3, внешний репозиторий требуется только для модуля `Mail::IMAPClient`, который необходим в случае, если вы забираете почту с сервера IMAP с TLS. Соответствующая строка будет выглядеть так:

```
zypper ar -f -n perl http://download.opensuse.org/repositories/devel:/
languages:/perl/openSUSE_12.3/ Perl
```

В первый раз, при использовании zypper после добавления строки с указанием репозитория, вас попросят ввести его ключ. Теперь, вы можете установить пропущенные модули, как показано ниже.

```
otrs-sles:/opt/otrs # zypper install -y "perl(YAML::LibYAML)"
Refreshing service 'susecloud'.
Retrieving repository 'perl' metadata [\]

New repository or package signing key received:
Key ID: DCCA98DDDCEF338C
Key Name: devel:languages:perl OBS Project <devel:languages:perl@build.opensuse.org>
Key Fingerprint: 36F0AC0BCA9D8AF2871703C5DCCA98DDDCEF338C
Key Created: Wed Oct 10 22:04:18 2012
Key Expires: Fri Dec 19 22:04:18 2014
Repository: perl

Do you want to reject the key, trust temporarily, or trust always? [r/t/a/?] (r): a
Retrieving repository 'perl' metadata [done]
Building repository 'perl' cache [done]
Loading repository data...
Reading installed packages...
'perl(YAML::LibYAML)' not found in package names. Trying capabilities.
Resolving package dependencies...

The following NEW package is going to be installed:
perl-YAML-LibYAML
```

```
The following package is not supported by its vendor:  
perl-YAML-LibYAML
```

```
Retrieving package perl-YAML-LibYAML-0.38-12.4.x86_64 (1/1), 75.0 KiB (196.0 KiB unpacked)  
Retrieving: perl-YAML-LibYAML-0.38-12.4.x86_64.rpm [done (55.7 KiB/s)]  
Installing: perl-YAML-LibYAML-0.38-12.4 [done]
```

Следующим шагом является настройка OTRS с помощью веб - установщика, как описано в в этом разделе.

1.2. Установка OTRS в операционной системе Red Hat Enterprise Linux или CentOS

В этом разделе описывается установка встроенных RPM пакетов на дистрибутив Red Hat Enterprise Linux (RHEL) или CentOS. Для OTRS 3.3 и выше, требуются RHEL 6 или CentOS 6, версия 5 не поддерживается. До начала установки посетите <http://www.otrs.com/downloads>, чтобы убедиться, что вы используете самые последние версии OTRS RPM пакетов.

1.2.1. Подготовка базы данных для OTRS

Для работы с OTRS можно использовать различные СУБД: MySQL, PostgreSQL, Oracle или Microsoft SQL Server. Наиболее популярной является MySQL. Эта глава показывает необходимые шаги для настройки MySQL на сервер с RHEL. Конечно, вы можете установить СУБД на выделенный сервер баз данных в целях масштабируемости или иных других.

Установите MySQL с использованием следующей команды с правами пользователя root:

```
[root@otrs-centos6 ~]# yum -y install mysql-server
```

Будет установлена MySQL со стандартными настройками. Необходимо сделать некоторые изменения, для лучшей совместимости с OTRS. С помощью текстового редактора откройте файл `/etc/my.cnf` добавьте следующие две строки в секции `[mysqld]`:

```
max_allowed_packet=20M  
query_cache_size=32M
```

Теперь выполните **rcmysql restart** для рестарта сервера баз данных и включения сделанных изменений. Затем выполните **/usr/bin/mysql_secure_installation** и следуйте инструкциям на экране для установки пароля для пользователя root СУБД, удаления анонимного доступа и удаления тестовой базы данных. Наконец, исполните **chkconfig -a mysql** для проверки настройки автоматического запуска mysql одновременно со стартом сервера.

1.2.2. Установка OTRS

Установите OTRS с использованием **yum** из командной строки. При этом будут выполнены связанные задачи, такие как установка веб-сервера Apache и ряда модулей Perl. Убедитесь, что OTRS RPM файл скопирован в текущий каталог.

```
[root@otrs-centos6 ~]# yum install --nogpgcheck otrs-3.3.*.rpm
...
Dependencies Resolved

=====
Package                Arch  Version                Repository                Size
=====
Installing:
otrs                    noarch 3.3.3-01                /otrs-3.3.3-01.noarch    74 M
Installing for dependencies:
apr                     x86_64 1.3.9-5.el6_2          updates                   123 k
...
procmail                x86_64 3.22-25.1.el6          base                       163 k

Transaction Summary
=====
Install      26 Package(s)

Total size: 80 M
Total download size: 6.0 M
Installed size: 88 M
Downloading Packages:
(1/25): apr-1.3.9-5.el6_2.x86_64.rpm          | 123 kB   00:00
...
(25/25): procmail-3.22-25.1.el6.x86_64.rpm    | 163 kB   00:00
-----
Total                                          887 kB/s | 6.0 MB   00:06
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : apr-1.3.9-5.el6_2.x86_64        1/26
  ...
  Installing : otrs-3.3.3-01.noarch            26/26
Check OTRS user ... otrs added.

Next steps:

[httpd services]
Restart httpd 'service httpd restart'

[install the OTRS database]
Make sure your database server is running.
Use a web browser and open this link:
http://myserver.example.com/otrs/installer.pl

[OTRS services]
Start OTRS 'service otrs start' (service otrs {start|stop|status|restart}).

((enjoy))

Your OTRS Team

Installed:
  otrs.noarch 0:3.3.3-01

Dependency Installed:
  ...

Complete!
[root@otrs-centos6 ~]#
```

Сценарий: Команда для установки OTRS.

Установка OTRS завершена. Осталось убедиться, что Apache запущен и будет всегда стартовать при любой перезагрузке сервера.

```
[root@otrs-centos6 ~]# chkconfig httpd on
[root@otrs-centos6 ~]# service httpd start
```

```
Starting httpd: httpd: Could not reliably determine the server's fully qualified domain
name, using 10.x.x.x for ServerName [ OK ]
[root@otrs-centos6 ~]#
```

Скрипт: Запуск веб сервера.

1.2.3. Установка дополнительных perl модулей

OTRS требуются некоторые дополнительные модули, кроме тех, что устанавливаются с помощью RPM. Вы можете доустановить их вручную. Выяснить, какие дополнительные модули требуются, можно с помощью скрипта `bin/otrs.CheckModules.pl` размещенного в каталоге `/opt/otrs`. Некоторые модули требуются только в случае включения дополнительных опций, например для взаимодействия с серверами IMAP(S) или вывода в формате PDF. На серверах с Red Hat или CentOS мы рекомендуем устанавливать эти модули из репозитория EPEL, который поддерживается проектом Fedora и производит высококачественные пакеты для RHEL и его ответвлений. Более подробно см. на [EPEL web сайт](#).

Если вы работаете на RHEL 6 или CentOS 6, наиболее свежие пакеты для EPEL можно загрузить с [этого сайта](#). Вы можете добавить этот репозиторий к yum в один шаг, копированием RPM URL, которую вы найдете на этой странице, и выполнив эту команду:

```
[root@otrs-centos6 otrs]# yum -y install http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
Loaded plugins: security
Setting up Install Process
epel-release-6-8.noarch.rpm | 14 kB 00:00
Examining /var/tmp/yum-root-7jrJef/epel-release-6-8.noarch.rpm: epel-release-6-8.noarch
Marking /var/tmp/yum-root-7jrJef/epel-release-6-8.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package epel-release.noarch 0:6-8 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
epel-release noarch 6-8 /epel-release-6-8.noarch 22 k

Transaction Summary
=====
Install 1 Package(s)

Total size: 22 k
Installed size: 22 k
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : epel-release-6-8.noarch 1/1
Verifying : epel-release-6-8.noarch 1/1

Installed:
epel-release.noarch 0:6-8

Complete!
[root@otrs-centos6 otrs]#
```

В первый раз, при использовании yum после добавления строки с указанием репозитория, вас попросят ввести его ключ. Теперь, вы можете установить пропущенные модули, как показано ниже.

```
[root@otrs-centos6 otrs]# yum -y install "perl(Text::CSV_XS)"
Loaded plugins: security
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package perl-Text-CSV_XS.x86_64 0:0.85-1.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch             Version          Repository        Size
=====
Installing:
perl-Text-CSV_XS        x86_64           0.85-1.el6       epel               71 k
=====
Transaction Summary
=====
Install      1 Package(s)

Total download size: 71 k
Installed size: 154 k
Downloading Packages:
perl-Text-CSV_XS-0.85-1.el6.x86_64.rpm | 71 kB    00:00
warning: rpmts_HdrFromFdno: Header V3 RSA/SHA256 Signature, key ID 0608b895: NOKEY
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
Importing GPG key 0x0608B895:
  Userid : EPEL (6) <epel@fedoraproject.org>
  Package: epel-release-6-8.noarch (@/epel-release-6-8.noarch)
  From    : /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
Is this ok [y/N]: y
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : perl-Text-CSV_XS-0.85-1.el6.x86_64      1/1
  Verifying  : perl-Text-CSV_XS-0.85-1.el6.x86_64      1/1

Installed:
  perl-Text-CSV_XS.x86_64 0:0.85-1.el6

Complete!
[root@otrs-centos6 otrs]#
```

Следующим шагом является настройка OTRS с помощью веб - установщика, как описано в в этом разделе.

1.2.4. Установка драйвера базы данных Oracle на Red Hat / CentOS

Для подключения СУБД Oracle к OTRS необходимо скомпилировать и установить DBD::Oracle драйвер для базы данных. Это немного сложнее чем установка других пакетов, т.к. Oracle является проприетарной СУБД и, так же как ни Red Hat ни CentOS, не разрешает публикацию драйверов в своих RPM репозиториях.

Прежде всего необходимо установить gcc, make и CPAN, после чего можно приступить к компиляции и установке драйвера. Ниже приведена команда для CentOS; для других версий она может выглядеть несколько иначе.

```
[root@otrs-centos6 otrs]# yum -y install gcc make "perl(CPAN)"
```

Следующим шагом является получение и установка клиента базы данных. Для этого надо зарегистрироваться и получить свободный аккаунт на сайте Oracle. Вы

можете загрузить драйверы с этой страницы: <http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html> Выберите версию для Linux x86 или x86-64, в зависимости от вашей системы. Вы можете узнать это с помощью команды **uname -i**. Это будет 'x86_64' для x86-64 или 'i386' для x86. Загрузите пакеты 'Instant Client Package - Basic', 'Instant Client Package - SQL*Plus', и 'Instant Client Package - SDK'. Сохраните их себе на диск. Теперь, от имени пользователя root вы можете установить пакеты с помощью следующей команды:

```
[root@otrs-centos6 otrs]# yum install oracle-instantclient*
```

После этого нужно задать значения двум переменным окружения и скомпилировать DBD::Oracle драйвер. Еще раз напомним, что выполнять это необходимо от имени пользователя root. Эти шаги изложены ниже. Обратите внимание, что для краткости некоторые строки, выдаваемые командой были удалены.

```
[root@otrs-centos6 otrs]# export ORACLE_HOME=/usr/lib/oracle/11.2/client64
[root@otrs-centos6 otrs]# export LD_LIBRARY_PATH=$ORACLE_HOME/lib
[root@otrs-centos6 otrs]# cpan
cpan[1]> look DBD::Oracle
...
Fetching with LWP:
  http://www.perl.org/CPAN/authors/id/P/PY/PYTHIAN/CHECKSUMS
Checksum for /root/.cpan/sources/authors/id/P/PY/PYTHIAN/DBD-Oracle-1.62.tar.gz ok
Scanning cache /root/.cpan/build for sizes
DONE
...
Working directory is /root/.cpan/build/DBD-Oracle-1.62-ZH6LNy
[root@localhost DBD-Oracle-1.62-ZH6LNy]# perl Makefile.PL
...
[root@localhost DBD-Oracle-1.62-ZH6LNy]# make
...
[root@localhost DBD-Oracle-1.62-ZH6LNy]# make install
...
cpan[2]> exit
Terminal does not support GetHistory.
Lockfile removed.
```

Теперь, надо отредактировать файл Kernel/Config.pm для установки параметра ORACLE_HOME. Следующим шагом является настройка OTRS с помощью веб - установщика, как описано в в этом разделе.

1.3. Установка OTRS на системы Debian или Ubuntu

Важно

Установите OTRS из исходного кода, и не используйте предоставленные Debian/Ubuntu пакеты для OTRS.

Установка требуемых модулей Perl будет легче, если использовать доступные пакеты:

```
apt-get install libapache2-mod-perl2 libdbd-mysql-perl libtimedate-perl libnet-dns-perl \
  libnet-ldap-perl libio-socket-ssl-perl libpdf-api2-perl libdbd-mysql-perl libsoap-lite-
perl \
  libgd-text-perl libtext-csv-xs-perl libjson-xs-perl libgd-graph-perl libapache-dbi-perl
```

1.4. Установка OTRS на операционной системе Microsoft Windows

1.4.1. Использование Windows Установщика

Устанавливать OTRS в операционной системе Microsoft Windows очень легко и просто. Перейдя по ссылке <http://www.otrs.com/try/> загрузите последнюю версию установщика для Win32-платформы и сохраните файл на жестком диске компьютера. Затем просто запустите установочный файл и выполните все шаги, предложенные мастером установки. Мастер установки произведет установку всех нужных компонентов для OTRS. Следующим шагом будет настройка OTRS с помощью веб-установщика, как описано в этом разделе.

Важно

Win-инсталлятор содержит все компоненты, необходимые для работы OTRS: веб-сервер Apache, сервер баз данных MySQL, Perl (со всеми необходимыми модулями), а также планировщик задач для Windowscron. Однако, вы можете его использовать для установки OTRS с Microsoft IIS Web Server вместо Apache. Также, можно сделать выбор и не устанавливать встроенную СУБД MySQL, например, если вы желаете работать с централизованной MySQL, PostgreSQL, на Microsoft SQL Server или с Oracle.

1.4.2. Установка драйвера базы данных Oracle на Windows

Если вы устанавливаете OTRS с использованием Win-установщика, он автоматически настроит все модули Perl, которые обычно используются. Есть одно исключение из сказанного: если вы будете использовать OTRS с Oracle, то потребуется доустановка драйвера Oracle. Это необходимо из-за того, что не существует открытого/open source драйвера для Oracle; мы не имеем права включать его в свои пакеты. Существует две версии perl для использования с Windows; можно выбрать Strawberry Perl, включенный в пакет и рекомендуемый для экспериментальных инсталляций; а можно выбрать Active State perl, который надо загрузить отдельно и установить. Последний вариант рекомендуется для рабочей среды.

1.4.2.1. Установка DBD::Oracle на ActiveState Perl

If you have installed ActiveState Perl it already comes with the Perl database driver DBD::Oracle only this should be linked to the Oracle C client. OTRS only works on 32-bit ActiveState Perl. This means you should also download the 32-bit Oracle client. Navigate to the Oracle client web site [Instant Client Downloads for Microsoft Windows \(32-bit\)](#) and choose 'Instant Client Package - Basic'. You'll need to sign up for a free Oracle Web account in order to download the software. Extract the download, a zip file, to the root of your drive, for instance C:\instantclient_11_2.

Now you should add the client libraries to your path and set an environment variable; 'ORACLE_HOME'. Navigate to Control Panel > System and Security > System, choose 'Advanced System Settings', click on 'New' under 'System variables', enter 'ORACLE_HOME' as Variable Name and use 'C:\instantclient_11_2' as the Variable Value. Select the 'PATH' variable, and append the directory 'C:\instantclient_11_2' to the path. If you now start a new cmd.exe session and run the otrs.CheckModules.pl script located in the OTRS directory, you should see that the DBD::Oracle driver is successfully installed. After this, you should restart the system in order for the other applications to detect the new environment variables.

1.4.2.2. Установка DBD::Oracle на Strawberry Perl

Для компиляции DBD::Oracle модуля, необходимо, чтобы установленные компоненты веб-установщика, находились в папке, путь к которой не содержал бы пробелов в именах. Он НЕ должен находиться в C:\Program Files\OTRS, а лучше, к примеру в C:\OTRSComponents или похожем.

On Strawberry Perl you should compile the Oracle driver manually. The OTRS Windows Installer bundles 32-bit Strawberry Perl. This means you should also download the 32-bit Oracle client. Navigate to the Oracle client web site [Instant Client Downloads for Microsoft Windows \(32-bit\)](#) and download the packages 'Instant Client Package - Basic', 'Instant Client Package - SQL*Plus', and 'Instant Client Package - SDK'. You'll need to sign up for a free Oracle Web account in order to download the software. Extract the downloads, all zip files, to the root of your drive, for instance C:\instantclient_11_2.

Now you should add the client libraries to your path and set an environment variable; 'ORACLE_HOME'. Navigate to Control Panel > System and Security > System, choose 'Advanced System Settings', click on 'New' under 'System variables', enter 'ORACLE_HOME' as Variable Name and use 'C:\instantclient_11_2' as the Variable Value. Select the 'PATH' variable, and append the directory 'C:\instantclient_11_2' to the path. After this, you should restart the system in order for the other applications to detect the new environment variables.

Now you can compile and install the DBD::Oracle module. For this, open a cmd.exe window, and start the cpan client. Download and extract DBD::Oracle via **look DBD::Oracle**, then type **perl Makefile.PL**, **dmake** and **dmake install**. Sample output is below.

```
C:\Users\max>cpan
cpan shell -- CPAN exploration and modules installation (v1.9800)
Enter 'h' for help.

cpan> look DBD::Oracle
Database was generated on Thu, 16 May 2013 14:52:41 GMT
Running look for module 'DBD::Oracle'

Trying to open a subshell in the build directory...
....
C:\OTRSCO~1\StrawberryPerl\cpan\build\DBD-Oracle-1.62-kq6los>perl Makefile.PL
Using DBI 1.623 (for perl 5.016003 on MSWin32-x86-multi-thread) installed in C:/
OTRSComponents/StrawberryPerl/perl/vendor/lib/auto/DBI/

Configuring DBD::Oracle for perl 5.016003 on MSWin32 (MSWin32-x86-multi-thread)
...
Checking if your kit is complete...
Looks good
LD_RUN_PATH=c:/instantclient_11_2/lib;c:/instantclient_11_2/rdbms/lib
Using DBD::Oracle 1.62.
Using DBD::Oracle 1.62.
Using DBI 1.623 (for perl 5.016003 on MSWin32-x86-multi-thread) installed in C:/
OTRSComponents/StrawberryPerl/perl/vendor/lib/auto/DBI/
Writing Makefile for DBD::Oracle
Writing MYMETA.yml and MYMETA.json
...

C:\OTRSCO~1\StrawberryPerl\cpan\build\DBD-Oracle-1.62-kq6los>dmake
... lots of output and compiler warnings ...

C:\OTRSCO~1\StrawberryPerl\cpan\build\DBD-Oracle-1.62-kq6los>dmake install
Files found in blib\arch: installing files in blib\lib into architecture depende
nt library tree
Installing C:\OTRSCO~1\StrawberryPerl\perl\site\lib\auto\DBD\Oracle\dbdimp.h
```

```
...
Appending installation info to C:\OTRSCO-1\StrawberryPerl\perl\lib\perllocal.pod
C:\OTRSCO-1\StrawberryPerl\cpan\build\DBD-Oracle-1.62-kq6los>exit

cpan> quit
Lockfile removed.

C:\Users\max>
```

1.4.2.3. Настройка OTRS с использованием Веб установщика

Веб-установщик может использоваться для настройки базовых параметров OTRS как описано в в этом разделе.

2. Установка из исходных кодов (Linux, Unix)

2.1. Подготовка к установке из исходных кодов

Если вы решили устанавливать OTRS из исходных кодов, перейдите по ссылке <http://www.otrs.com/try/> и загрузите архив с исходными кодами в любом удобном для вас формате: .tar.gz, .tar.bz2, или .zip

Распакуйте архив (например с помощью команды **tar**) в каталог /opt а затем переименуйте каталог otrs-x.x.x на otrs (см. ниже Сценарий ниже).

```
linux:/opt# tar xf /tmp/otrs-x.x.x.tar.gz
linux:/opt# mv otrs-x.x.x otrs
linux:/opt# ls
otrs
linux:/opt#
```

Script: Установка OTRS, первые шаги.

OTRS не должен выполняться с правами администратора root. На следующем шаге настройки OTRS необходимо добавить нового пользователя. Домашняя директория нового пользователя должна находиться в каталоге /opt/otrs. Если веб-сервер не работает с теми правами что и новый пользователь 'otrs', а именно так и есть в большинстве систем, то его нужно добавить в группу пользователей веб-сервера (см Сценарий ниже).

```
linux:/opt# useradd -r -d /opt/otrs/ -c 'OTRS user' otrs
linux:/opt# usermod -G www otrs
linux:/opt#
```

(SUSE=www, Red Hat/CentOS/Fedora=apache, Debian/Ubuntu=www-data)

Сценарий: Добавление нового пользователя 'otrs' и добавление его в группу.

Затем нужно скопировать некоторые конфигурационные файлы, точнее демонстрационные копии конфигурационных файлов, которые позже будут использоваться системой. Файлы находятся в каталоге /opt/otrs/Kernel и /opt/otrs/Kernel/Config и имеют расширение .dist (см. Сценарий ниже).

```
linux:/opt# cd otrs/Kernel/
```

```
linux:/opt/otrs/Kernel# cp Config.pm.dist Config.pm
linux:/opt/otrs/Kernel# cd Config
linux:/opt/otrs/Kernel/Config# cp GenericAgent.pm.dist GenericAgent.pm
```

Сценарий: Копирование демонстрационных копий конфигурационных файлов

На завершающем этапе установки OTRS необходимо установить соответствующие права доступа к файлам. Для этого используйте сценарий **otrs.SetPermissions.pl**, находящийся в директории `bin` домашнего каталога пользователя 'otrs'. Скрипт можно вызвать со следующими параметрами:

```
otrs.SetPermissions.pl {Home directory of the OTRS user} { --otrs-user= OTRS user} { --web-user= Web server user} [ --otrs-group= Group of the OTRS user] [ --web-group= Group of the web server user]
```

Если ваш веб-сервер работает с теми же правами что и пользователь 'otrs', то команда установки надлежащих прав доступа будет выглядеть так: **otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=otrs**. На SUSE-системах веб-сервер работает с правами пользователя 'wwwrun'. На Debian-системах - 'www-data'. Для установки надлежащих прав доступа используйте команду **otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=wwwrun --otrs-group=nogroup --web-group=www**.

2.2. Установка Perl-модулей

Исходя из Таблицы 3-1. для работы OTRS необходимо установить некоторые дополнительные модули Perl-. При установке OTRS из исходных кодов, эти модули придется установить вручную. Конечно же это намного проще сделать используя менеджер пакетов, который прилагается к вашему Linux-дистрибутиву (yast, apt-get) или, как описано в этом разделе, использовать оболочку Perl shell и CPAN. Если вы используете ActiveState Perl, например, на Windows, то можно использовать PPM, встроенный менеджер пакетов Perl/Perl Package Manager. Мы рекомендуем использовать менеджер пакетов, если это возможно.

Используйте команду **otrs.CheckModules.pl**, чтобы проверить, какие модули необходимо установить. Этот сценарий находится в директории `bin` домашнего каталога пользователя 'otrs' (см. Сценарий ниже). Некоторые модули необязательны.

```
linux:~# cd /opt/otrs/bin/
linux:/opt/otrs/bin# ./otrs.CheckModules.pl
  o CGI.....ok (v3.60)
  o Crypt::PasswdMD5.....ok (v1.3)
  o Crypt::SSLeay.....Not installed! (Optional - Required for Generic
Interface SOAP SSL connections.)
  o CSS::Minifier.....ok (v0.01)
  o Date::Format.....ok (v2.22)
  o Date::Pcalc.....ok (v1.2)
  ...
```

Сценарий: Проверка необходимых модулей.

Для установки недостающих модулей старайтесь использовать менеджер управления пакетами, входящий в дистрибутив Linux. Таким образом, в случае выхода обновлений или новых решений по обеспечению большей безопасности, пакеты будут обновляться автоматически. Чтобы узнать как установить дополнительные пакеты обратитесь к документации, которая поставляется вместе с дистрибутивом вашей операционной системы. Если модуль (соответствующая версия) не доступен из репозитория пакетов, его всегда можно установить используя CPAN, Comprehensive Perl Archive Network.

Для установки любого из выше описанных модулей через CPAN, нужно выполнить команду **perl -e shell -MCPAN**. Оболочка Perl запустится в интерактивном режиме и модуль CPAN будет загружен. Если CPAN уже сконфигурирован, то с помощью команды **install** "имя модуля" (`install module_name`) можно начинать установку необходимых модулей. CPAN автоматически отслеживает зависимости между Perl-модулями и тут же оповещает, какие еще модули нужно установить.

Execute also the commands **perl -cw bin/cgi-bin/index.pl** **perl -cw bin/cgi-bin/customer.pl** and **perl -cw bin/otrs.PostMaster.pl** after changing into the directory /opt/otrs. If the output of both commands is "syntax OK", your Perl is properly set up (see Script below).

```
linux:~# cd /opt/otrs
linux:/opt/otrs# perl -cw bin/cgi-bin/index.pl
cgi-bin/installer.pl syntax OK
linux:/opt/otrs# perl -cw bin/cgi-bin/customer.pl
cgi-bin/customer.pl syntax OK
linux:/opt/otrs# perl -cw bin/otrs.PostMaster.pl
bin/otrs.PostMaster.pl syntax OK
linux:/opt/otrs#
```

Сценарий: Проверить синтаксис.

2.3. Настройка веб-сервера Apache

Сначала, вы должны установить веб-сервер Apache2 и `mod_perl`; это обычно делается с помощью системного менеджера пакетов. Ниже вы найдете команды, необходимые для установки Apache на наиболее распространенные дистрибутивы Linux.

```
# rhel / centos: linux:# yum install httpd mod_perl # suse: linux:# zypper install apache2-mod_perl # debian/ubuntu: linux:# apt-get install apache2 libapache-mod-perl2
```

Для простого и удобного доступа к веб-интерфейсу OTRS через короткий адрес, нужно использовать `Alias` и `ScriptAlias`. Большинство установок Apache имеют директорию `conf.d`. Очень часто в Linux-системах эта директория находится в /etc/apache или /etc/apache2. Войдите в систему с правами администратора, затем перейдите в директорию `conf.d` и скопируйте соответствующий шаблон конфигурационного файла /opt/otrs/scripts/apache2-httpd.include.conf в файл `otrs.conf`, который находится в каталоге настроек Apache.

Перезагрузите веб-сервер, чтобы новые настройки вступили в силу. На большинстве систем запустить/перезагрузить веб-сервер можно с помощью команды **/etc/init.d/apache2 restart** (см. Сценарий ниже).

```
linux:/etc/apache2/conf.d# /etc/init.d/apache2 restart
Forcing reload of web server: Apache2.
linux:/etc/apache2/conf.d#
```

Сценарий: Перезапуск веб-сервера.

Теперь веб-сервер полностью настроен для работы OTRS.

2.4. Настройка базы данных

2.4.1. Установка базы данных OTRS вручную.

Если вы не можете воспользоваться веб-инсталлятором для установки базы данных OTRS, то ее нужно установить вручную, как описано в этой главе. Сценарий с

SQL-инструкциями для создания и настройки базы данных находится в директории `scripts/database` домашнего каталога пользователя 'otrs' (см. Сценарий ниже).

```
linux:~# cd /opt/otrs/scripts/database/  
linux:/opt/otrs/scripts/database# ls  
otrs-initial_insert.db2.sql      otrs-schema.mysql.sql  
otrs-schema.oracle.sql  
otrs-initial_insert.mssql.sql    otrs-schema-post.db2.sql  
otrs-initial_insert.mysql.sql    otrs-schema.postgresql.sql  
otrs-initial_insert.oracle.sql  
otrs-initial_insert.postgresql.sql otrs-schema-post.mssql.sql  
otrs-initial_insert.xml          otrs-schema-post.mysql.sql  
otrs-schema.db2.sql             otrs-schema-post.oracle.sql  
otrs-schema-post.postgresql.sql  
otrs-schema.mssql.sql           otrs-schema.xml  
linux:/opt/otrs/scripts/database#
```

Сценарий: Файлы, необходимые для создания и настройки базы данных.

При установке базы данных, для различных СУБД существует свой, определенный порядок обработки .sql-файлов.

Создание базы данных OTRS вручную, шаг за шагом

1. Создание базы данных: Используя интерфейс базы данных или свой любимый менеджер баз данных, создайте базу данных, которую планируете использовать для OTRS.
2. Создание таблиц: Используя файлы `otrs-schema.DatabaseType.sql` (например `otrs-schema.oracle.sql`, `otrs-schema.postgresql.sql`) можно создать таблицы в базе данных для OTRS.
3. Установка исходных данных системы: Чтобы OTRS работал должным образом, необходимо установить некоторые исходные системные данные (например различные состояния заявок, типы заявок и уведомлений). В зависимости от типа выбранной вами базы данных используйте один из следующих файлов `otrs-initial_insert.mysql.sql`, `otrs-initial_insert.oracle.sql`, `otrs-initial_insert.postgresql.sql` или `otrs-initial_insert.mssql.sql`.
4. Создание связей между таблицами: Последний шаг - создание связей между различными таблицами базы данных OTRS. Для этого используйте файл `otrs-schema-post.DatabaseType.sql` (например `otrs-schema-post.oracle.sql`, `otrs-schema-post.postgresql.sql`).

После завершения установки базы данных необходимо проверить и установить соответствующие права доступа для базы данных OTRS. Сделать это необходимо так, чтобы только один пользователь имел соответствующие права доступа. Настройка прав доступа отличается в зависимости от выбранного вами сервера базы данных и должна производиться с помощью графического интерфейса базы данных или с помощью программы клиента.

Если база данных и права доступа настроены правильно, то на следующем этапе ее нужно подключить к OTRS. Откройте файл `Kernel/Config.pm`, находящийся в домашнем каталоге пользователя 'otrs' и измените параметры в соответствии с вашими требованиями так, как это показано в Сценарии ниже.

```
# DatabaseHost  
# (The database host.)  
$Self->{'DatabaseHost'} = 'localhost';  
  
# Database  
# (The database name.)
```

```

$Self->{Database} = 'otrs';

# DatabaseUser
# (The database user.)
$Self->{DatabaseUser} = 'otrs';

# DatabasePw
# (The password of database user.)
$Self->{DatabasePw} = 'some-pass';

```

Сценарий: Настраиваемые параметры

2.5. Настрой планировщика задач (cron jobs) для OTRS

Для правильной работы системы OTRS необходим планировщик задач (cron jobs). Планировщик задач (cron jobs) должен запускаться с теми же правами, что и модули OTRS. Именно по этому cron jobs должен быть внесен в crontab-файл пользователя 'otrs'.

Все сценарии с cron jobs находятся в директории var/cron домашнего каталога пользователя 'otrs' (см. Сценарий ниже).

```

linux:~# cd /opt/otrs/var/cron
linux:/opt/otrs/var/cron# ls
aaa_base.dist          generic_agent.dist    rebuild_ticket_index.dist
cache.dist            pending_jobs.dist     session.dist
fetchmail.dist        postmaster.dist       unlock.dist
generic_agent-database.dist  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#

```

Сценарий: Файлы, необходимые для создания заданий планировщика задач (cron jobs).

Эти сценарии имеют расширение '.dist'. Их нужно скопировать в соответствующие файлы без расширения. Если вы работаете в оболочке bash, то можете воспользоваться командой из Сценария ниже.

```

linux:/opt/otrs/var/cron# for foo in *.dist; do cp $foo `basename $foo .dist`; done
linux:/opt/otrs/var/cron# ls
aaa_base          generic_agent-database.dist  rebuild_ticket_index
aaa_base.dist    generic_agent.dist          rebuild_ticket_index.dist
cache            pending_jobs                session
cache.dist      pending_jobs.dist          session.dist
fetchmail        postmaster                  unlock
fetchmail.dist  postmaster.dist            unlock.dist
generic_agent    postmaster_mailbox
generic_agent-database  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#

```

Сценарий: Копирование и переименование всех файлов, необходимых для создания заданий планировщика задач - cron jobs.

В Таблице 3-2 приводятся различные задания cron jobs.

Таблица 2.1. Описание некоторых сценариев планировщика задач cron job.

Сценарий	Назначение
aaa_base	Определяет основные настройки для crontab пользователя 'otrs'.

Сценарий	Назначение
cache	Удаляет из диска устаревшие ("просроченные") кэш-записи. Очищает кэш-погрузчик для CSS и JavaScript файлов.
fetchmail	Этот сценарий может быть использован в том случае, если новые емайлы будут поступать в систему обработки заявок через fetchmail.
generic_agent	Выполняет задания из GenericAgent, которые не сохраняются в базе данных, а в собственных конфиг-файлах.
generic_agent-database	Выполняет задания из GenericAgent, которые хранятся в базе данных.
pending_jobs	Проверяет систему на наличие заявок, ожидающих решения, закрывает их или отправляет напоминание, если это необходимо.
postmaster	Проверяет очередь сообщений системы обработки заявок, и доставляет те сообщения, которые находятся в этой очереди.
postmaster_mailbox	Получает почту с POP3-счетов, которые были указаны в админке, в разделе "Учетные записи PostMaster".
rebuild_ticket_index	Восстанавливает индекс заявки, что значительно повышает скорость просмотра заявок в разделе QueueView.
session	Удаляет старые и больше не используемые ID-сессий (session IDs).
unlock	Открывает заявки, которые были ранее заблокированы в системе.

Чтобы настроить все задания cron jobs используйте скрипт bin/Cron.sh, который находится в домашнем каталоге пользователя 'otrs'. При запуске, скрипту нужно передать один из трех параметров, указывающий на выполняемое действие: создание, удаление, переустановка заданий. Допустимы следующие параметры:

```
Cron.sh {start} {stop} {restart} [OTRS user]
```

Поскольку задания планировщика cron jobs должны быть установлены в crontab-файле пользователя 'otrs', то нужно залогиниться как пользователь 'otrs'. Если вы вошли в систему как суперпользователь, с правами администратора, и хотите изменить пользователя - используйте команду **su otrs**. Для установки заданий cron jobs выполните команды приведенные в Сценарии ниже.

Предупреждение

Обратите внимание, что при использовании файла Cron.sh другие задания, установленные в crontab-файле для пользователя 'otrs' будут перезаписаны или удалены. Внесите все необходимые изменения в файл Cron.sh, чтобы сохранить другие crontab-задания.

```
linux:/opt/otrs/var/cron# cd /opt/otrs/bin/
linux:/opt/otrs/bin# su otrs
linux:~/bin$ ./Cron.sh start
```

```
/opt/otrs/bin
Cron.sh - start/stop OTRS cronjobs
Copyright (C) 2001-2009 OTRS AG, http://otrs.org/
(using /opt/otrs) done
linux:~/bin$ exit
exit
linux:/opt/otrs/bin#
```

Сценарий: Установка cron jobs.

Команда **crontab -l -u otrs**, которую можно запускать с правами администратора, отображает содержание crontab-файла пользователя 'otrs', так что можно легко проверить правильность расположения всех записей (см. Сценарий ниже).

```
linux:/opt/otrs/bin# crontab -l -u otrs
# --
# cron/aaa_base - base crontab package
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# Who gets the cron emails?
MAILTO="root@localhost"
# --
# cron/cache - delete expired cache
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete expired cache weekly (Sunday mornings)
20 0 * * 0 $HOME/bin/otrs.CacheDelete.pl --expired >> /dev/null
30 0 * * 0 $HOME/bin/otrs.LoaderCache.pl -o delete >> /dev/null
# --
# cron/fetchmail - fetchmail cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch every 5 minutes emails via fetchmail
*/5 * * * * /usr/bin/fetchmail -a >> /dev/null
# --
# cron/generic_agent - otrs.GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 20 minutes
*/20 * * * * $HOME/bin/GenericAgent.pl >> /dev/null
# example to execute GenericAgent.pl on 23:00 with
# Kernel::Config::GenericAgentMove job file
#0 23 * * * $HOME/bin/otrs.GenericAgent.pl -c "Kernel::Config::GenericAgentMove" >> /dev/
null
# --
# cron/generic_agent - GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 10 minutes
*/10 * * * * $HOME/bin/otrs.GenericAgent.pl -c db >> /dev/null
# --
# cron/pending_jobs - pending_jobs cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check every 120 min the pending jobs
```



```
45 */2 * * * $HOME/bin/otrs.PendingJobs.pl >> /dev/null
# --
# cron/postmaster - postmaster cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check daily the spool directory of OTRS
10 0 * * * $HOME/bin/otrs.ReprocessMails.pl >> /dev/null
# --
# cron/postmaster_mailbox - postmaster_mailbox cron of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch emails every 10 minutes
*/10 * * * * $HOME/bin/otrs.PostMasterMailbox.pl >> /dev/null
# --
# cron/rebuild_ticket_index - rebuild ticket index for OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# just every day
01 01 * * * $HOME/bin/otrs.RebuildTicketIndex.pl >> /dev/null

# --
# cron/session - delete old session ids of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete every 120 minutes old/idle session ids
55 */2 * * * $HOME/bin/otrs.DeleteSessionIDs.pl --expired >> /dev/null

# --
# cron/unlock - unlock old locked ticket of the OTRS
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# unlock every hour old locked tickets
35 * * * * $HOME/bin/otrs.UnlockTickets.pl --timeout >> /dev/null

linux:/opt/otrs/bin#
```

Сценарий: Файл *Crontab*.

2.6. Планировщик OTRS

Планировщик это отдельный процесс OTRS выполняющий асинхронные задания

При установке из OTRS RPM сервис Планировщика устанавливается автоматически. При установке из исходных кодов, его можно установить копированием `scripts/otrs-scheduler-linux` файла в `/etc/init.d` и назначив соответствующие права. Это обеспечит запуск сервиса Планировщика при старте системы.

3. Использование веб установщика

Вы можете использовать веб-установщик OTRS, после установки OTRS для установки и настройки базы данных OTRS. Веб-установщик это страница браузера, которую можно открыть по ссылке <http://localhost/otrs/installer.pl>.

Когда запустится веб-инсталлятор, выполните, пожалуйста, следующие шаги для установки системы:

1. Проверьте информацию об OTRS-офисах и нажмите кнопку Далее (см. Рис ниже).

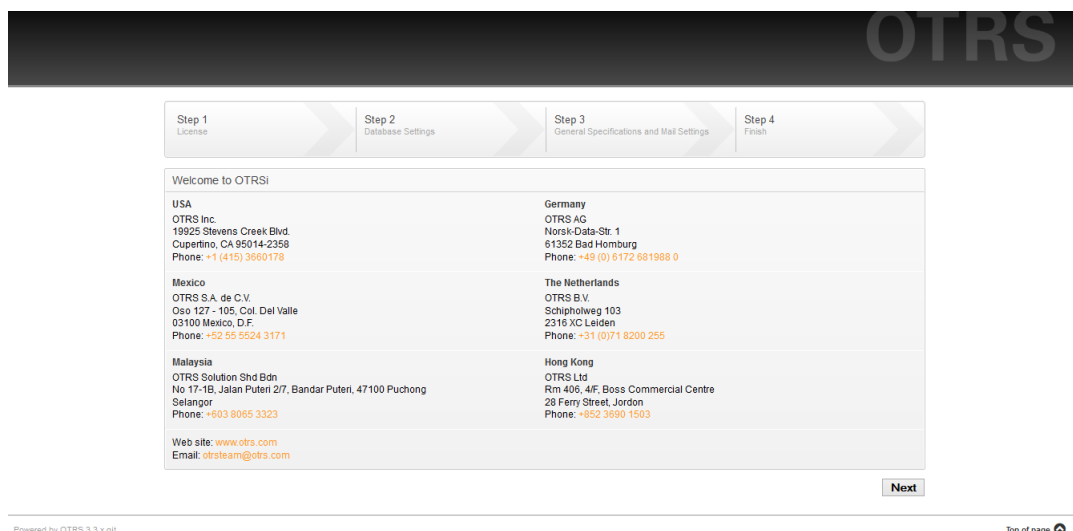


Рисунок: Экран приветствия.

2. Прочитайте соглашения лицензии GNU Affero General Public License (см. Рисунок ниже) и примите их, нажав на соответствующую кнопку внизу страницы.

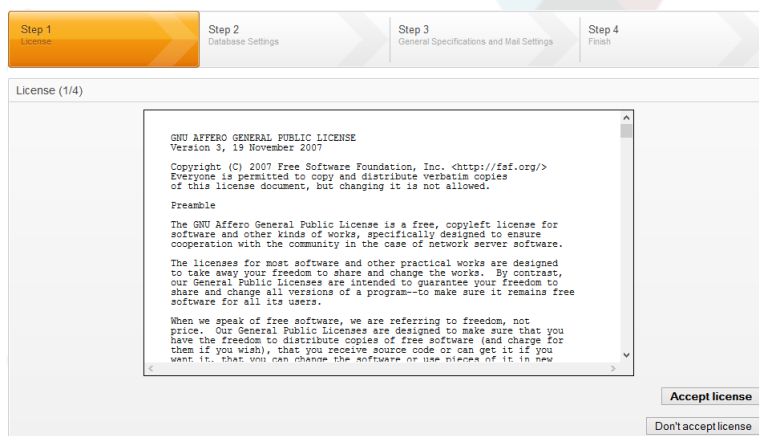


Рисунок: GNU Affero General Public License.

3. Выберите СУБД, которую вы будете использовать с OTRS. Если вы выбрали MySQL, PostgreSQL или Microsoft SQL Server в качестве СУБД, вы также можете выбрать, будете ли вы использовать веб-установщик для создания базы данных или ваш администратор БД уже создал пустую базу данных. После этого нажмите кнопку "Далее" (см. рисунок ниже).

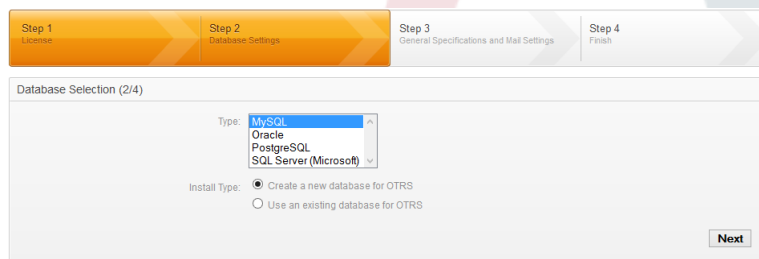


Рисунок: Выбор Базы данных.

4. В зависимости от того, какую СУБД вы выбрали и желаете использовать веб-установщик для создания БД или использовать существующую БД, на предыдущем

шаге, этот экран может иметь различный вид. Введите ваши данные/полномочия для доступа к БД на этом экране.

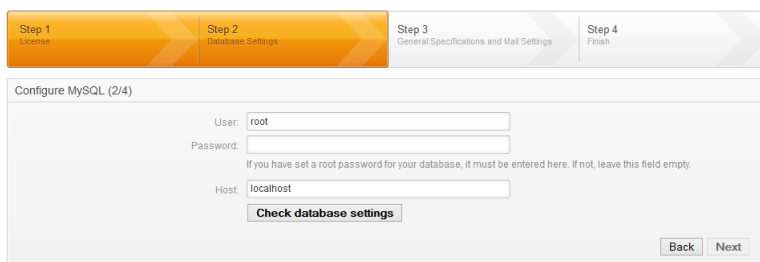


Figure: Database credentials.

5. Создайте нового пользователя базы данных, выберите имя базы данных и нажмите кнопку 'Далее' (см. Рисунок ниже).

Предупреждение

OTRS сгенерирует надежный пароль для вас. Можно ввести свой собственный пароль. Пароль будет записан в конфигурационном файле `Kernel/Config.pm`, так что запоминать его необязательно.

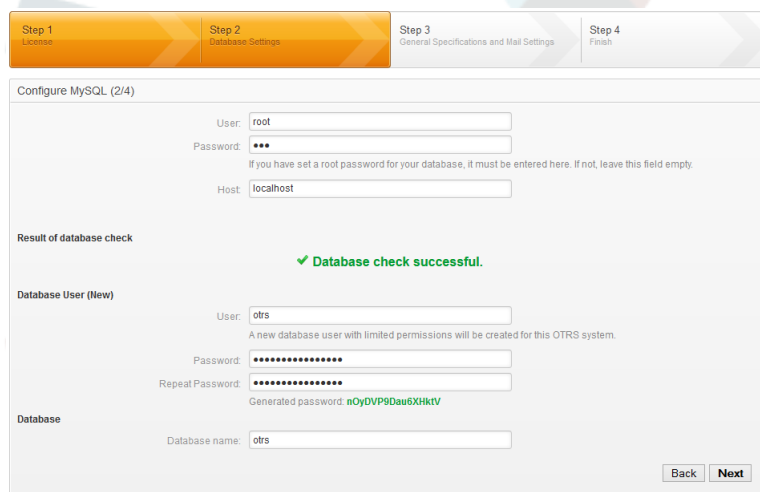


Рисунок: Настройки базы данных.

6. База данных будет создана и заполнена, как показано на рисунке. Нажмите "Далее" для перехода к следующему экрану.

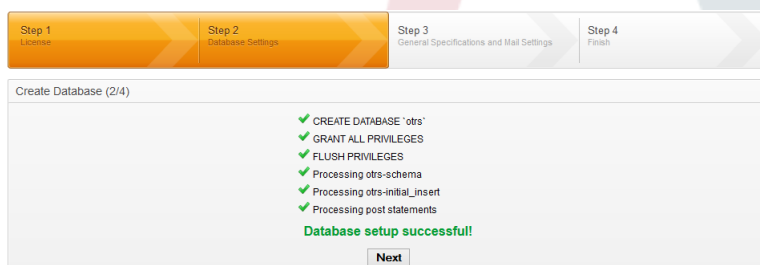


Рисунок: Уведомление об успешной установке базы данных.

7. Укажите все необходимые настройки системы и нажмите кнопку 'Далее' (см. Рисунок ниже).

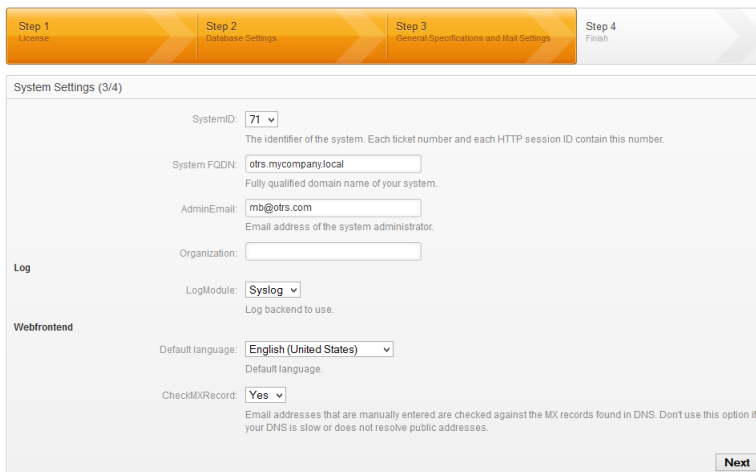


Рисунок: Настройки системы.

8. Если хотите, можно указать необходимые данные для настройки входящей и исходящей почты, можете пропустить этот шаг, нажав кнопку в нижней части экрана (см. Рисунок ниже).

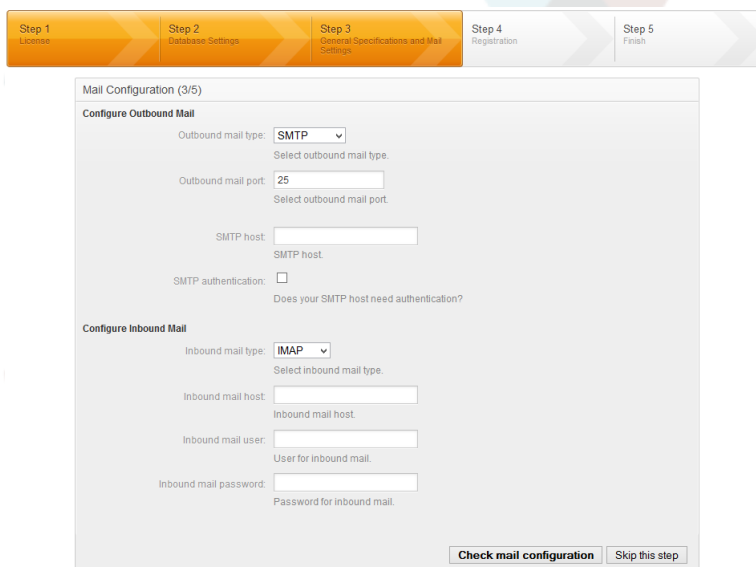


Рисунок: Настройка электронной почты.

9. Поздравляем! Установка OTRS завершена и можно начинать работу с системой (см. Рисунок ниже). Для входа в веб-интерфейс OTRS, откройте веб-браузер и наберите в адресной строке <http://localhost/otrs/index.pl>. Чтобы войти в систему OTRS с правами администратора, используйте логин 'root@localhost' и сгенерированный системой пароль. Затем вы можете настроить систему в соответствии с вашими требованиями.

Предупреждение

Пожалуйста, запишите пароль сгенерированный системой для учетной записи 'root@localhost'.

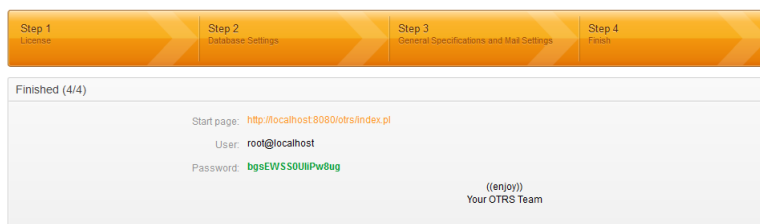


Рисунок: Финальная страница Веб установщика.

4. Обновление OTRS-фреймворка

Эти инструкции предназначены для тех, кто обновляет OTRS 3.2 до версии 3.3, их можно применять как при обновлении системы из RPM так и в случае использования исходных кодов (тарболов).

Примечание

Учтите, что OTRS 3.3 требует версии Perl, как минимум 5.10.0. Перед обновлением, убедитесь, что ваш сервер работает с этой версией. Проверить версию Perl можно с помощью команды **perl -v** в командной строке. Единственный известный дистрибутив Linux, который использует perl 5.8 и до сих пор поддерживается разработчиком это Red Hat Enterprise Linux (RHEL) 5 и его производный CentOS 5. Если вы работаете с этими платформами и планируете переход на OTRS 3.3, вы также должны запланировать обновление вашей операционной системы до версии которая поддерживает нужную версию perl, например, RHEL 6 или CentOS 6.

Если вы используете более старую версию OTRS, то сначала выполните обновление до версии 3.1 (1.1->1.2->1.3->2.0->2.1->2.2->2.3->2.4->3.0->3.1->3.2->3.3 ...)! Необходимо выполнить полное обновление до каждой версии в промежутке, включая преобразование баз данных и скрипты perl для обновления.

Обратите внимание, что если вы обновляете OTRS 2.2 или выше, вы должны выполнить дополнительные шаги, о которых написано здесь http://bugs.otrs.org/show_bug.cgi?id=6798.

Внутри одной версии, вы можете пропустить промежуточные релизы/patch level при обновлении. К примеру, вы можете провести обновление прямо с OTRS 3.3.1 до версии 3.3.4. В этом случае можно пропустить шаги 9 и 13-16.

Обратите внимание, что если вы производите промежуточное обновление с версии OTRS 3.3.6 или более ранней, необходимо активизировать новое задание для cron, управляющее процессом Планировщика/Scheduler . Скопируйте файл /opt/otrs/var/cron/scheduler_watchdog.dist в /opt/otrs/var/cron/scheduler_watchdog и обновите crontab - /opt/otrs/bin/Cron.sh start (как пользователь "otrs", не "root").

1. Остановите все соответствующие службы.

в т.ч. (зависящие от использующихся услуг):

```
shell> /etc/init.d/cron stop
shell> /etc/init.d/postfix stop
shell> /etc/init.d/apache stop
```

2. Сделайте резервные копии всех данных из \$OTRS_HOME (по умолчанию OTRS_HOME=/opt/otrs):

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm
- var/*
- а также базу данных

3. Убедитесь, что вы создали резервную копию всех данных ;-)

4. Установить новую систему (опционально)

Если есть возможность, сначала попробуйте произвести установку на отдельном тестовом компьютере.

5. Установка нового релиза (из tar-архива или с помощью RPM-пакетов).

- С помощью тарбола (tarball):

```
shell> cd /opt
shell> mv otrs otrs-old
shell> tar -xzf otrs-x.x.x.tar.gz
shell> mv otrs-x.x.x otrs
```

Восстановить старые конфигурационные файлы

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm

Восстановить TicketCounter.log

Чтобы OTRS продолжил работу с правильными номерами заявок, восстановите файл TicketCounter.log в \$OTRS_HOME/var/log/ (по умолчанию: OTRS_HOME=/opt/otrs). Это особенно важно если вы используете incremental/последовательную нумерацию заявок.

Восстановить данные статьи

Если вы настроили хранение данных сообщений/заметок в OTRS в файловой системе нужно восстановить article папку в \$OTRS_HOME/var/ (по умолчанию: OTRS_HOME=/opt/otrs).

- С помощью RPM-пакетов:

```
shell> rpm -Uvh otrs-x.x.x.-01.rpm
```

В этом случае обновление из RPM автоматически восстанавливает старые конфигурационные файлы.

6. Собственные темы

Примечание: Темы OTRS 3.2 и 3.3 НЕ совместимы между собой, так что не используйте старые темы!

Темы хранятся в `$OTRS_HOME/Kernel/Output/HTML/*/*.dtl` (default: `OTRS_HOME=/opt/otrs`).

Помните также, что `.dtl`-файлы, начиная с версии OTRS 3.3 должны быть только в кодировке `utf-8`.

7. Установка прав доступа к файлам.

В случае использования тарбола (`tarball`), выполните

```
shell> cd /opt/otrs/  
shell> bin/otrs.SetPermissions.pl
```

с правами, необходимыми для установки системы.

8. Проверка требуемых Perl модулей

Проверьте наличие всех требуемых `perl` модулей и доустановите те, которые могли быть пропущены по разным причинам.

```
shell> /opt/otrs/bin/otrs.CheckModules.pl
```

9. Схема обновления:

Примечание: новые таблицы, созданные в процессе MySQL UPGRADING будут созданы с методом доступа по умолчанию, установленном в вашем MySQL сервере. В MySQL 5.5 применяется новый метод - InnoDB. Если существующие таблицы, например, "users", имеют метод доступа, например, MyISAM, может появиться сообщение об ошибке при создании внешних ключей.

Имеется два варианта: (1) вы можете изменить стандартный метод доступа MySQL обратно в MyISAM, при этом новые таблицы будут иметь тот же метод, что и старые, или (2) перенастроить существующие таблицы на использование InnoDB.

Любые проблемы, связанные с использованием метода доступа будут отражены в отчете скрипта `otrs.CheckDB.pl`. Выполните его для обнаружения возможных проблем.

```
shell> cd /opt/otrs/  
  
# MySQL:  
shell> bin/otrs.CheckDB.pl  
shell> cat scripts/DBUpdate-to-3.3.mysql.sql | mysql -p -f -u root otrs  
  
# PostgreSQL 8.2+:  
shell> cat scripts/DBUpdate-to-3.3.postgresql.sql | psql --set ON_ERROR_STOP=on --single-transaction otrs otrs  
  
# PostgreSQL, older versions:  
shell> cat scripts/DBUpdate-to-3.3.postgresql_before_8_2.sql | psql --set ON_ERROR_STOP=on --single-transaction otrs otrs
```

Примечание: При использовании СУБД PostgreSQL 8.1 или более ранней, необходимо включить использование нового драйвера для этих старых версий. Выполните это путем включения новой строки в ваш `Kernel/Config.pm` типа:

```
$self->{DatabasePostgresqlBefore82} = 1;
```

Запустите сценарий перемещения (как пользователь 'otrs', а НЕ root):

```
shell> scripts/DBUpdate-to-3.3.pl
```

Не продолжайте процесс обновления, если этот скрипт обрабатывает неправильно, по вашему мнению. Иначе, вы можете потерять данные.

10Обновите конфигурацию системы и удалите все данные из кэша. Пожалуйста, запустите (от имени 'otrs', НЕ 'root'):

```
shell> bin/otrs.RebuildConfig.pl  
shell> bin/otrs.DeleteCache.pl
```

11Перезапустите сервисы.

в т.ч. (зависящие от использующихся услуг):

```
shell> /etc/init.d/cron start  
shell> /etc/init.d/postfix start  
shell> /etc/init.d/apache start
```

Теперь вы можете войти в систему.

12Проверьте установленные пакеты

Проверьте в менеджере пакетов все ли пакеты по прежнему помечены как правильно установлены или какие либо из них требуют переустановки или же обновления.

Следующие пакеты будут автоматически деинсталлированы после процесса обновления (если были установлены ранее):

- OTRSPostMasterFilterExtensions
- OTRSFreeTextFromCustomerUser
- OTRSExternalTicketNumberRecognition
- OTRSDashboardQueueOverview
- OTRSImportantArticles
- OTRSImportantArticlesITSM
- OTRSDashboardTicketCalendar
- OTRSMultiServiceSelect
- OTRSMultiQueueSelect
- OTRSDynamicFieldMultiLevelSelection
- OTRSEventBasedTicketActions
- OTRSTicketAclEditor
- OTRSCustomerProcessSelection

- OTRSACLExtensions
- OTRSGenericStandardTemplates
- OTRSExtendedDynamicDateFieldSearch
- OTRSDashboardTicketOverviewFilters

13 Проверка настроек конфигурации OTRSFreeTextFromCustomerUser

Примечание: Этот шаг применяется лишь только в том случае, если вы использовали пакет OTRSFreeTextFromCustomerUser ранее.

Если вы использовали этот модуль ранее, необходимо его перенастроить. Модуль будет автоматически деинсталлирован скриптом обновления, т.к. он теперь непосредственно включен в OTRS.

Если вы желаете использовать эту возможность, включите параметр "Ticket::EventModulePost###930-DynamicFieldFromCustomerUser" для активизации ее и сделайте настройки в параметре "DynamicFieldFromCustomerUser::Mapping".

14 Импортируйте ваши списки доступа/ACL в новый редактор ACL (по желанию)

В версии OTRS 3.3 появился новый графический редактор ACL в Панели Администратора. Необходимо перенести все ваши существующие ACL (например, из Config.pm или дополнительных файлов) в редактор с помощью скрипта bin/otrs.ImportACLsFromConfig.pl, чтобы, в дальнейшем, их можно было использовать в редакторе. После успешного завершения процедуры импорта убедитесь, что вы удалили все ACL из Config.pm (или других файлов). Также, необходимо использовать кнопку "Синхронизировать/Deploy" в редакторе ACL для синхронизации всех импортированных ACL в системе.

15 Обновление конфигурации веб сервера

Примечание: это применимо только если вы используете веб-сервер Apache, не используйте файл конфигурации непосредственно в каталоге установки OTRS (например, в символической ссылке из каталога конфигурации Apache).

Обновите файл конфигурации Apache для OTRS, т.к. были сделаны некоторые изменения (см. scripts/apache2-httpd.include.conf).

16 Обновление и активация планировщика задач cronjobs

В OTRS есть несколько стандартных заданий cron в файле \$OTRS_HOME/var/cron/*.dist. Для их применения скопируйте их в файл, опустив в окончании его имени расширение ".dist". Выполните это, чтобы быть уверенным в том, что вы используете последние версии заданий cron и вновь включенные задания.

```
shell> cd var/cron
shell> for foo in *.dist; do cp $foo `basename $foo .dist`; done
```

Проверьте скопированные файлы и внесите пользовательские изменения или/и дополнения, которые вы, возможно, сделали в предыдущих версиях.

Для применения этих cronjobs в вашей системе выполните скрипт Cron.sh. Выполняйте его от имени системного пользователя OTRS!

```
shell> /opt/otrs/bin/Cron.sh start
```

17Молодцы!

5. Обновление с помощью Windows Installer

Если вы установили OTRS на Windows, его можно легко обновить, загрузив новую версию win-установщика и запустив ее. Установщик автоматически обнаружит установленную OTRS и выполнит ее обновление. Он остановит сервисы OTRS и стартует их заново после обновления, так, что ваши пользователи могут и не заметить простоя. Тем не менее запланируйте достаточное время для обновления. Также, убедитесь, что вы имеете резервные копии сервера до начала процедуры обновления!

5.1. Выполнение промежуточных/patch level обновлений на Windows

Если вам необходимо установить 'patch level' обновление (например, с OTRS 3.3.1 до OTRS 3.3.2) на Windows, все что вам нужно - это загрузить новый установщик и запустить его. Конечно, как всегда, предварительно, должны быть сделаны все нужные резервные копии!

5.2. Выполнение обновлений "малого" уровня на Windows

"Малые" обновления (от OTRS 3.2 до OTRS 3.3) требуют немного больше времени, т.к. они зачастую, включают шаг реорганизации базы данных. Установщик определит это и автоматически выполнит обновление. После установки нужно будет выполнить шаги 12, 13 и 14 прилагаемой к релизу инструкции раздел по обновлению.

Учтите, что если в вашей системе установлены дополнительные пакеты, такие как Feature Add-Ons, то их также потребуется обновить. Убедитесь в том, что вы получили их до начала обновления системы. После окончания обновления системы, вы можете установить эти пакеты используя раздел "Управление пакетами" в Панели Администратора. Пакеты из "стандартного" репозитория, такие как Survey, FAQ и ITSM, могут быть обновлены после окончания обновления системы.

6. Дополнительные приложения

Для расширения функциональности OTRS-фреймворка, можно также установить дополнительные пакеты приложений. Это можно сделать с помощью менеджера пакетов из Панели Администрирования, который загружает приложения из онлайн репозитория и управляет зависимостями между этими пакетами. Также приложения можно установить из локальных файлов.

6.1. FAQ (Часто Задаваемые Вопросы)

FAQ - это компонент Базы Знаний. Он позволяет редактировать и просматривать статьи FAQ. Статьи можно просматривать и ограничивать к ним доступ для агентов, клиентов и анонимных пользователей. Также они могут быть структурированы по группам и быть прочитаны на различных языках.

Глава 3. Первые шаги

В этой главе приводится общий обзор OTRS и структура веб-интерфейса. Дается объяснение таким терминам как: агенты, клиенты и администраторы. В этом разделе вы войдете в систему с правами OTRS-администратора и более внимательно ознакомитесь с настройками пользователей, доступными для каждого из типов аккаунта.

1. Веб-интерфейс для агентов

Агентский веб-интерфейс позволяет агентам отвечать на вопросы клиентов, создавать новые заявки для клиентов и агентов, создавать заявки на основе телефонных звонков клиентов, писать и редактировать записи FAQ-модуля, редактировать данные клиентов и так далее.

Предположим, что OTRS доступен по ссылке <http://www.example.com>, таким образом, агенты и OTRS-администраторы могут войти в OTRS набрав в строке браузера следующий URL-адрес: <http://www.example.com/otrs/index.pl> (см. Рисунок ниже).

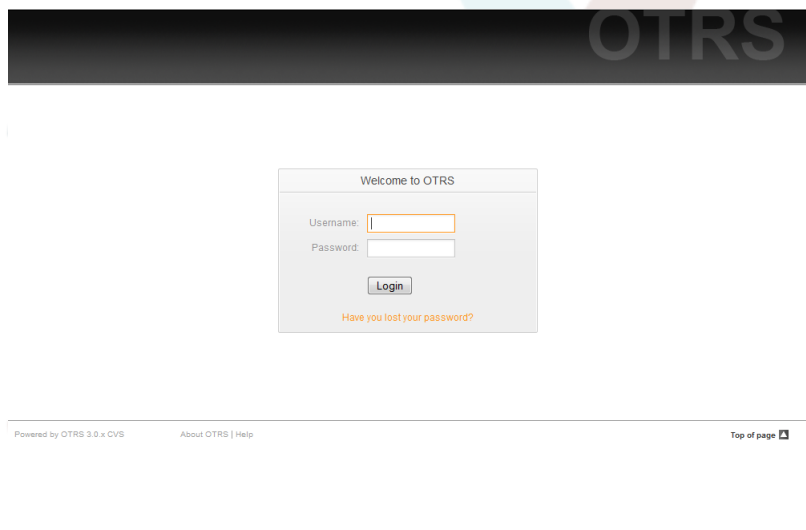


Рисунок: Агентский интерфейс, экран входа.

2. Пользовательский веб-интерфейс

В OTRS клиенты имеют отдельный веб-интерфейс, с помощью этого веб-интерфейса они могут создавать новые учетные записи, просматривать собственные заявки, создавать и редактировать заявки, изменять настройки своей учетной записи и т.д.

Используя выше приведенный пример, страница входа для клиентов будет доступна по ссылке <http://www.example.com/otrs/customer.pl> (см. Рисунок ниже).

Company Support

Login

Username Password

[Forgot password?](#)

Not yet registered? [Sign up now.](#)

Powered by OTRS 3.0.x CVS

Рисунок: Пользовательский веб-интерфейс, страница входа.

3. Общедоступный веб-интерфейс

В дополнение к веб-интерфейсу для агентов и клиентов, OTRS имеет открытый веб-интерфейс, предоставляющий доступ к FAQ-модулю, который устанавливается отдельно. Он предоставляет открытый доступ к FAQ-системе и позволяет пользователям без авторизации искать информацию в базе знаний.

В нашем примере общий веб-интерфейс будет доступен по ссылке: <http://www.example.com/otrs/faq.pl> , <http://www.example.com/otrs/public.pl>

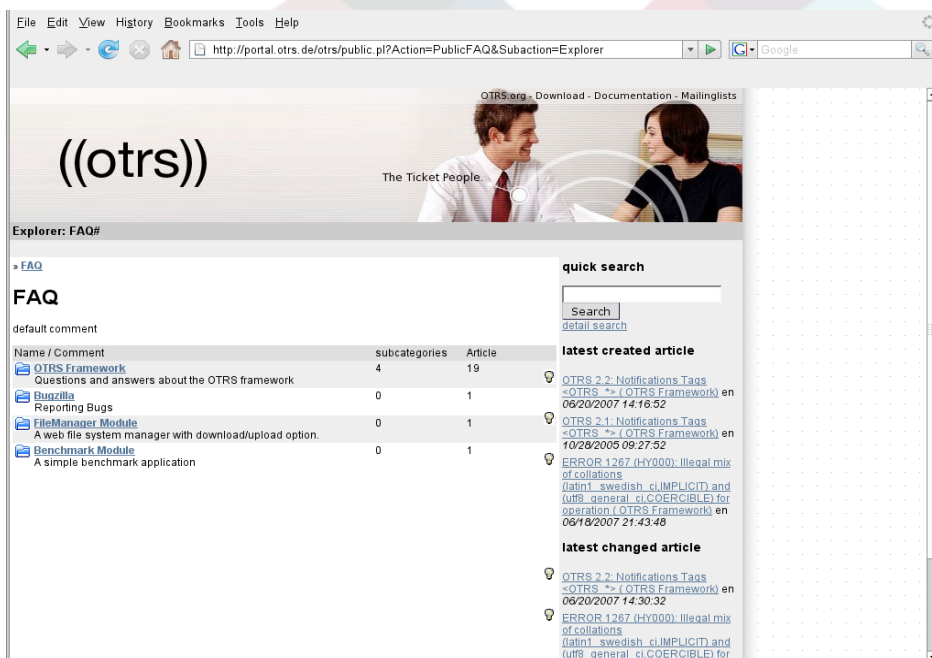


Рисунок: Общий веб-интерфейс.

4. Первое применение

Перейдите к странице входа, как это описано в разделе Агентский веб-интерфейс . Введите имя пользователя и пароль. Ввиду того что после новой установки в систе-

ме нет ни одного зарегистрированного пользователя, войдите в первый раз с правами OTRS-администратора, используя для этого имя пользователя 'root@localhost' и пароль 'root'.

Предупреждение

Данные доступа этого аккаунта действительны для каждой копии новоустановленной системы OTRS. Именно по этой причине пароль OTRS-администратора необходимо изменить как можно быстрее. Это можно сделать после входа в систему с правами администратора перейдя по ссылке preferences.

Если не хотите входить в систему с правами OTRS-администратора, просто введите имя и пароль своего обычного агентского аккаунта.

В случае, если вы забыли пароль всегда можно отправить запрос в систему для получения нового. Для этого просто нажмите на ссылку, которая находится под кнопкой Вход, введите адрес своей электронной почты, который вы указали в настройках своего OTRS-аккаунта и нажмите кнопку Отправить (см. Рисунок).

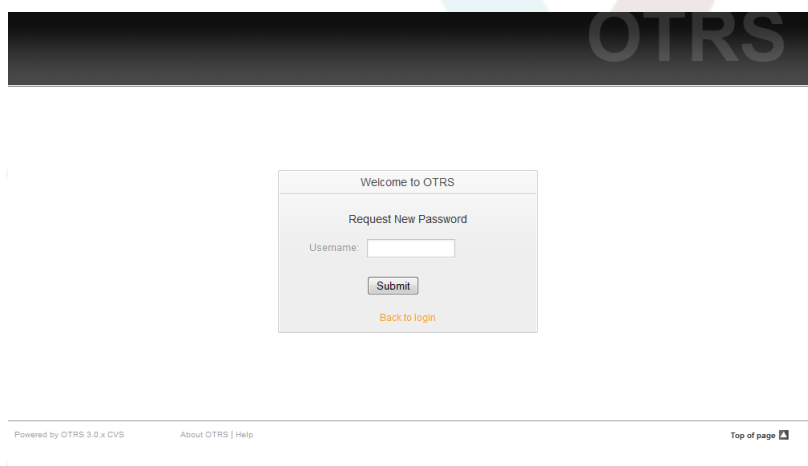


Рисунок: Запрос на восстановление пароля.

5. Веб-интерфейс - обзор

После успешного входа в систему, вы попадаете на страницу Дайджеста (см. рисунок ниже) и получаете доступ к меню системы и иконкам быстрого доступа. В ней отображаются ваши заблокированные заявки, с нее можно быстро перейти к просмотру: очередей, статусов и эскалаций, созданию новых заявок, на основе телефонных звонков и электронных писем. Также она предоставляет краткую сводку по новым, ожидающим, эскалированным и открытым заявкам.

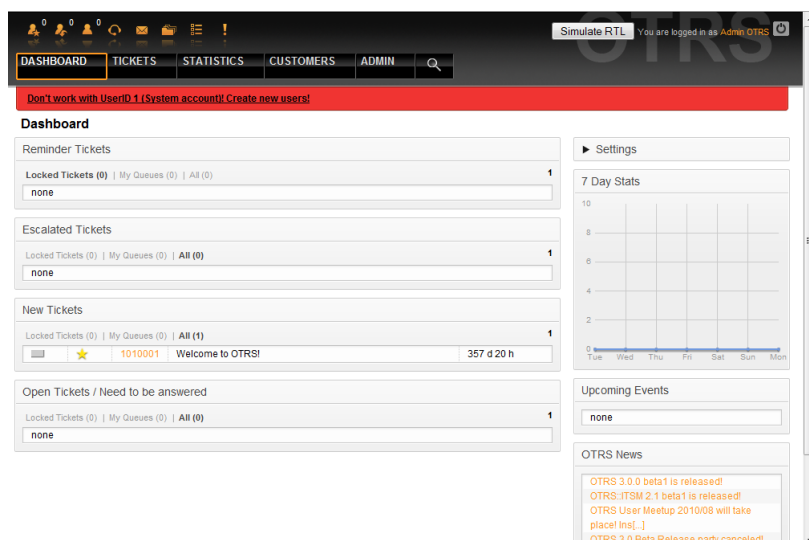


Рисунок: Панель управления веб-интерфейса агентов

Для большей ясности, общий веб-интерфейс разделен на несколько частей /областей. В верхней части окна отображается строка с общей информацией: имя пользователя, кнопка выхода, иконки, отображающие количество заблокированных заявок и позволяют прямой доступ к ним, ссылки для создания новой заявки на основе телефонного звонка или сообщения электронной почты. Также есть иконки для просмотра информации об очередях, статусах и эскалациях.

Под иконками находится навигационная строка. Она показывает меню, позволяющие переходить к разным модулям системы и позволяет выполнять некоторые глобальные действия. При нажатии на кнопку Панель Управления вы перейдете на страницу Панели Инструментов, которая по умолчанию отображается после входа в систему. Если нажать ссылку меню Заявки, откроется подменю, предлагающее различные действия, допустимые над заявками: создание новой заявки (на основе телефонного звонка или сообщения электронной почты), поиск конкретных заявок. Кнопка Отчеты также является частью меню и позволяет просматривать существующую статистическую информацию, создать новую а также производить импорт существующей статистики. Кнопка Клиенты позволяет перейти на страницу Управления Клиентами. Для администрирования системы нажмите кнопку Администрирование. На этой странице можно создавать аккаунты для других агентов, очереди и т.д. Для поиска заявок используйте кнопку Поиск.

Если установлены дополнительные приложения, такие как, например, FAQ или Опросы, кнопки в меню для перехода к ним также отображаются.

Красная полоса, которая находится под панелью навигации предназначена для отображения различных системных сообщений. Если войти в систему с правами OTRS-администратора, будет выведено сообщение о том, что в системе не следует работать с правами администратора.

Под названиями разделов, каждый из них находится в отдельном блоке есть несколько подразделов, содержащих ссылки и пояснения об их содержании и использовании.

Эти блоки содержат ссылки, они могут отображаться в одной или нескольких колонках, на определенные страницы с формами или инструментарием для выполнения определенных действий, например, добавление, изменение или удаление записей, проверку журнала сисмы, изменение конфигурации и т.д.

В нижней части экрана отображается "подвал" сайта (см. Рисунок ниже). Он содержит прямые ссылки на официальный сайт OTRS или ссылку для перехода на начало страницы. Также, может отображаться версия OTRS.

Обычно, строка иконок, меню и "подвал" - одинаковы на всех экранах веб-интерфейса.

Powered by OTRS 3.0 x CVS

About OTRS | Help

Top of page

Рисунок: Нижний колонтитул.

6. Панель Управления/Дайджест

Экран Дайджеста это главная страница системы, здесь вы можете увидеть информацию о заявках и других данных, связанных с работой над ними. Это должно быть начальной точкой повседневной работы агента, по умолчанию на ней представлен краткий обзор заявок, которые отложены, эскалированы, новых и открытых заявках и некоторая другая, полезная информация.

Важным достоинством экрана Дайджест является его полная настраиваемость. Это означает, что вы можете настроить каждую из частей экрана, скрывая или отображая их. Можно изменять набор и порядок отображения колонок в таблицах, менять в них порядок сортировки и пр. Каждый из элементов называется "Виджет". В системе имеется набор готовых виджетов, но модульная структура экрана готова к легкой интерации в нее пользовательских виджетов.

В левой колонке находится информация о заявках: напоминания, эскалации, новые или открытые. В каждой из категорий можно просматривать все заявки, к которым у вас есть доступ, сколько заявок заблокировано и сколько находится в "Моих Очередях". Есть, также другие виды виджетов и все они описаны ниже.

Виджеты левой стороны Дайджеста.

- Виджеты списков заявок

Виджеты этой категории имеют примерно одинаковый вид и смысл. Они отображают списки заявок в определенных, заданных, состояниях. Количество строк в таблицах, набор колонок и их порядок отображения настраивается с помощью кнопки "Параметры" (она появляется при наведении указателя мыши на правый верхний угол виджета). Эти виджеты поддерживают следующие фильтры:

- Мои заблокированные заявки

Список заблокированных заявок агента.

- Мои наблюдаемые заявки

Заявки агента, которые находятся в списке наблюдаемых. Параметр `Ticket::Watcher` должен быть включен для их отображения.

- Заявки, где я ответственный

Заявки, для которых агент является ответственным. Параметр `Ticket::Responsible` должен быть включен, чтобы этот фильтр работал.

- Заявки в моих очередях

Заявки в очередях, которые для агента заданы как "Мои очереди".

- Все Заявки

Все заявки доступные агенту.

Эти виджеты:

- Заявки с напоминанием

Отложенные заявки, у которых назначенная дата напоминания наступила.

- Эскалированные заявки

Эскалированные заявки

- Новые Заявки

Заявки имеющие состояние "Новая/New".

- Открытые Заявки / Ожидают ответа

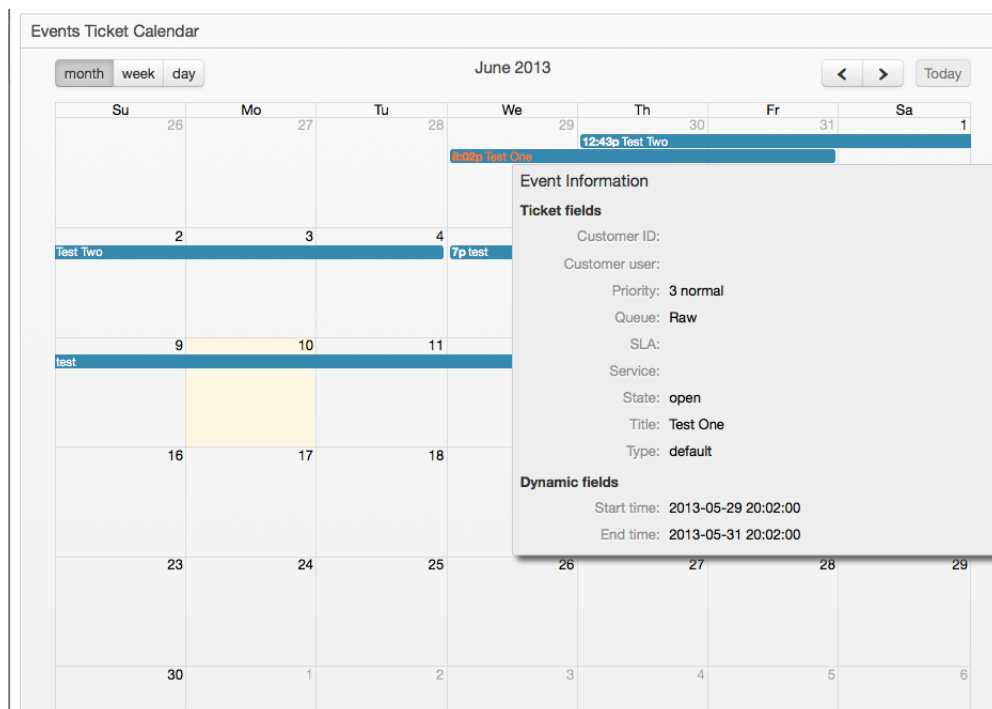
Заявки в состоянии "Открыта" и готовы для работы с ними.

- Календарь событий по заявкам

Событие календаря (для этого виджета) задается при создании новой заявки. Функция Календарь событий для заявок должна быть включена и требуется наличие двух новых полей на экране создания заявки, одно для времени начала и второе для времени окончания события. Эти значения определяют длительность события.

Виджет может отображаться в следующих представлениях: месяц, неделя и день, и агент может передвигаться по шкале времени, используя стрелки вправо и влево.

Как было отмечено ранее, только включения виджета - недостаточно. В систему должна быть добавлена пара динамических полей "Дата/Время" (используйте для этого панель настройки динамических полей в Панели Администратора) и указать их в настройках виджета в SysConfig. Оба поля должны быть указаны для отображения на экранах создания заявок, заполнены при создании заявки или при другом действии, позволяющем обновить эти значения (например, в пункте меню заявки - Свободные/Дополнительные поля) для описания временного интервала для события календаря (время начала и окончания). Можно задать отображение этих полей на экране просмотра заявки/TicketZoom для контроля, если сочтете это необходимым.



Дальнейшие настройки для этого виджета можно найти в подгруппе "Frontend::Agent::Dashboard::EventsTicketCalendar" в SysConfig:

- **Ширина Календаря**
 Задает ширину фрейма для Календаря в процентах. Стандартно - 95%.
- **DynamicFieldStartTime**
 Задает динамическое поля для времени начала.
- **DynamicFieldEndTime**
 Задает динамическое поля для времени окончания.
- **Очереди**
 Заявки только заданных в этом параметре очередей будут показаны в календаре.
- **DynamicFieldsForEvents**
 Задает динамические поля, отображаемые во всплывающем окне календаря. При наведении мыши на заявку.
- **TicketFieldsForEvents**
 Задает атрибуты заявки, отображаемые во всплывающем окне календаря. При наведении мыши на заявку.
- **Итоги по очередям**
 Этот виджет отображает итоги по очередям в виде матрицы, в которой строки показывают итоги в очереди, а столбцы состояние заявок, т.е каждая ячейка таблицы содержит количество заявок определенного состояния, принадлежащих конкретной очереди.

Виджет также показывает итоги по строкам и столбцам, где итоги по колонке это сумма заявок каждого состояния во всех очередях, тогда как итог по строке показывает сумму всех заявок в каждой очереди всех представленных в таблице состояний.

Перечень отображаемых в таблице состояний и очередей может быть изменен в Sysconfig.

Нажимая мышко по конкретному значению в ячейке таблицы можно быстро перейти к списку заявок, составивших это значение для более детального просмотра агентом.

С правой стороны расположен специальный виджет, позволяющий управлять отображением виджетов на экране Дайджеста. Это виджет "Параметры". Кликните по заголовку для его открытия и вы увидите список доступных виджетов, как показано на рисунке. Каждый виджет имеет область для пометки/checkbox, используйте их для пометки нужных для отображения виджетов (не помеченные виджеты не отображаются), после чего нажмите кнопку "Сохранить настройки" для сохранения изменений. Этот виджет навсегда зафиксирована на экране и его нельзя закрыть.

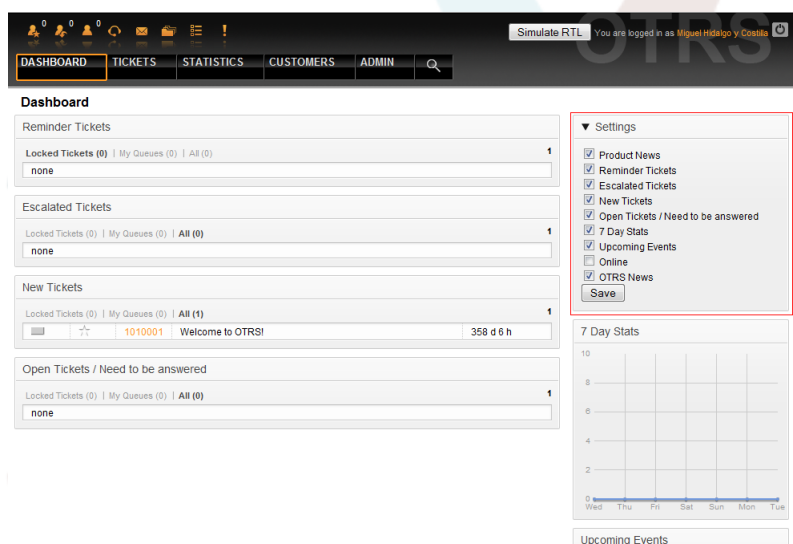


Рисунок: Настройки Панели Управления.

Виджеты правой стороны Дайджеста.

- Статистика за неделю (7 дней)

Он показывает график активности по заявкам за прошедшие 7 дней и отображает 2 линии. Одна, синего цвета показывает количество вновь созданных заявок в день и вторя, обычно оранжевая, количество закрытых заявок в день.

- Предстоящие события

Здесь перечисляются заявки подлежащие эскалации или уже эскалированные. Информация этого виджета очень полезна, чтобы решить, какие заявки требуют вашего внимания или просто контролировать что должно произойти.

- Новости OTRS

Полный список событий в компании OTRS, а также информация о новых релизах продуктов или заплатках.

- Онлайн

Здесь отображается сводка об агентах вошедших в систему, он также включает и секцию о клиентах on-line, обычно, эти виджеты скрыты, их можно включить используя виджет "Параметры" описанный выше.

7. Что такое очередь?

На физическом уровне для почтового ящика создается специальный файл (иногда его еще называют почтовый спул), в который помещаются и хранятся все сообщения. Новые сообщения электронной почты просто добавляются в конец этого файла. Почтовая программа, которая используется для чтения и написания писем (такие программы еще называются пользовательскими почтовыми агентами - MUA (Mail User Agent) читает этот файл и отображает его содержимое.

Очереди OTRS можно сравнить с Inbox-файлом, в котором хранятся входящие сообщения электронной почты, так как в них тоже может храниться множество сообщений. Очереди имеют много функций, схожих с теми, которые присущи Inbox-файлу. Как агенты так и пользователи, все должны помнить о том, в какой очереди содержится заявка. Агенты могут открывать, редактировать заявки, размещенные в очереди, а также, перемещать их из одной очереди в другую. Но зачем перемещать заявки?

Чтобы объяснить это с более практической точки зрения давайте вспомним пример с компанией Максима, которая была описана в разделе пример системы обработки заявок. Максим установил OTRS чтобы избавиться от существующего хаоса в системе поддержки и предоставить возможность своей команде лучше управлять службой поддержки клиентов, покупающих видеомагнитофоны.

На данный момент достаточно одной очереди, чтобы хранить в ней все запросы клиентов. Но через некоторое время Максим решает начать продажи DVD-магнитофонов. Теперь у клиентов есть вопросы касающиеся не только видеомагнитофонов, но и вопросы относительно нового продукта. С течением времени в очередь, ранее созданную Максимом, начинает приходить все больше и больше писем и управляться с ними становится все труднее и труднее.

Однажды Максим принимает решение немного оптимизировать структуру системы поддержки и добавляет две новых очереди. Таким образом в системе используется уже три очереди. Новые заявки, только поступившие в систему хранятся в очереди "raw". Две новые очереди: одна для хранения заявок касающихся видеомагнитофонов и другая - для dvd-магнитофонов.

Максим просит Сандру просматривать очередь "raw" и сортировать (распределять) емейлы либо в очередь "видеомагнитофоны" или в "dvd-магнитофоны", в зависимости от того, что запросил клиент. Джон имеет доступ только к очереди "видеомагнитофоны", а Джо может отвечать на заявки из очереди "dvd-магнитофоны". Максим имеет возможность редактировать заявки, находящиеся во всех очередях.

OTRS поддерживает управление доступом для пользователей, групп и ролей, таким образом можно очень легко настроить очереди, которые доступны только для некоторых учетных записей. Чтобы все запросы приходили в различные очереди, Максим может использовать правила фильтрации. Иначе, если используются два различных email-адреса, Сандра будет сортировать и направлять в другие очереди только те заявки, которые не были отсортированы автоматически.

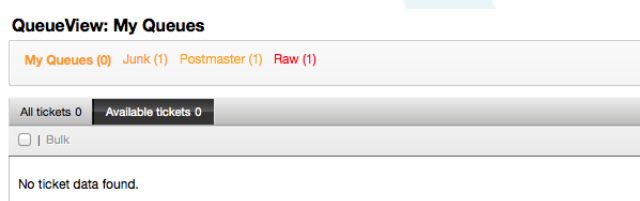
Сортировка входящих сообщений в различные очереди позволяет сохранять систему поддержки в структурированном и аккуратном виде. Поскольку ваши агенты находятся в разных группах и имеют различные права для доступа к очередям, система может быть оптимизирована еще больше. Очереди могут быть использованы

для определения структуры всей компании а также для определения рабочего потока процессов. Максим может к примеру создать еще одну очередь и назвать ее "продажи", которая будет содержать подочереды "запросы", "предложения", "заказы", "оплаты" и т.д. Такой подход поможет Максиму структурировать и оптимизировать процесс работы с транзакциями.

Чем лучше структурирована система поддержки, тем меньше времени требуется для различных задач. Это приводит к уменьшению рабочего времени, соответственно и к уменьшению денежных затрат. Очереди могут помочь оптимизировать бизнес-процессы вашей компании.

8. Что такое просмотр очередей?

Просмотр очередей дает возможность видеть все очереди, в которых есть заявки, и в которых агент имеет права RW.

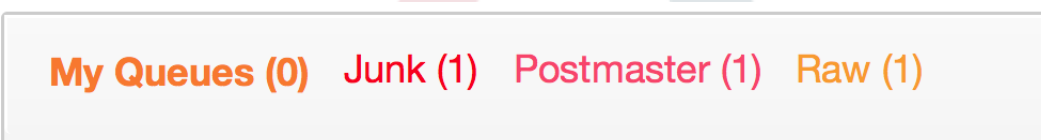


Обзор очередей предлагает различные возможности для повседневной работы в OTRS. Первая из них это "Мои очереди". В личных настройках агента или при администрировании агентов может быть задан перечень очередей, которые предназначены агенту для работы. Все заявки этих очередей будут появляться по умолчанию при переходе к Заявки -> Обзор очередей в меню.

Другой возможностью предлагаемой в Обзоре очередей является навигация по отдельным очередям и подочередям через их раскрытие/drill down с которыми надо работать.

В обоих типах просмотра пользователь имеет дополнительную возможность смотреть либо только разблокированные заявки (фильтр по умолчанию), либо можно выбрать показ всех доступных заявок. Заявки должны быть в одном из доступных для просмотра состояний. По умолчанию это 'open, new, pending reminder, pending auto'.

Для помощи пользователю имеются визуальные оповещения.



Визуальные Оповещения

- Highlight Age 1: Устанавливает возраст в минутах (первый уровень) для подсветки очередей, содержащих непросмотренные заявки. Смотрите выше для очереди "Raw".
- Highlight Age 2: Устанавливает возраст в минутах (второй уровень) для подсветки очередей, содержащих непросмотренные заявки. Смотрите выше для очереди "Postmaster".

- **Blink:** Включает механизм мерцания для очереди содержащей наиболее старые заявки. Поддерживается не всеми браузерами. В данном случае они выглядят красными, как это видно для очереди "Junk".
- **Bold:** Текущая очередь будет выделена жирным шрифтом, как это видно для "Мои очереди/My Queues".

9. Пользовательские настройки

Пользователи OTRS, такие как: клиенты, агенты и OTRS-администраторы могут настроить свои учетные записи в соответствии с их потребностями. Агенты могут открыть страницу с настройками аккаунта кликнув ссылку с именем пользователя, которая находится в верхнем правом углу веб-интерфейса (см. Рисунок ниже), а клиенты должны нажать ссылку "Предпочтения" (см. Рисунок ниже).

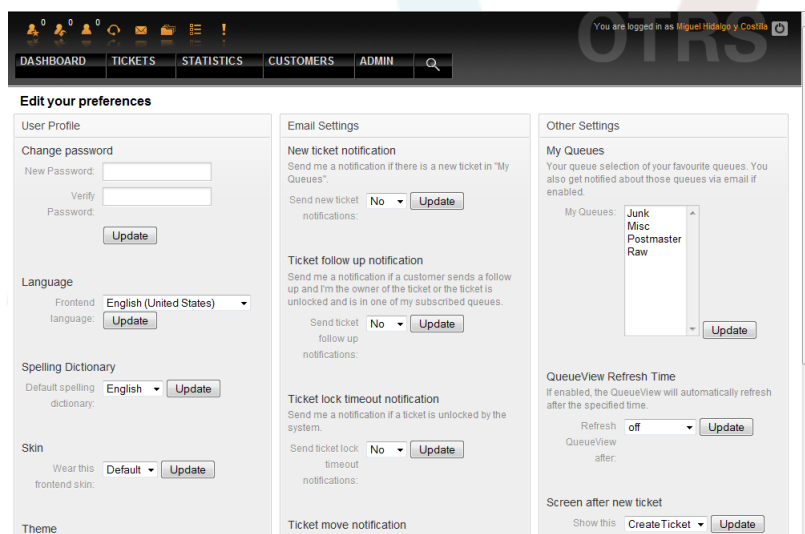


Рисунок: Личные предпочтения Агентов.

Агент может менять 3 различных категории настроек: профиль пользователя, настройки электронной почты и другие параметры. Возможности по умолчанию:

Профиль пользователя

- Изменение текущего пароля.
- Выбор языка интерфейса.
- Переключение между темами интерфейса.
- Изменение темы веб-интерфейса.
- Активация и настройка функции "время вне офиса".

Настройки электронной почты

- Выберите события, при наступлении которых, система должна отсылать уведомления агентам на их емейлы.

Другие настройки

- В меню "Мои Очереди" выберите те очереди, которые хотите отслеживать более внимательно.

- Установите время обновления страницы для шаблона отображения очередей.
- Выберите страницу, которая должна отображаться после создания заявки.

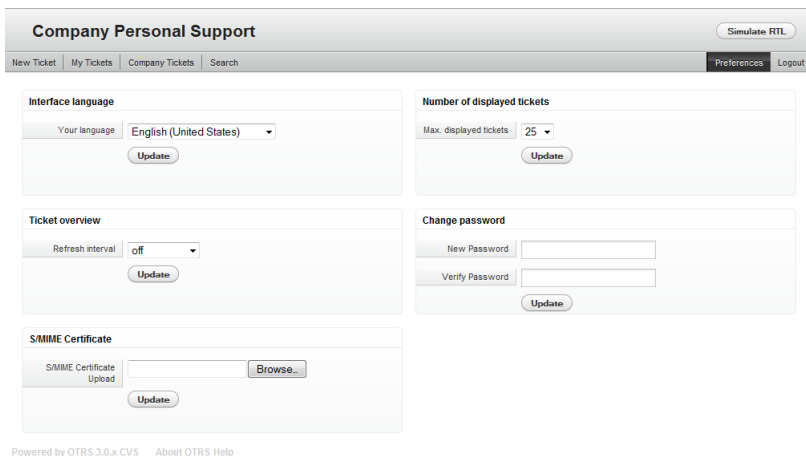


Рисунок: Личные предпочтения Клиента.

Клиент может выбрать язык веб-интерфейса системы, изменять время обновления страницы в шаблоне просмотра очередей, выбирать максимальное количество отображаемых заявок. Кроме этого также можно установить новый пароль.

Глава 4. Администрирование

1. Панель управления OTRS

1.1. Основы

Администраторы используют страницу администрирования веб-интерфейса OTRS для настройки системы - добавления агентов, клиентов и очередей, заявок и настройки почты, установки дополнительных пакетов, таких как FAQ и ITSM и многое другое.

Агенты, входящие в группу *admin* могут получить доступ к админке, нажав на кнопку *Администрирование* на панели навигации (см. нижеприведенный Рисунок). Остальные агенты не будут видеть эту ссылку.

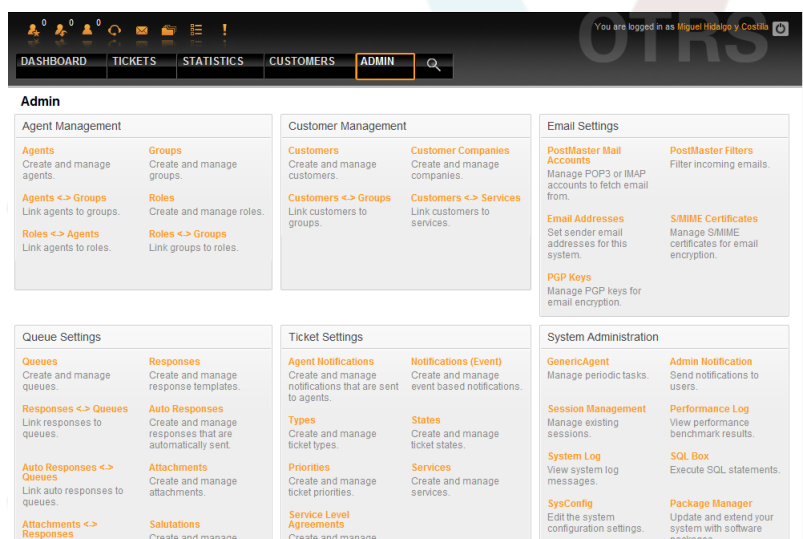


Рисунок: Страница OTRS-администратора.

1.2. Агенты, Группы и Роли

1.2.1. Агенты

Нажав на ссылку *Агенты* вы получите доступ к странице управления агентами OTRS (см. нижеприведенный Рисунок). Администраторы могут добавлять, изменять или деактивировать аккаунты агентов. Администраторы также могут управлять личными настройками агентов, например изменять язык их веб-интерфейса, настройки уведомлений.

Примечание

Учетная запись (аккаунт) агента может быть деактивирована, но не удалена. Деактивация осуществляется путем установки параметра Действительный в значение *не действительный* или *не действительный-временно*.

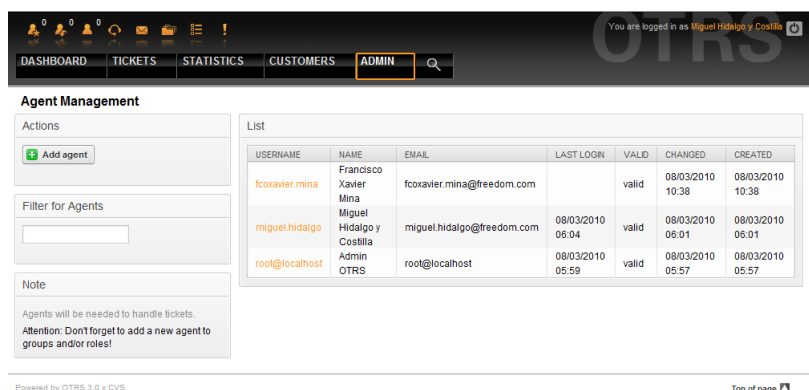


Рисунок: Управление Агентами.

Чтобы создать агента, нажмите на кнопку "Добавить агента", введите все необходимые данные и нажмите кнопку Отправить, которая находится в нижней части экрана, как это показано на нижеприведенном Рисунке.

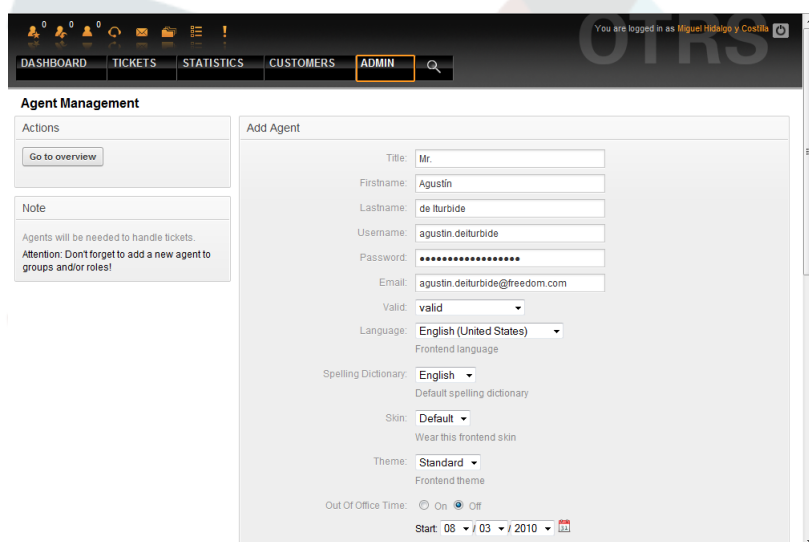


Рисунок: Добавление нового агента.

После того, как новая учетная запись агента создана, его нужно сделать членом одной или нескольких групп или ролей. За дополнительной информацией о группах и ролях обращайтесь к разделу Группы и Роли данной главы.

1.2.2. Группы

Каждая учетная запись агента должна принадлежать по крайней мере к одной группе или роли. После установки системы есть три предопределенных группы, которые приведены в Таблице 5-1.

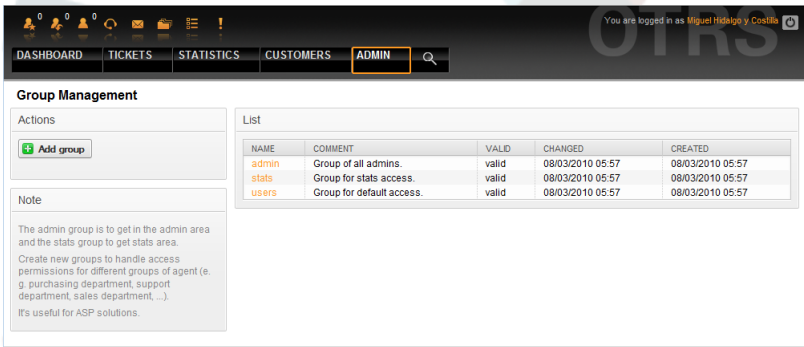
Таблица 4.1. Предопределенные группы, доступны для новой (только что) установленной системы OTRS

Группа	Описание
admin	Допускается для выполнения административных задач в системе.
отчеты	Компетентные для генерации статистики и доступа к модулю статистики OTRS.
users	Агенты должны входить в эту группу с правами для чтения и записи. Тогда они смогут получить доступ ко всем функциям системы обработки заявок.

Примечание

В новой установке OTRS, группа *users* изначально пуста. Агент 'root@localhost' по умолчанию является членом групп admin и stats.

Чтобы перейти на страницу управления группами, нажмите в панели администрирования ссылку *Группы* (см. нижеприведенный Рисунок).



Group Management

Actions

[Add group](#)

Note

The admin group is to get in the admin area and the stats group to get stats area.
 Create new groups to handle access permissions for different groups of agent (e.g. purchasing department, support department, sales department, ...).
 It's useful for ASP solutions.

NAME	COMMENT	VALID	CHANGED	CREATED
admin	Group of all admins	valid	08/03/2010 05:57	08/03/2010 05:57
stats	Group for stats access	valid	08/03/2010 05:57	08/03/2010 05:57
users	Group for default access	valid	08/03/2010 05:57	08/03/2010 05:57

Powered by OTRS 3.0.x CVS Top of page

Рисунок: Управление группами.

Примечание

Как в случае с агентами, группа в OTRS может быть только деактивирована, но не удалена. Деактивация осуществляется путем установки флага Действительный в значение *не действительный* или *не действительный-временно*.

Чтобы добавить агента в группу, или изменить принадлежность агентов к группе, используйте ссылку на Панели Управления *Агенты <-> Группы* (см. нижеприведенный Рисунок).

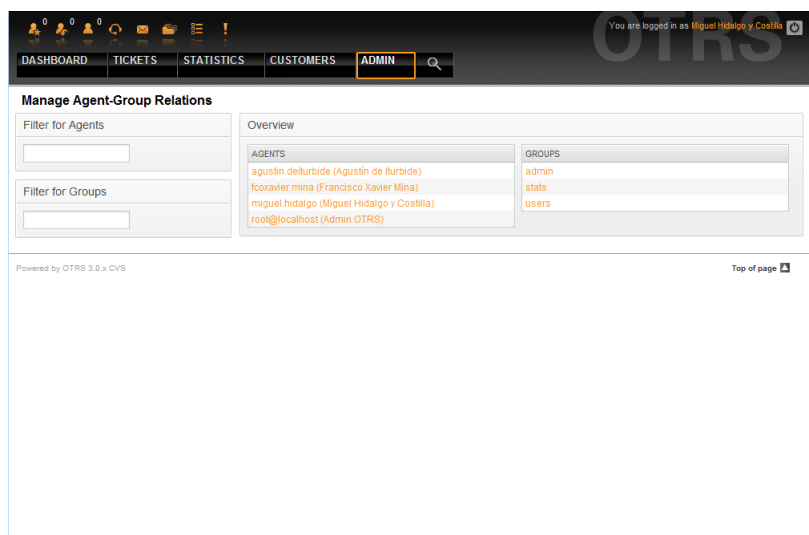


Рисунок: Управление группами.

На этой странице приводится обзор всех групп и агентов. Для поиска той или иной сущности можно использовать фильтры. Если вы хотите изменить группы, в которые входит определенный агент, просто нажмите на его имя (см. нижеприведенный Рисунок). Для изменения агентов, связанных с группой просто нажмите на название нужной группы (см. нижеприведенный Рисунок).

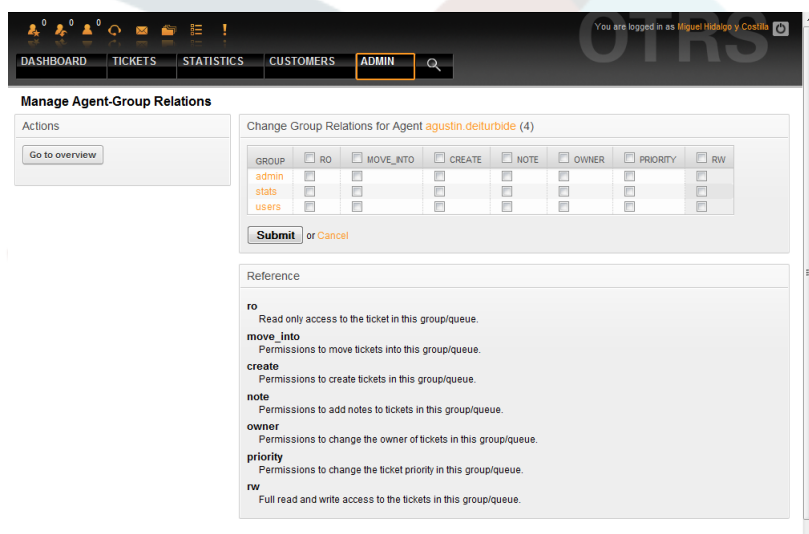


Рисунок: Изменение группы, к которой принадлежит агент.

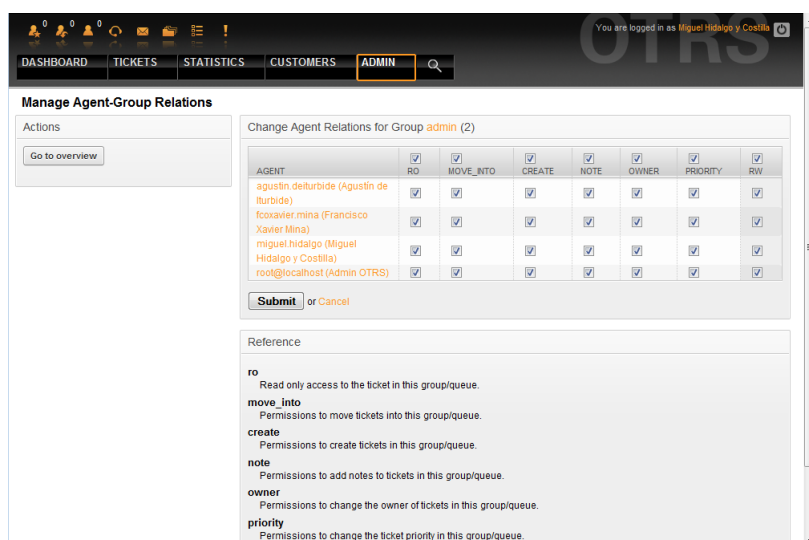


Рисунок: Изменение агентов, входящих в определенную группу.

Каждая группа имеет набор прав, связанный с ней и, каждый член этой группы (агент) может иметь некоторую собственную комбинацию прав. Список разрешений/прав приведен в Таблице 5-2.

Таблица 4.2. Права, связанные с OTRS-группами

Права	Описание
ro	Права "только для чтения" заявок, записей и очередей этой группы.
move into	Права для перемещения заявок или записей между очередями или для области, принадлежащей группе.
create	Права для создания заявок или записей в очередях или области группы.
owner	Права для изменения владельца заявок или записей в очередях принадлежащих этой группе.
priority	Права для изменения приоритета заявок или записей в очереди или области, принадлежащей этой группе.
rw	Полный доступ к заявкам или записям в очереди или области, принадлежащей к этой группе.

Примечание

По умолчанию, Обзор Очередей/QueueView отображает только заявки в очередях, в которых агент имеет права *rw*, т.е доступ к заявкам, с которыми он должен работать. Если вы хотите изменить этот порядок, вы можете установить параметр `Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets` в значение *Да*.

Не все доступные права отображаются по умолчанию. Эти дополнительные права могут быть добавлены.

Таблица 4.3. Дополнительные группы прав

Права	Описание
отчеты	Предоставляет доступ к странице отчетов.
bounce	The right to bounce an email message (with bounce button in ticketZoom).
compose	Право составления ответа на заявку.
клиент	Право изменить клиента для этой заявки
forward	The right to forward a messages (with the forward button).
в ожидании	Право перевести заявку в ожидание.
телефон	Право добавить содержание звонка клиента к заявке. В виде сообщения/заметки.
ответственный	Право изменить ответственного за заявку.

Примечание

Эти права могут быть добавлены изменением параметра System::Permission

1.2.3. Роли

Роли являются мощным средством для простого и быстрого управления правами доступа множества агентов. Они особенно полезны для больших и сложных систем поддержки, с большим количеством агентов, групп и очередей. В приведенном ниже примере объясняется как и когда они могут использоваться.

Давайте представим, что у нас есть система, в которой зарегистрировано 100 агентов, 90 из них имеют доступ только к очереди "поддержка", в которой хранятся все запросы о поддержке. Очередь "поддержка" содержит несколько под очередей. Остальные 10 агентов имеют права доступа ко всем очередям, существующим в системе. Эти 10 агентов работают с заявками, просматривают "raw"-очередь и перемещают спам-сообщения в очередь "junk".

В один день компания открывает новый отдел для продажи товаров. Запрос/принятие заказа, подтверждение, счета и т.д. должны быть обработаны, и некоторые из агентов компании должны делать это используя OTRS. Разным агентам нужно предоставить доступ к очередям, которые планируется создать.

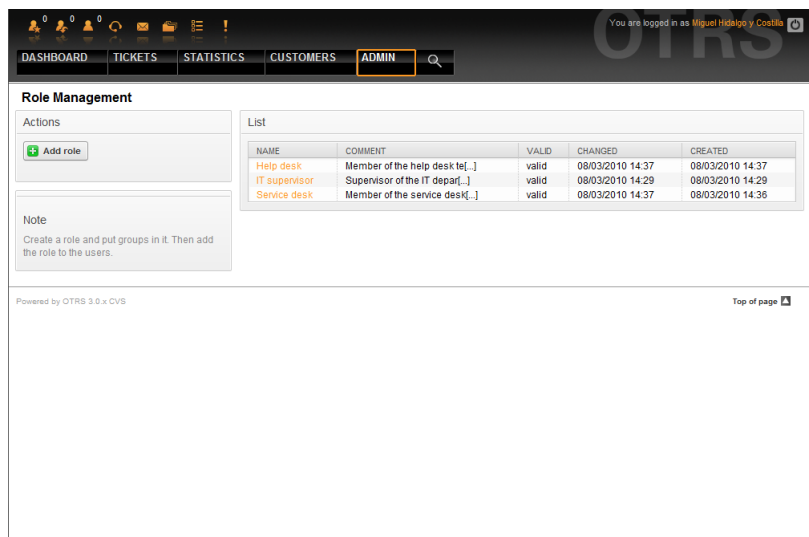
Поскольку изменения вручную прав доступа для разных агентов займет много времени, то для определения различных уровней доступа можно создать роли. Затем, агенты могут быть добавлены к одной или нескольким ролям, таким образом их права будут изменены автоматически. Если будет создана новая учетная запись агента, ее также можно с легкостью добавить к одной или нескольким ролям.

Примечание

Роли действительно полезны когда мы имеем дело со сложно структурированными компаниями и управляем сложными системами OTRS. Тем не менее следует быть аккуратным с этим механизмом. Перемешивание назначения агентов в группы назначением агентам ролей может привести к переусложненной схеме, управлять которой, поддерживать актуальной и просто понимать будет затруднительно. Если вы желаете использовать только роли и

отключить настройку Агенты <-> Группы в пангели администратора, можно сделать это в параметре Frontend::Module###AdminUserGroup в SysConfig. Будьте аккуратны, во избежание удаления уже сделанных назначений агентов в группы!

Для управления ролями перейдите по ссылке *Роли* в Панели Администрирования (см. нижеприведенный Рисунок).



The screenshot shows the OTRS Role Management interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The ADMIN tab is selected. Below the navigation bar, the page title is "Role Management". On the left, there is an "Actions" section with a green "Add role" button and a "Note" section with instructions: "Create a role and put groups in it. Then add the role to the users." On the right, there is a "List" section containing a table with the following data:

NAME	COMMENT	VALID	CHANGED	CREATED
Help desk	Member of the help desk te[...]	valid	08/03/2010 14:37	08/03/2010 14:37
IT supervisor	Supervisor of the IT depart[...]	valid	08/03/2010 14:29	08/03/2010 14:29
Service desk	Member of the service desk[...]	valid	08/03/2010 14:37	08/03/2010 14:36

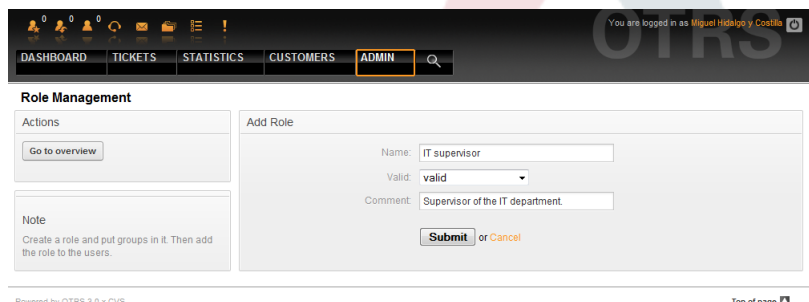
At the bottom of the page, it says "Powered by OTRS 3.0.x CVS" and "Top of page".

Рисунок: Управление Ролями.

Примечание

Как в случае с агентом и группами, однажды созданные роли могут быть деактивированы, но не удалены. Чтобы отключить, установите опцию Действительный в значение *не действительный* или *не действительный-временно*.

Приведен обзор всех ролей в системе. Нажмите на имя роли, чтобы отредактировать ее настройки. По умолчанию, в новой, только что установленной системе OTRS нету predefined ролей. Чтобы создать новую, нажмите кнопку "Добавить роль", введите все необходимые данные и нажмите кнопку Отправить (см. нижеприведенный Рисунок).



The screenshot shows the OTRS Role Management interface with the "Add Role" form. The navigation bar is the same as in the previous screenshot. The "Actions" section now has a "Go to overview" button. The "Add Role" form has the following fields:

- Name: IT supervisor
- Valid: valid (dropdown menu)
- Comment: Supervisor of the IT department.

At the bottom of the form, there are "Submit" and "Cancel" buttons. At the bottom of the page, it says "Powered by OTRS 3.0.x CVS" and "Top of page".

Рисунок: Добавление новой роли.

Чтобы получить список всех ролей и агентов в системе, перейдите по ссылке Роли <-> Агенты в Панели Администрирования. Также, для поиска конкретного элемента можно использовать фильтры. Если необходимо изменить роли связанные с определенным агентом, просто кликните на имя агента (см. нижеприведенный Рисунок). Чтобы изменить агентов, связанных с ролью, нажмите на роль, которую нужно отредактировать (см. нижеприведенный Рисунок).

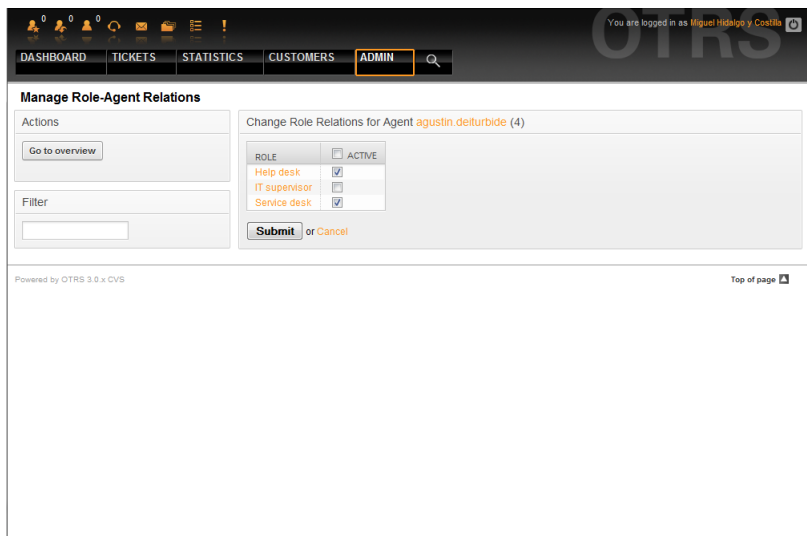


Figure: Изменение Ролей, связанных с Агентом.

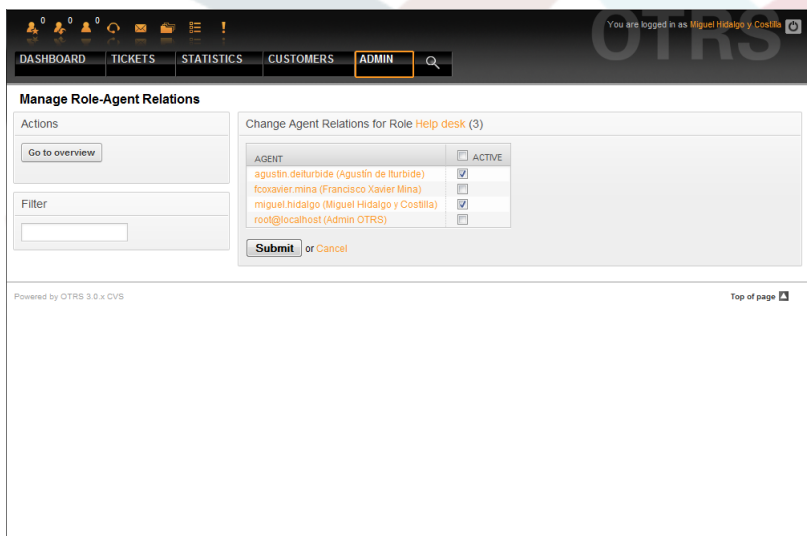


Figure: Изменение Агентов, связанных с конкретной Ролью.

Чтобы получить список всех ролей и групп системы, перейдите по ссылке Роли <-> Группы на Панели Управления. На экране появится страница, приведенная ниже на Рисунке. Для поиска конкретного элемента также можно использовать фильтры.

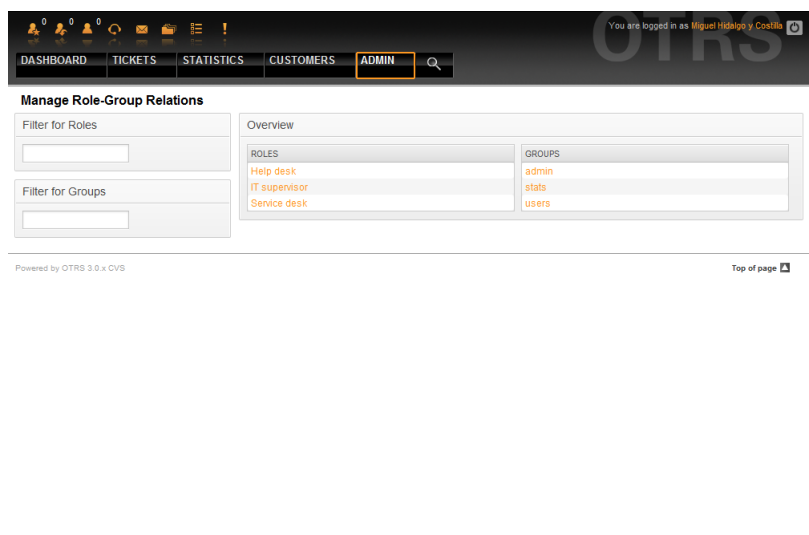


Figure: Управление отношениями Роли-Группы.

Чтобы определить различные права доступа для роли, нажмите на имя роли или группы (см. ниже Рисунок 5.13 и 5.14, соответственно).

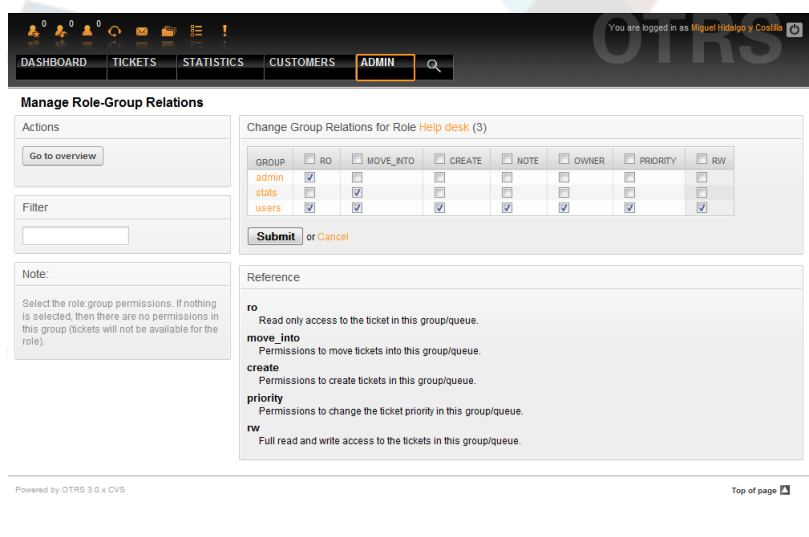


Рисунок: Изменение связи между Группой и Ролью.

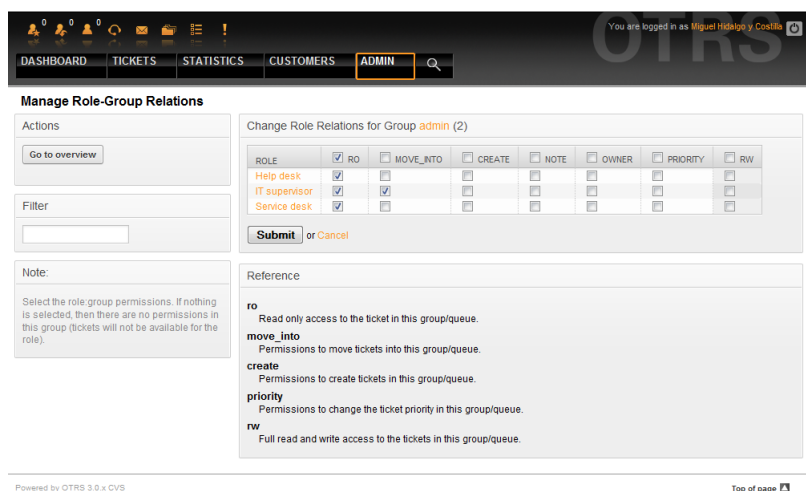


Рисунок: Изменение отношения Роли к Группе.

1.3. Клиенты и Группы Клиентов

1.3.1. Клиенты

OTRS поддерживает различные типы пользователей. Используя ссылку "Клиенты" (с помощью навигационной панели, или на странице Панели Администратора), можно управлять учетными записями клиентов (см. нижеприведенный Рисунок), которые могут войти в систему используя Пользовательский веб-интерфейс (customer.pl). Используя этот интерфейс клиенты могут создавать и просматривать состояние своих заявок и изменений в них. Важно помнить что клиенты необходимы в системе для истории заявок.

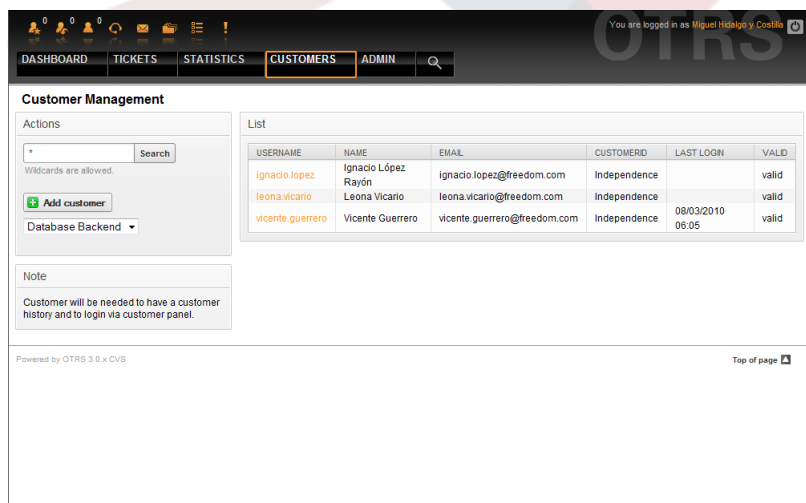


Рисунок: Управление клиентами.

Нажав на имя клиента, по зарегистрированным клиентам можно производить поиск, редактировать их данные. Также есть возможность изменить клиентский backend, за более подробной информацией обращайтесь к главе внешние хранилища данных.

Чтобы создать новую учетную запись клиента, нажмите кнопку "Добавить клиента" (см. нижеприведенный Рисунок). Некоторые поля являются обязательными, т.е.

они должны содержать данные, поэтому если вы оставите их пустыми, они будут выделены красным цветом.

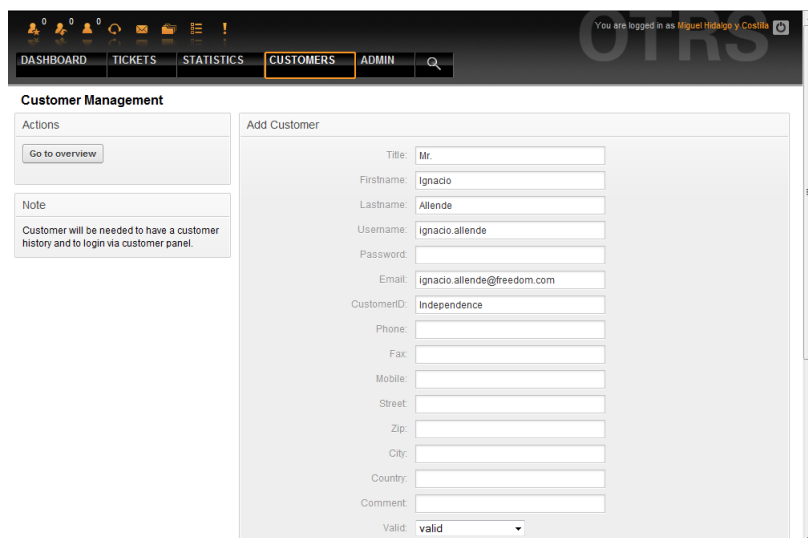


Рисунок: Добавление клиента.

Клиент может получить доступ к системе путем ввода логина и пароля. CustomerID нужен для идентификации клиента и его заявок. Поскольку адрес электронной почты является уникальным, его можно использовать в качестве уникального идентификатора (ID).

Примечание

Как и в случае с агентами, группами и ролями, система не позволяет удалять клиентов, а только деактивировать их, установив опцию Действительный в значение *недействительный* или *недействительный-временно*.

1.3.2. Группы Клиентов

Клиенты также могут быть добавлены в группу, что может быть полезно, если вы хотите объединить нескольких клиентов той же компании для доступа к одной или нескольким очередям. Сначала создайте группу, в которую будут входить клиенты, для этого перейдите по ссылке Модуль Управления Группами. Затем добавьте очереди и выберите новую группу для этих очередей.

Следующим шагом будет включение поддержки групп клиентов. Это можно сделать используя параметр CustomerGroupSupport в SysConfig/Управлении Конфигурациями. Используя параметр CustomerGroupAlwaysGroups, можно указать группы по умолчанию, к которым будет принадлежать новый клиент, таким образом каждая новая учетная запись будет добавляться в эти группы.

Перейдя по ссылке "Клиенты <-> Группы" вы можете управлять отношениями конкретного пользователя к разным группам (см. нижеприведенный Рисунок).

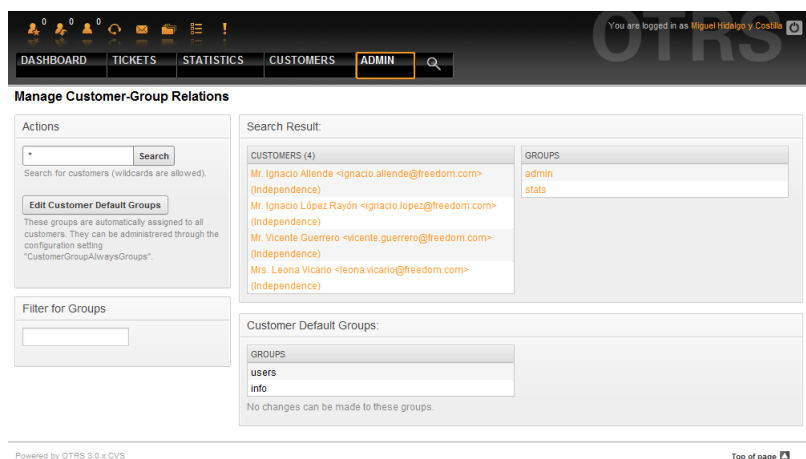


Рисунок: Управление отношениями Клиент-Группа.

Для определения различных групп, в которые должен входить клиент, и наоборот, нажмите на соответствующее имя пользователя или группы (см. ниже Рисунок 5.16 и 5.17, соответственно).

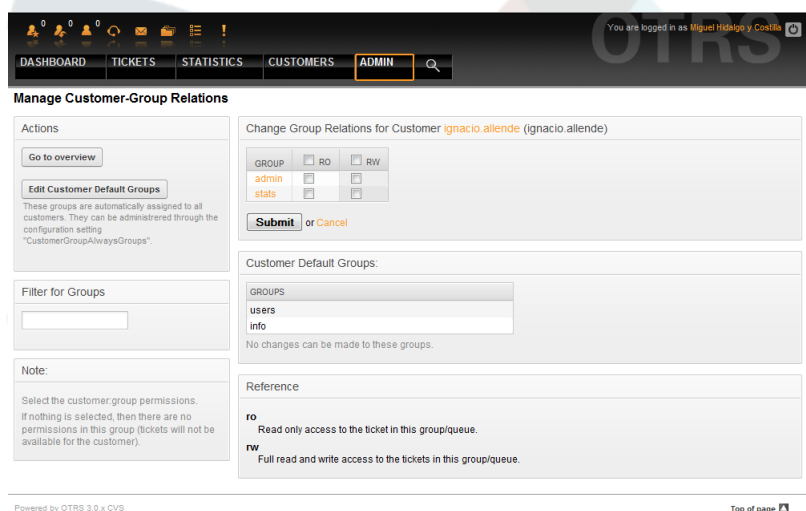
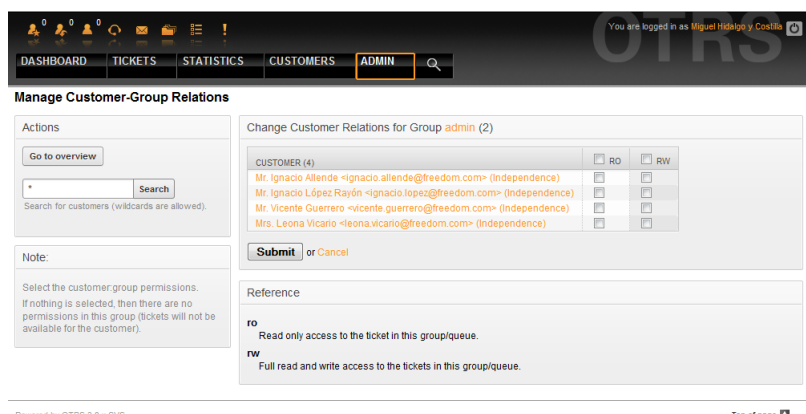


Рисунок: Изменение связи между Группой и Клиентом.



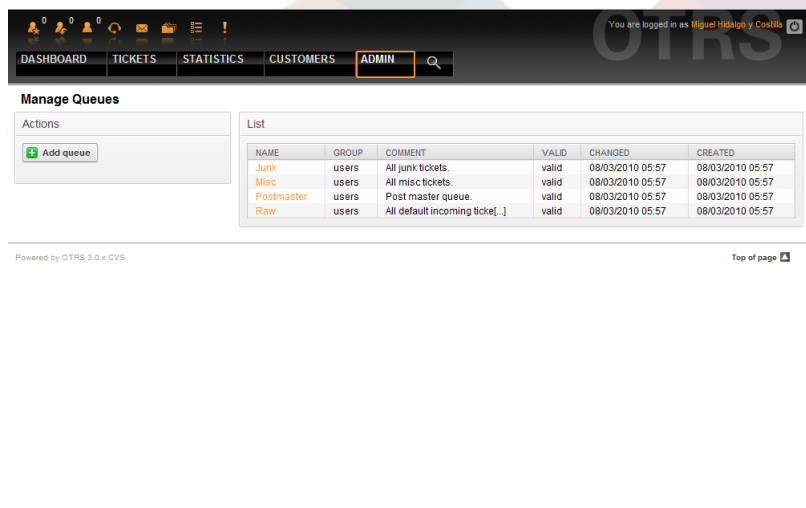
Powered by OTRS 3.0.x CVS

Top of page

Рисунок: Изменение связи между Клиентом и Группой.

1.4. Очереди

Используя ссылку "Очереди", находящуюся в Панели Администрирования, можете управлять очередями системы (см. нижеприведенный Рисунок). В новой, только установленной системе OTRS существует 4 predetermined очереди: Junk, Misc, Postmaster и Raw. Если не используются ранее определенные фильтры и правила, все входящие сообщения сохраняются в очереди "Raw". Очередь "Junk" используется для хранения спам-сообщений.

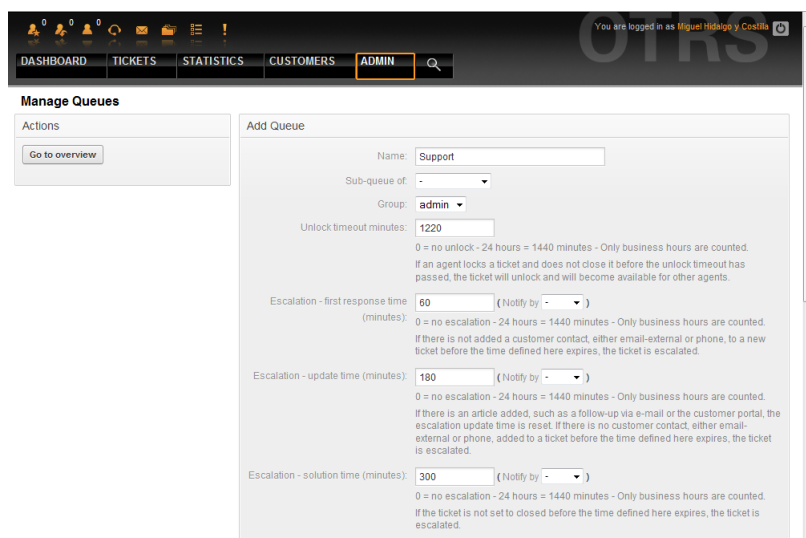


Powered by OTRS 3.0.x CVS

Top of page

Рисунок: Управление очередями.

Здесь можно добавлять и корректировать очереди (см. нижеприведенный Рисунок). Можно определить группу, которая должна использовать очередь. Также очередь можно установить как под-очередь уже существующей очереди.



Manage Queues

Actions

Add Queue

Name:

Sub-queue of:

Group:

Unlock timeout minutes:
0 = no unlock - 24 hours = 1440 minutes - Only business hours are counted.
 If an agent locks a ticket and does not close it before the unlock timeout has passed, the ticket will unlock and will become available for other agents.

Escalation - first response time (minutes): (Notify by:
0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.
 If there is not added a customer contact, either email-external or phone, to a new ticket before the time defined here expires, the ticket is escalated.

Escalation - update time (minutes): (Notify by:
0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.
 If there is an article added, such as a follow-up via e-mail or the customer portal, the escalation update time is reset. If there is no customer contact, either email-external or phone, added to a ticket before the time defined here expires, the ticket is escalated.

Escalation - solution time (minutes): (Notify by:
0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.
 If the ticket is not set to closed before the time defined here expires, the ticket is escalated.

Рисунок: Добавление новой очереди.

Для очереди можно определить время разблокировки - если агент заблокировал заявку и не закрыл ее до истечения времени блокировки, заявка будет автоматически разблокирована и доступна для других агентов.

Есть три типа настроек эскалации времени, которые могут быть связаны на уровне очередей:

Эскалация - Время Первого Отклика

- После создания заявки, если истекает оговоренное время без какой-либо связи с клиентом как по электронной почте так и по телефону, - заявка будет эскалирована.

Эскалация - Время обновления

- Если какой-нибудь клиент прислал ответ на заявку по электронной почте или изменил ее через пользовательский портал, время эскалации будет обнулено. Если до истечения времени ожидания контакта с клиентом не было, заявка эскалируется.

Эскалация - Время решения

- Если до истечения оговоренного времени заявка не закрыта - она эскалируется.

Используя опцию 'Ticket lock after a follow-up', можно определить будет ли заявка установлена в значение 'заблокирована' для старого владельца, если она была закрыта а потом открыта заново. Такое поведение гарантирует, что последующая заявка будет обрабатываться агентом, который работал с этой заявкой ранее.

Параметром системный адрес определяется адрес электронной почты, который будет использоваться для исходящих заявок этой очереди. Также, для ответов по электронной почте, есть возможность ассоциировать/связать очереди с приветствиями и подписями. За более подробной информацией перейдите по ссылке адреса электронной почты, приветствия и подписи.

Примечание

Как в случае с агентами, группами и клиентами, очереди не могут быть удалены, а только отключены, путем установки опции "Действительная" в значение *недействительная* или *недействительная-временно*.

1.5. Приветствия, подписи, вложения и шаблоны

1.5.1. Приветствия

Приветствие это текстовый шаблон для ответа. Приветствия могут быть связаны с одной или многими очередями, как это описано в в разделе очереди. Приветствия используются только в том случае, если на заявку из очереди, связанной с этим приветствием создается ответ. Для управления различными приветствиями в системе воспользуйтесь ссылкой "Приветствия" в панели администрирования (см. нижеприведенный Рисунок).

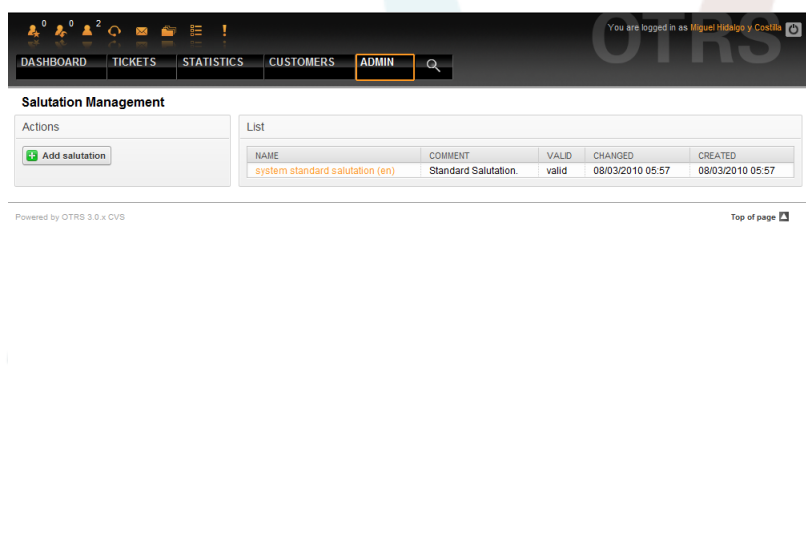


Рисунок: Управление приветствиями.

После установки в системе есть одно предопределенное приветствие: "system standard salutation (en)".

Чтобы создать новое приветствие, нажмите на кнопку "Добавить приветствие", введите необходимые данные и нажмите кнопку "Отправить" (см. Рисунок ниже)

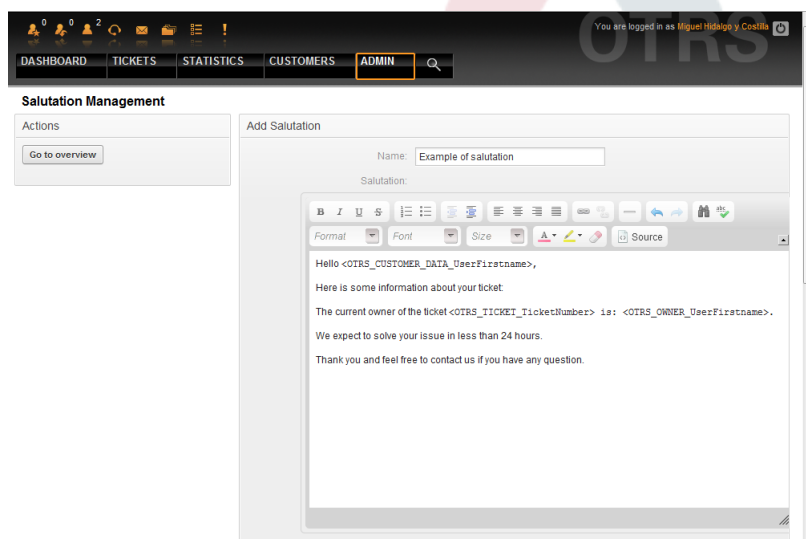


Рисунок: Добавление нового приветствия.

В приветствиях можно использовать переменные. Когда вы отвечаете на заявку, имена переменных будут заменены на их значения.

Различные переменные, которые можно использовать в ответах приведены в нижней части экрана приветствия. Если вы используете, например, переменную <OTRS_LAST_NAME> то в ответ будет включена фамилия отправителя заявки.

Примечание

Как и в случае с другими сущностями системы OTRS, приветствия не можно удалять, только деактивировать их, установив параметр Действительный в значение *не действительный* или *не действительный-временно*.

1.5.2. Подписи

Другим текстовым модулем для шаблонов являются подписи. Подписи могут быть связаны с очередями, как это описано в разделе очереди. Как только подпись связана с очередью она будет включена в текст ответа каждой заявки, находящейся в этой очереди. Перейдя по ссылке "Подписи" в Панели Администрирования, можно упарвлять подписями системы (см. Рисунок ниже).

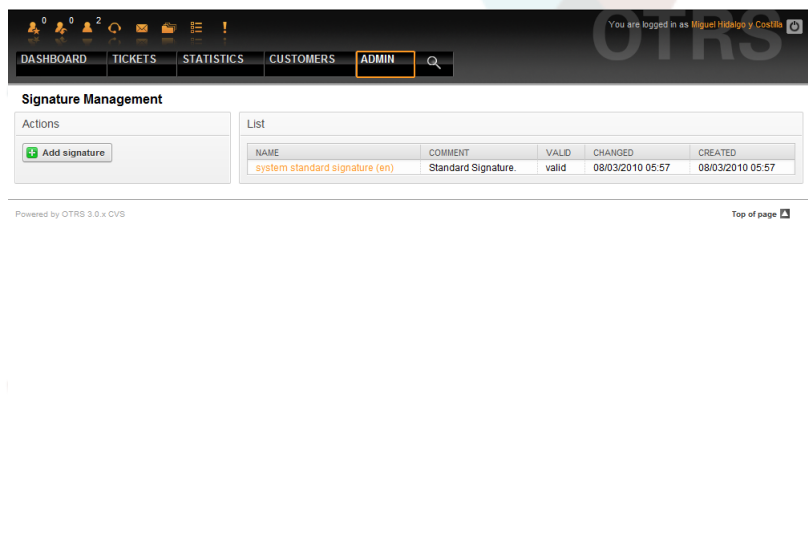


Рисунок: Управление подписями.

В новой, только установленной системе OTRS есть одна предустановленная подпись: "system standard signature (en)".

Чтобы создать новую подпись, нажмите кнопку "Добавить подпись", введите все необходимы данные и нажмите кнопку "Отправить" (см. нижеприведенный Рисунок).

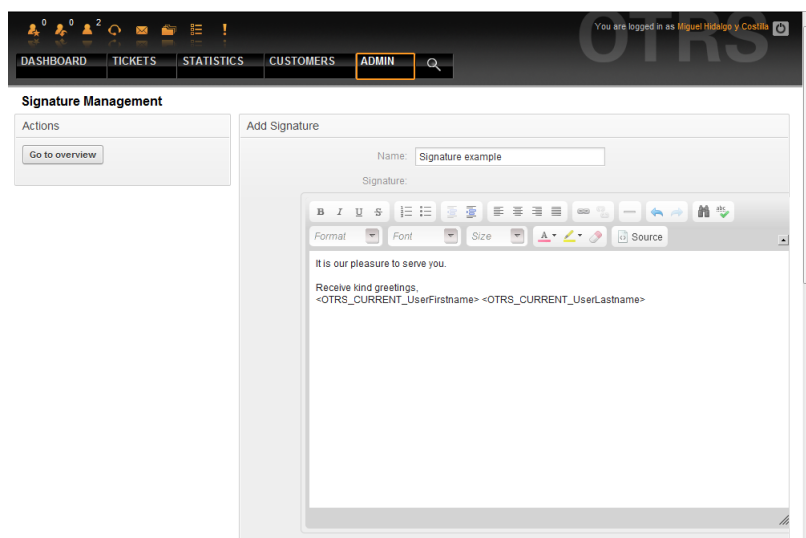


Рисунок: Добавление новой подписи.

Как и приветствия, подписи также могут содержать динамический контент, например, имя и фамилию агента, который отвечает на заявку, а также могут использоваться переменные для замены текста подписи каждой заявки. Переменные, которые можно использовать ищите в нижней части страницы с подписями. Если вы, к примеру, включаете переменную `<OTRS_LAST_NAME>` то она будет заменена фамилией агента, который отвечает на заявку.

Примечание

Как и в случае с приветствиями, подписи не могут быть удалены, только деактивированы путем установки параметра Действительный в значения *не действительный* или *не действительный-временно*.

1.5.3. Вложения

При желании к ответу можно добавить одно или несколько вложений. Если ответ выбран, вложения будут прикреплены к сообщению. При необходимости, непосредственно перед отправкой ответа клиенту, агент может удалить вложения.

Используя ссылку "Вложения" на Панели Администрирования, можно загрузить вложения в базу данных системы (см. нижеприведенный Рисунок).

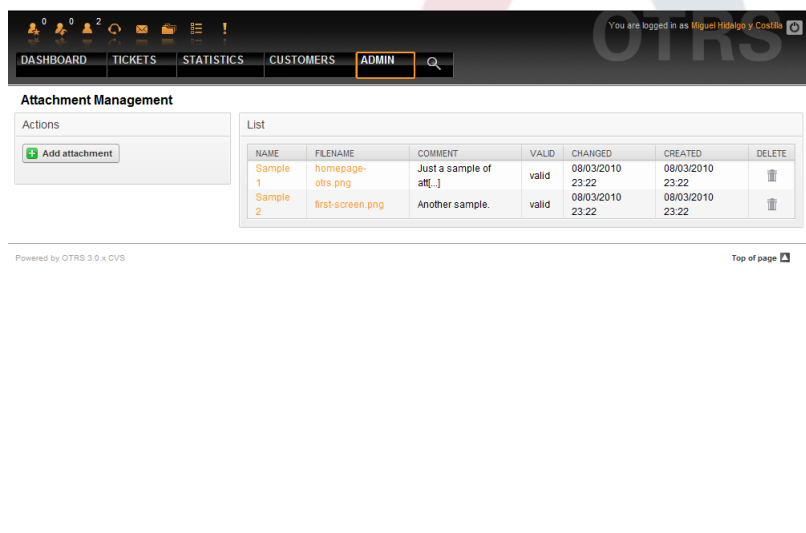
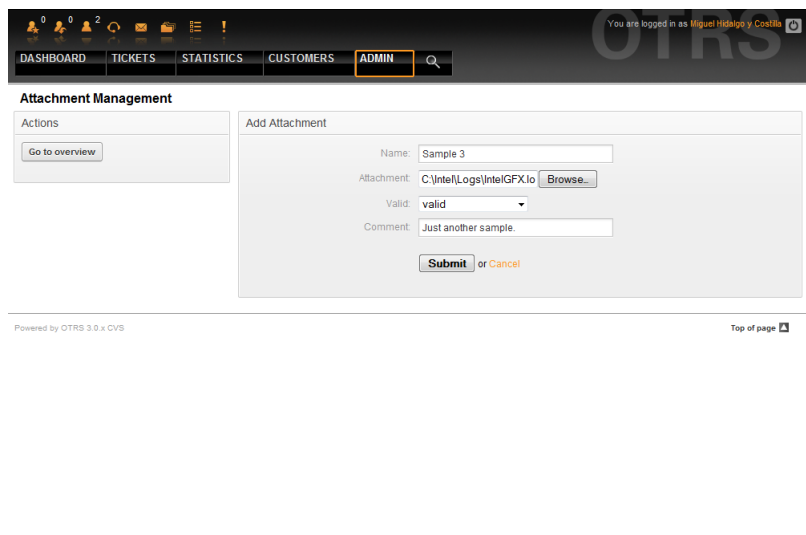


Рисунок: Управление вложениями.

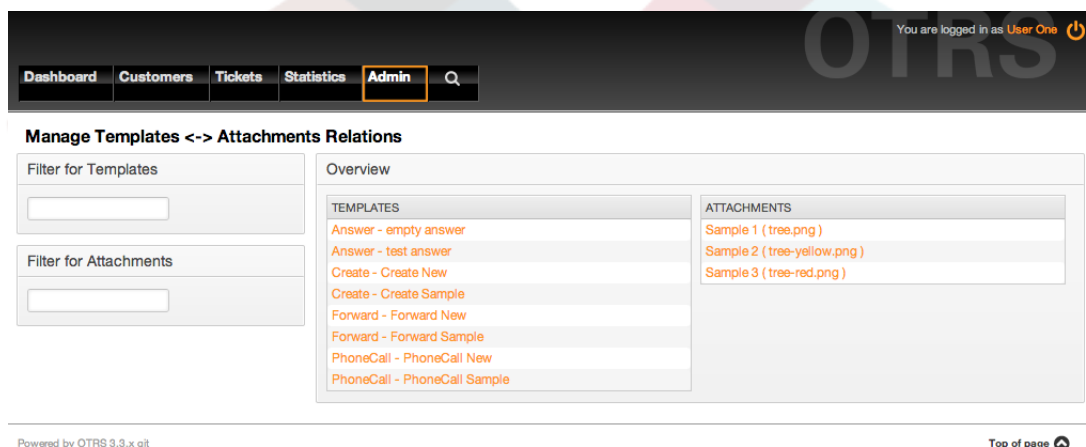
Чтобы создать новое вложение, нажмите кнопку "Добавить вложение", введите всю необходимую информацию и нажмите кнопку "Отправить" (см. нижеприведенный Рисунок).



The screenshot shows the 'Attachment Management' section of the OTRS Admin interface. It includes a navigation bar with 'ADMIN' selected. The main area is titled 'Attachment Management' and contains an 'Add Attachment' form. The form fields are: Name (Sample 3), Attachment (C:\Intel\Logs\Intel\GFX\lo with a 'Browse...' button), Valid (a dropdown menu set to 'valid'), and Comment (Just another sample). There are 'Submit' and 'Cancel' buttons at the bottom of the form. A 'Go to overview' button is located in the 'Actions' section on the left.

Рисунок: Добавление нового вложения.

Когда вложение уже сохранено, его можно связать с одним или несколькими ответами. Для этого перейдите по ссылке "Вложение <-> Ответы" в Панели Администрирования (см. Рисунок ниже).

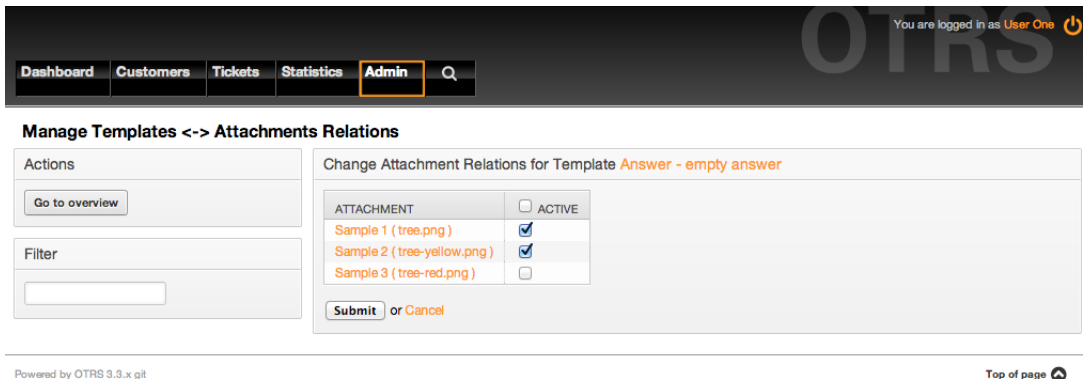


The screenshot shows the 'Manage Templates <-> Attachments Relations' section of the OTRS Admin interface. It features two filter input fields on the left: 'Filter for Templates' and 'Filter for Attachments'. The main area is titled 'Overview' and contains two columns: 'TEMPLATES' and 'ATTACHMENTS'. The 'TEMPLATES' column lists various template names, and the 'ATTACHMENTS' column lists specific attachment files.

TEMPLATES	ATTACHMENTS
Answer - empty answer	Sample 1 (tree.png)
Answer - test answer	Sample 2 (tree-yellow.png)
Create - Create New	Sample 3 (tree-red.png)
Create - Create Sample	
Forward - Forward New	
Forward - Forward Sample	
PhoneCall - PhoneCall New	
PhoneCall - PhoneCall Sample	

Рисунок: Связать вложения с шаблонами..

Чтобы связать различные вложения с конкретным шаблоном и наоборот, нажмите на название соответствующего шаблона или вложения (см. ниже Рисунки 5.27 и 5.28, соответственно).



OTRS You are logged in as User One

Dashboard Customers Tickets Statistics **Admin** Q

Manage Templates <-> Attachments Relations

Actions

Go to overview

Filter

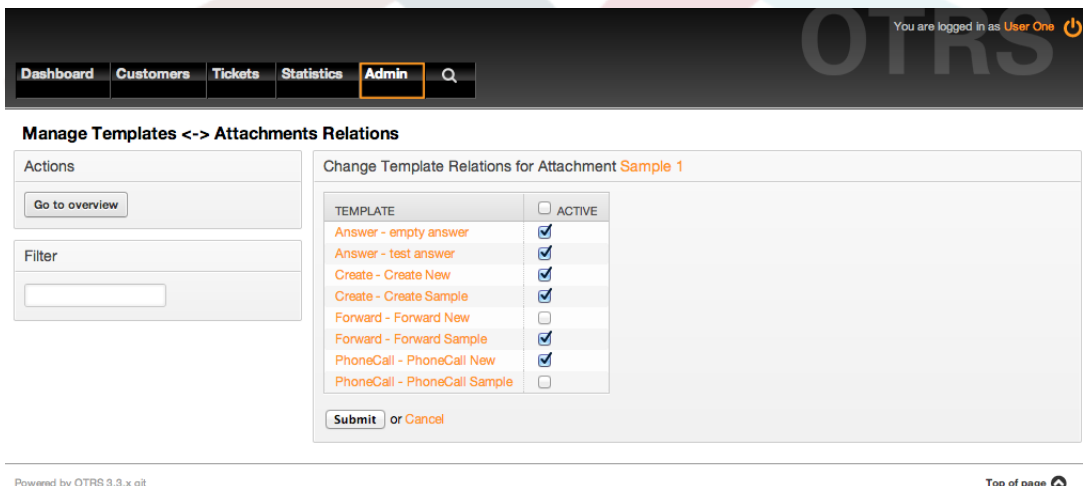
Change Attachment Relations for Template Answer - empty answer

ATTACHMENT	ACTIVE
Sample 1 (tree.png)	<input checked="" type="checkbox"/>
Sample 2 (tree-yellow.png)	<input checked="" type="checkbox"/>
Sample 3 (tree-red.png)	<input type="checkbox"/>

Submit or Cancel

Powered by OTRS 3.3.x git Top of page

Рисунок: Изменение связей между Вложением и Шаблоном.



OTRS You are logged in as User One

Dashboard Customers Tickets Statistics **Admin** Q

Manage Templates <-> Attachments Relations

Actions

Go to overview

Filter

Change Template Relations for Attachment Sample 1

TEMPLATE	ACTIVE
Answer - empty answer	<input checked="" type="checkbox"/>
Answer - test answer	<input checked="" type="checkbox"/>
Create - Create New	<input checked="" type="checkbox"/>
Create - Create Sample	<input checked="" type="checkbox"/>
Forward - Forward New	<input type="checkbox"/>
Forward - Forward Sample	<input checked="" type="checkbox"/>
PhoneCall - PhoneCall New	<input checked="" type="checkbox"/>
PhoneCall - PhoneCall Sample	<input type="checkbox"/>

Submit or Cancel

Powered by OTRS 3.3.x git Top of page

Рисунок :Изменить связь Шаблона с Вложением.

1.5.4. Шаблоны

Для ускорения работы в OTRS и стандартизовать вид ответов, можно задать шаблоны. Шаблон может быть связан с одной или несколькими очередями и наоборот.

Существует несколько различных видов шаблонов, которые могут быть использованы в различных местах в OTRS и в различных целях. Ниже приведен список доступных типов шаблонов:

- Ответ: Будет использоваться в качестве ответа на заявку
- Создать: Будет использоваться в качестве Телефонной Заявки или Email заявки
- Переслать: Используется для пересылки сообщения кому-то другому
- Телефонный звонок: Для регистрации входящих и исходящих звонков агентом

Использовать шаблоны ответов можно двумя способами: при подробном просмотре заявки в меню сообщений/заметок, или более быстрым - с экрана Обзора статусов или заявок в формате предпросмотра/Large. Во вновь установленной OTRS, шаблон (ответа) "пустой ответ" задан для всех очередей.


As soon as Forward templates are added and assigned to queues, the "Forward" button in ticket zoom (that normally leads to a empty text forward screen) will change into a selection control, the selection is filled with the added Forward templates, by choosing one of the templates, the forward screen will be shown prefilled with the template text and attachments (similar to the reply selection box with the Answer templates).

Создание шаблонов типа Create/Новая и PhoneCall/Звонок клиента приводит к отображения поля выбора "Text Template/Текст шаблона" на соответствующих экранах, выбор шаблона из списка приведет к заполнению полей "Текст" и "Вложение" (если они доступны в шаблоне). Отметим, что любые ранее сделанные изменения в поле "Текст" или для вложения, будут перекрыты данными из шаблона.

Нажатие на ссылку "Шаблоны" на Панели Управления OTRS, приведет к открытию раздела Управление шаблонами (см. Рисунок ниже).

Manage Templates

Actions

 Add template










Filter

Hint

A template is a default text which helps your agents to write faster tickets, answers or forwards.

Attention: Don't forget to add new templates to queues.

List


TYPE	NAME	ATTACHMENTS	COMMENT	VALIDITY	CHANGED	CREATED	DELETE
Answer	empty answer	2		valid	08/07/2013 13:47	08/06/2013 16:45	
Answer	test answer	1		valid	08/13/2013 15:01	08/06/2013 16:45	
Create	Create New	2		valid	09/27/2013 10:06	08/07/2013 15:01	
Create	Create Sample	1	My comment	valid	09/27/2013 10:06	08/08/2013 10:02	
Forward	Forward New	0	test	valid	08/12/2013 08:03	08/08/2013 10:06	
Forward	Forward Other	4		invalid	09/27/2013 10:07	08/09/2013 16:53	
Forward	Forward Sample	2		valid	09/27/2013 10:07	08/07/2013 15:20	
PhoneCall	PhoneCall New	2		valid	09/27/2013 10:08	08/15/2013 14:35	
PhoneCall	PhoneCall Sample	1		valid	09/27/2013 10:08	08/15/2013 12:18	


Powered by OTRS 3.3.x git

Top of page 

Рисунок: Управление шаблонами.

Чтобы создать новый шаблон, нажмите кнопку "Добавить шаблон", введите необходимые данные (убедитесь, что выбран правильный тип шаблона) и нажмите кнопку Отправить (см. Рисунок ниже).

You are logged in as **User One** 

Dashboard Customers Tickets Statistics **Admin** 

Manage Templates

Actions

[Go to overview](#)

Hint

A template is a default text which helps your agents to write faster tickets, answers or forwards.





























Attention: Don't forget to add new templates to queues.

Add Template

★ Type:

★ Name:

Template:

B I U S |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Чтобы добавить/удалить шаблоны для одной или нескольких очередей, используйте ссылку "Шаблоны <-> Очереди" в Панели Администрирования (см. Рисунок ниже). Также, чтобы получить информацию о той или иной сущности, можно воспользоваться фильтрами.

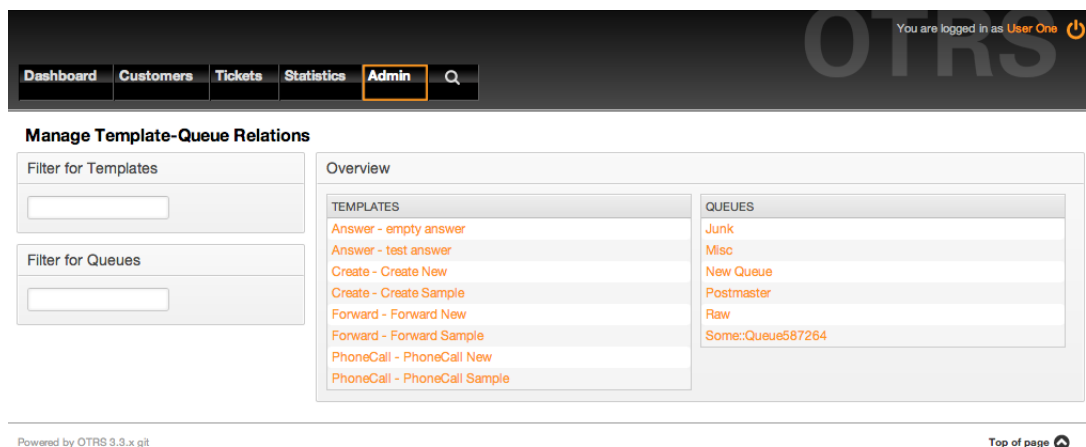




Рисунок: Управление связями Шаблон - Очередь.

Чтобы определить различные шаблоны для очереди и наоборот, нажмите на соответствующий шаблон или очередь (см. ниже Рисунки 5.32 и 5.33, соответственно).

OTRS You are logged in as **User One** 

Dashboard Customers Tickets Statistics **Admin** 

Manage Template-Queue Relations

Change Queue Relations for Template **Answer - empty answer**

Actions
[Go to overview](#)

Filter

QUEUE	ACTIVE
Junk	<input checked="" type="checkbox"/>
Misc	<input checked="" type="checkbox"/>
New Queue	<input type="checkbox"/>
Postmaster	<input checked="" type="checkbox"/>
Raw	<input checked="" type="checkbox"/>
Some:Queue587264	<input type="checkbox"/>

[Submit](#) or [Cancel](#)




Powered by OTRS 3.3.x git Top of page 

Рисунок: Изменение связи между Очередью и Шаблоном.

OTRS You are logged in as **User One** 

Dashboard Customers Tickets Statistics **Admin** 

Manage Template-Queue Relations

Change Template Relations for Queue **Junk**

Actions
[Go to overview](#)

Filter

TEMPLATE	ACTIVE
Answer - empty answer	<input checked="" type="checkbox"/>
Answer - test answer	<input type="checkbox"/>
Create - Create New	<input checked="" type="checkbox"/>
Create - Create Sample	<input checked="" type="checkbox"/>
Forward - Forward New	<input type="checkbox"/>
Forward - Forward Sample	<input checked="" type="checkbox"/>
PhoneCall - PhoneCall New	<input checked="" type="checkbox"/>
PhoneCall - PhoneCall Sample	<input checked="" type="checkbox"/>

[Submit](#) or [Cancel](#)


Powered by OTRS 3.3.x git Top of page 

Рисунок: Изменение связи между Шаблоном и Очередью.

При выборе шаблона, дополнительная информация должна быть добавлена к тексту шаблона, в зависимости от его типа:

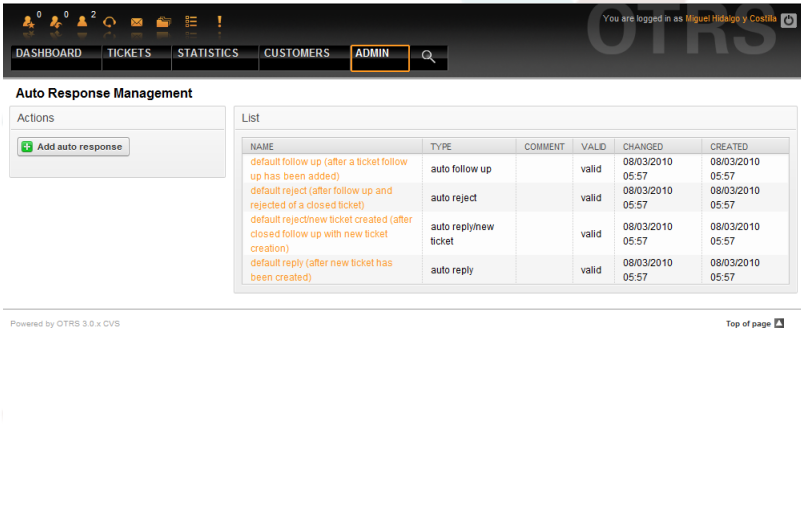
Шаблоны типа PhoneCall/Звонок клиента и Create/Новая не добавляют никаких дополнительных данных к тексту шаблона, в то время как для типа New Email Ticket/Новая почтовая заявка добавляется подпись, связанная с очередью к телу окончательного тела письма (для отображения подписи открывается отдельный фрейм).

В текст шаблона ответа также включается приветствие, связанное с очередью заявки, затем, текст шаблона, после него, цитируемый текст заявки, и, наконец, подпись, связанная с очередью.

Шаблоны типа Forward/Переслать аналогичны шаблонам ответов, но в них не включается приветствие.

1.6. Авто ответы

OTRS позволяет отсылать клиентам автоматические ответы при наступлении в системе определенных событий, таких как создание заявки в определенной очереди, прием откликов клиентов на заявку, закрытие или отказ от заявки и т.д. Для управления ответами используйте ссылку "Авто ответы" на Панели Администрирования (см. Рисунок ниже).



The screenshot shows the OTRS Admin interface. At the top, there is a navigation menu with 'ADMIN' selected. Below the menu, the 'Auto Response Management' section is visible. On the left, there is an 'Actions' panel with a green button labeled 'Add auto response'. On the right, there is a 'List' table with the following data:

NAME	TYPE	COMMENT	VALID	CHANGED	CREATED
default follow up (after a ticket follow up has been added)	auto follow up		valid	08/03/2010 05:57	08/03/2010 05:57
default reject (after follow up and rejected of a closed ticket)	auto reject		valid	08/03/2010 05:57	08/03/2010 05:57
default reject new ticket created (after closed follow up with new ticket creation)	auto reply/new ticket		valid	08/03/2010 05:57	08/03/2010 05:57
default reply (after new ticket has been created)	auto reply		valid	08/03/2010 05:57	08/03/2010 05:57

At the bottom of the interface, it says 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Рисунок: Управление Автоматическими ответами (авто-ответами).

Чтобы создать новый авто-ответ, нажмите на кнопку "Добавить авто-ответ", введите необходимые данные и нажмите кнопку "Отправить" (см. Рисунок ниже).

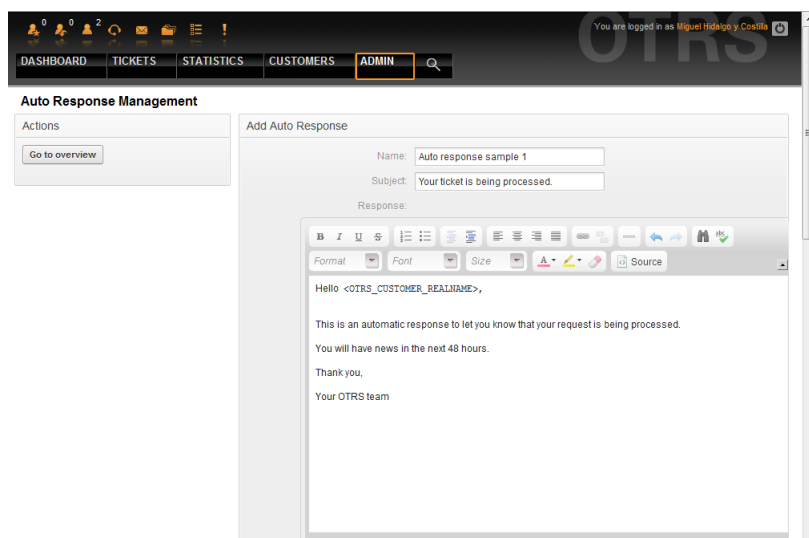


Рисунок: Добавление Автоматического ответа (автоответа).

Как и в случае с подписями и приветствиями, тема и текст авто ответа может быть сгенерирована с помощью переменных. Если, например, в текст/тело авто ответа вставить переменную `<OTRS_CUSTOMER_EMAIL[5]>`, то первые 5 строк текста из письма клиенту будут вставлены в авто ответ. Более подробную информацию о допустимых переменных, которые можно использовать можно найти в нижней части экрана, как это показано на Рисунке.

Для каждого авто-ответа можно указать специальное событие, при наступлении которого этот авто-ответ будет отправляться. Предопределенные события, доступные после новой установки приведены в Таблицу 5-3.

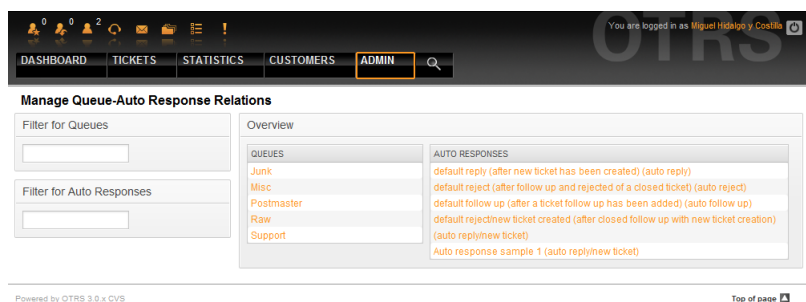
Таблица 4.4. События для Автоматических ответов

Название	Описание
auto reply	Создание заявки в определенной очереди.
auto reply/new ticket	Открытие уже закрытой заявки, например если клиент отвечает на определенную заявку.
auto follow up	Прием входящих заявок.
auto reject	Автоматический отказ от заявки, сделанный системой.
auto remove	Удаление заявок, завершённых системой.

Примечание

Как и в случае с другими сущностями OTRS, Авто ответы также не могут быть удалены, только деактивированы путем установки опции Действительный в значения *не действительный* или *не действительный-временно*.

Чтобы добавить авто ответ к очереди, используйте ссылку "Авто Ответы <-> Очереди" в Панели Администрирования (см. Рисунок ниже). Все системные события отображаются для каждой очереди и авто ответ с тем же событием может быть выбран или удален используя список.

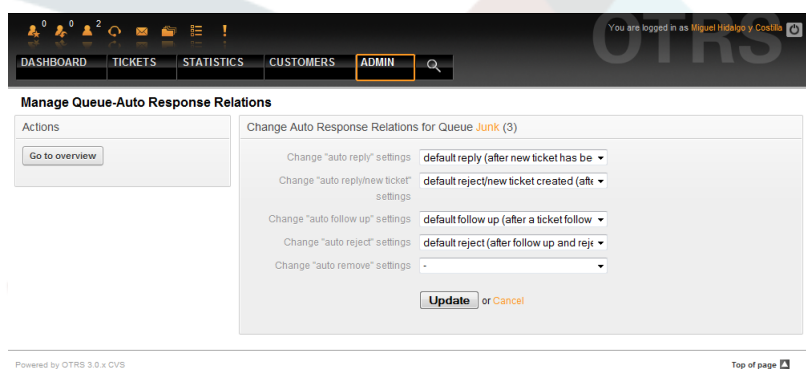


Powered by OTRS 3.0 x CVS

Top of page

Рисунок: Управление связью Очередь-Автоматический ответ.

Чтобы определить различные авто ответы для очереди, нажмите на соответствующее имя очереди (см. Рисунок ниже). Также есть возможность редактировать тот или иной авто ответ, чтобы сделать это, кликните на ответ и внесите необходимые изменения, так как будто создаете новый авто ответ.



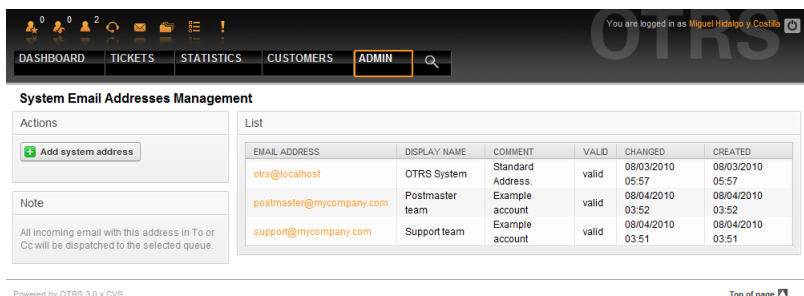
Powered by OTRS 3.0 x CVS

Top of page

Рисунок: Изменение связи между Автоматическим Ответом и Очередью.

1.7. Адреса электронной почты

Чтобы OTRS мог отсылать емейлы, нужен действительный адрес электронной почты, который будет использоваться системой. OTRS способен работать с несколькими адресами электронной почты поскольку большинство систем использует больше одного почтового ящика. Очередь может быть связана с несколькими емейл-адресами и наоборот. При создании очереди можно установить адрес электронной почты, который будет использоваться для исходящих писем. Для управления всеми адресами электронной почты системы перейдите по ссылке "Email-адреса" в Панели Администрирования (см. Рисунок ниже).



System Email Addresses Management

Actions

[Add system address](#)

Note

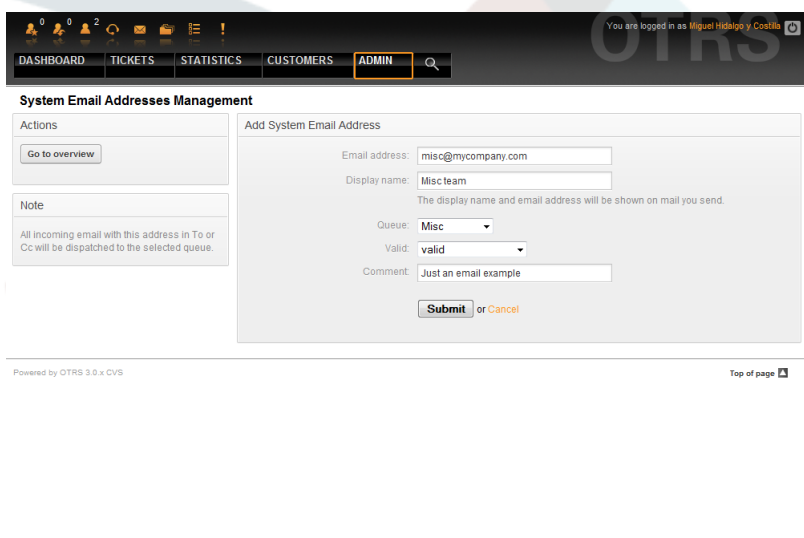
All incoming email with this address in To or Cc will be dispatched to the selected queue.

EMAIL ADDRESS	DISPLAY NAME	COMMENT	VALID	CHANGED	CREATED
otrs@localhost	OTRS System	Standard Address.	valid	08/03/2010 05:57	08/03/2010 05:57
postmaster@mycompany.com	Postmaster team	Example account	valid	08/04/2010 03:52	08/04/2010 03:52
support@mycompany.com	Support team	Example account	valid	08/04/2010 03:51	08/04/2010 03:51

Powered by OTRS 3.0.x CVS Top of page

Рисунок: Управление Системными Адресами Электронной Почты.

При создании нового почтового адреса (см. Рисунок ниже) можно выбрать очередь или под-очередь, которая должна быть связана с этим адресом. Эта связь позволяет системе сортировать входящие сообщения по полю "Кому" и разместить заявку в соответствующую очередь.



System Email Addresses Management

Actions

[Go to overview](#)

Note

All incoming email with this address in To or Cc will be dispatched to the selected queue.

Add System Email Address

Email address:

Display name:

The display name and email address will be shown on mail you send.

Queue:

Valid:

Comment:

or

Powered by OTRS 3.0.x CVS Top of page

Рисунок: Добавление адреса электронной почты системы.

Примечание

Как и другие сущности OTRS, емейл адреса не могут быть удалены, только деактивированы, установив опцию Действительный в значение не действительный или не действительный-временно.

1.8. Уведомления

OTRS позволяет отправлять уведомления агентам и клиентам при наступлении в системе определенных событий. Агенты могут выбрать системные события для своих собственных уведомлений перейдя по ссылке предпочтения

Используя ссылку "Уведомления Агентов" в Панели Администрирования можно управлять уведомлениями системы (см. Рисунок ниже). Используя фильтры можно быстро найти необходимую информацию.

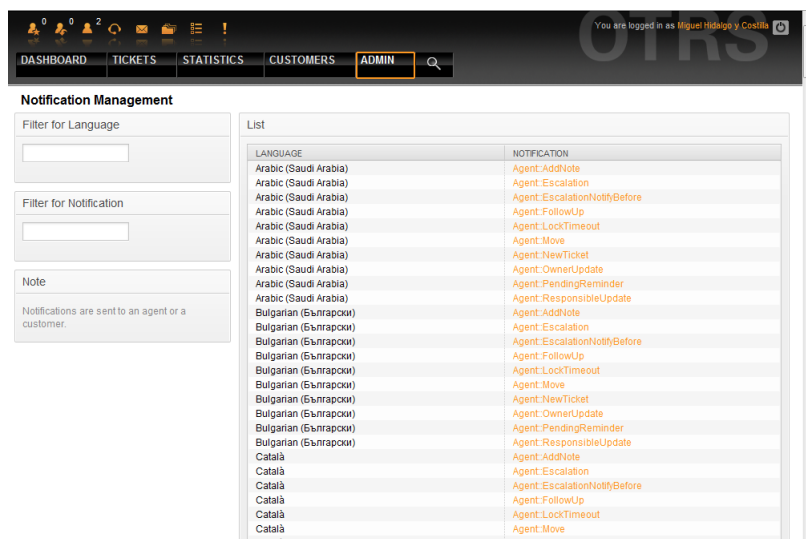


Рисунок: Управление уведомлениями.

Также есть возможность настроить тему и текст уведомления. Выберите из списка уведомление, которое необходимо изменить и его содержимое будет загружено для редактирования (см. Рисунок). Пожалуйста, обратите внимание, что уведомление с тем же именем доступно для каждого из установленных в вашей системе языков.

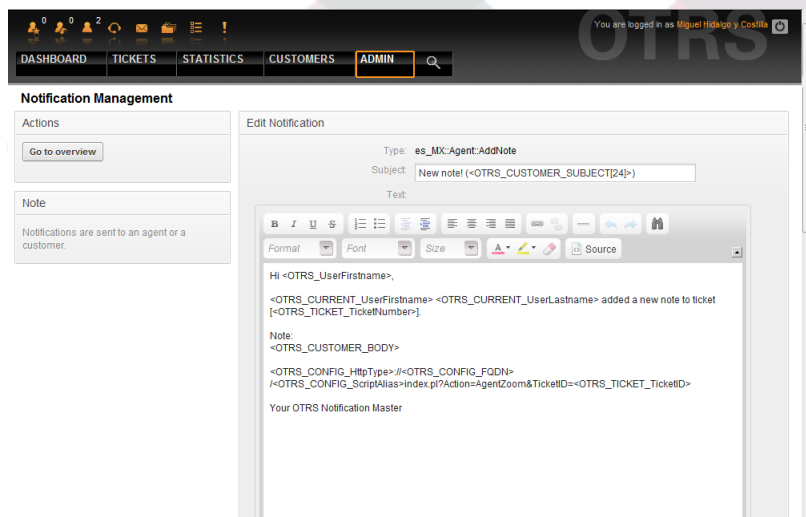


Рисунок: Настройка уведомлений.

Как в случае с подписями и приветствиями, используя специальные переменные можно автоматически создавать контент для уведомлений. В нижней части экрана приводится полный список этих переменных, см. Рисунок.

Также есть возможность создавать уведомления основанные на системных событиях. Можно подробно указать когда и кому такое уведомление будет отправлено. Выбрать можно из большого количества параметров, таких как: группа(ы) получателя, агент(ы), роль(и), адресс(а) электронной почты, тип события, который генерирует уведомление, тип заявки, состояние, приоритет, очередь, блокировка, сервис, SLA и т.д.

Чтобы увидеть список всех событий, основанных на уведомлениях, нажмите на ссылку "Уведомления (События)" в Панели Администрирования (см. Рисунок).

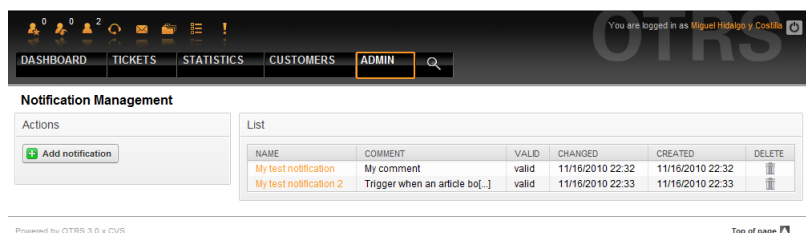


Рисунок: Управления Уведомлениями, базирующихся на Событиях.

Как показано на Рисунке, можно создать новое уведомление нажав на кнопку "Добавить" (см. Рисунок).

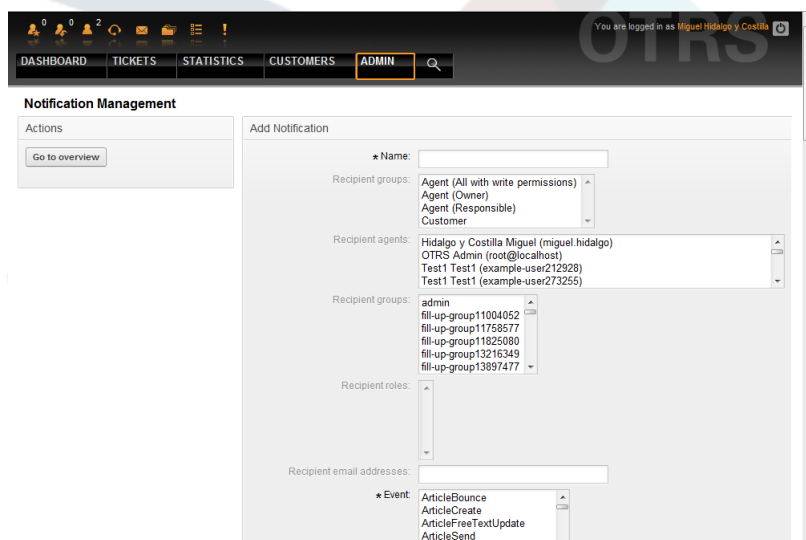


Рисунок: Регистрация Уведомления, основаного на Событии.

Обратите внимание, что содержимое уведомления, которое базируется на событии также может быть создано динамически, используя специальные переменные, приведенные в списке, в нижней части экрана, как это показано на Рисунке.

1.9. SMIME

OTRS может обрабатывать зашифрованные сообщения S/MIME и подписывать исходящие письма. Для использования этой возможности необходимо вначале активировать ее и изменить ряд параметров в SysConfig/Конфигурации системы.

Ссылка "S/MIME Certificates" на Панели Управления OTRS позволяет управлять вашими S/MIME сертификатами (см. Рисунок ниже). Можно добавлять или удалять сертификаты а также производить поиск в SMIME-данных.

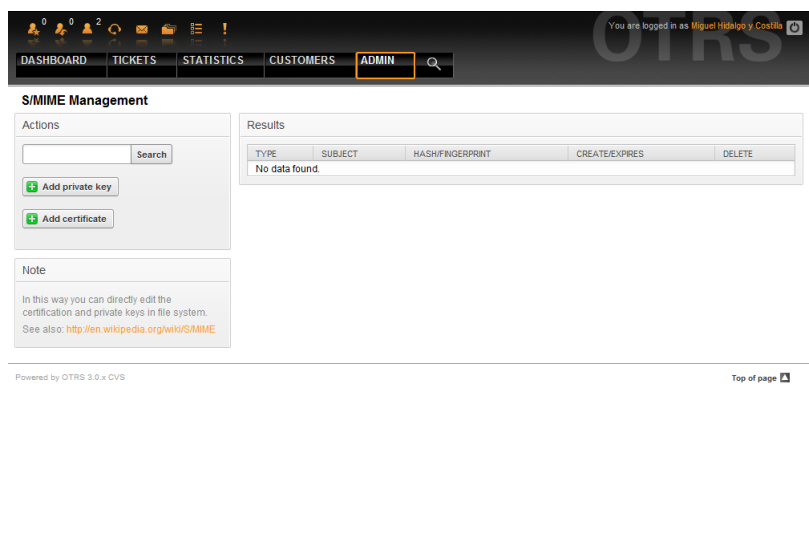


Рисунок: Управление S/MIME.

1.10. PGP

OTRS может обрабатывать ключи PGP, что позволяет шифровать/дешифровать и подписывать исходящие сообщения. Перед использованием этой возможности необходимо активировать ее и изменить некоторые параметры конфигурации в SysConfig.

Используя ссылку "PGP-ключи" на Панели Управления OTRS, можно управлять ключами пользователя, который желает использовать PGP вместе с OTRS (см. Рисунок ниже), например локальный пользователь OTRS или пользователь веб-сервера. Также есть возможность добавлять, удалять ключи и подписи, производить поиск по ключам.

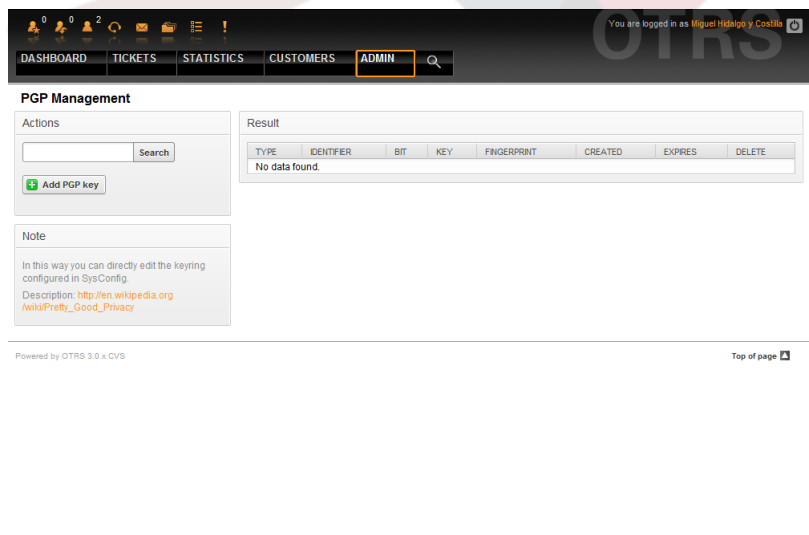


Рисунок: Управление PGP.

1.11. Состояния

Используя ссылку "Состояния" на Панели Управления OTRS, можно управлять различными статусами заявок, которые вы хотите использовать в системе (см. Рисунок ниже).

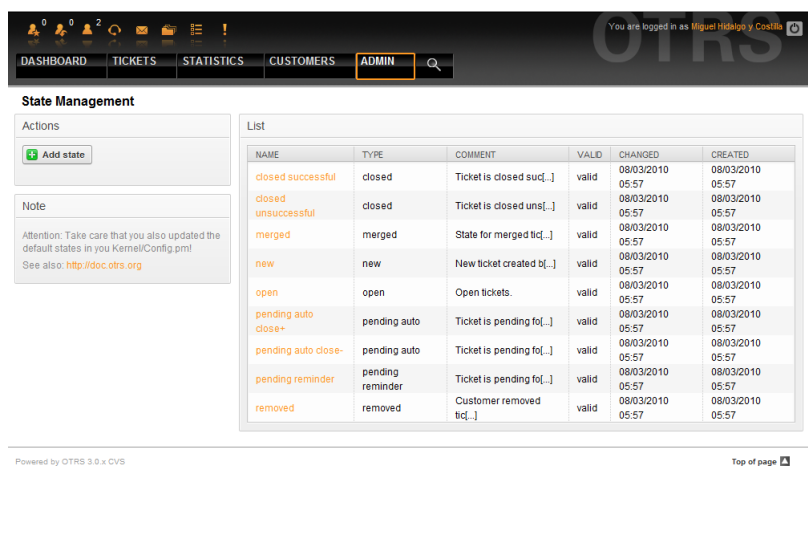


Рисунок: Управление состояниями.

После установки в системе будут следующие предустановленные состояния:

- закрыта успешно
- закрыта неуспешно
- объединенные
- новая
- открытая
- ожидание авто закрытие+
- ожидание авто закрытие-
- ожидание напоминание
- удаленная

Каждое состояние связано с типом, который нужно указать когда создается новое состояние. По умолчанию есть такие типы состояний:

- закрытая
- объединенные
- новая
- открытая
- ожидание авто
- ожидание напоминание
- удаленная

1.12. SysConfig

Ссылка SysConfig - указывает на мест,о где хранится большинство конфигурационные параметров OTRS.

Чтобы загрузить графический интерфейс для конфигурации системы, перейдите по ссылке "SysConfig" в Панели Администрирования (см. Рисунок ниже). В систе-

му можно как загружать собственные конфигурационные файлы, так и архивировать текущие настройки в файл. Почти все конфигурационные параметры OTRS-фреймворка и дополнительно установленных приложений могут быть просмотрены и изменены посредством веб-интерфейса. Поскольку все конфигурационные параметры хранятся в группах и подгруппах, то можно быстро переходить между различными параметрами. Также можно осуществлять полнотекстовый поиск по всем конфигурационным параметрам.

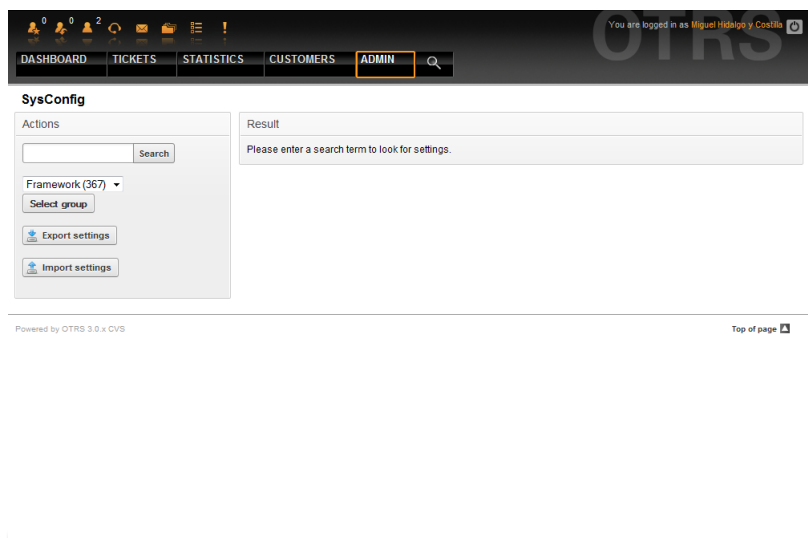


Рисунок: Графический интерфейс для настройки системы (SysConfig).

Более подробно графический интерфейс для настройки системы рассматривается в главе "Настройка системы через веб-интерфейс".

1.13. Использование почтовый ящиков

Есть несколько возможных вариантов доставки новых писем электронной почты в систему обработки заявок. Одним из них является скрипт `otrs.PostMaster.pl`, который напрямую помещает письма в систему. Другой способ - это использование почтовых ящиков, управляемых посредством веб-интерфейса. Перейдя по ссылке "Почтовые ящики PostMaster" в Панели Администрирования, OTRS загружает консоль для управления почтовыми ящиками (см. Рисунок ниже). OTRS поддерживает почтовые протоколы: POP3, POP3S, IMAP и IMAPS.

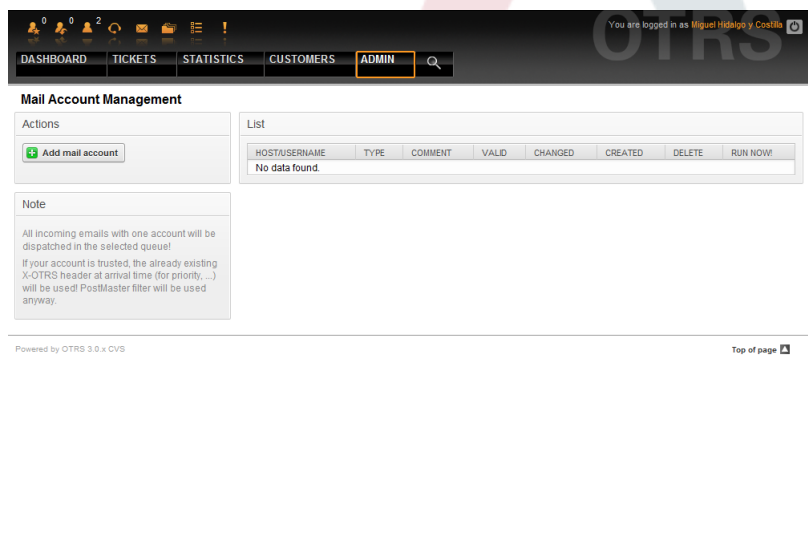


Рисунок: Управление почтовыми ящиками.

За более подробной информацией обращайтесь к разделу Почтовые аккаунты PostMaster.

1.14. Фильтрация входящих почтовых сообщений

OTRS имеет возможность фильтрации входящих сообщений. Например, можно автоматически помещать определенные письма в заданные очереди или присвоить заданное состояние или тип заявки некоторым письмам. Фильтры применяются ко всем входящим письмам. Управлять этими фильтрами можно по ссылке "PostMaster фильтры" с Панели Администрирования (см. рисунок ниже).

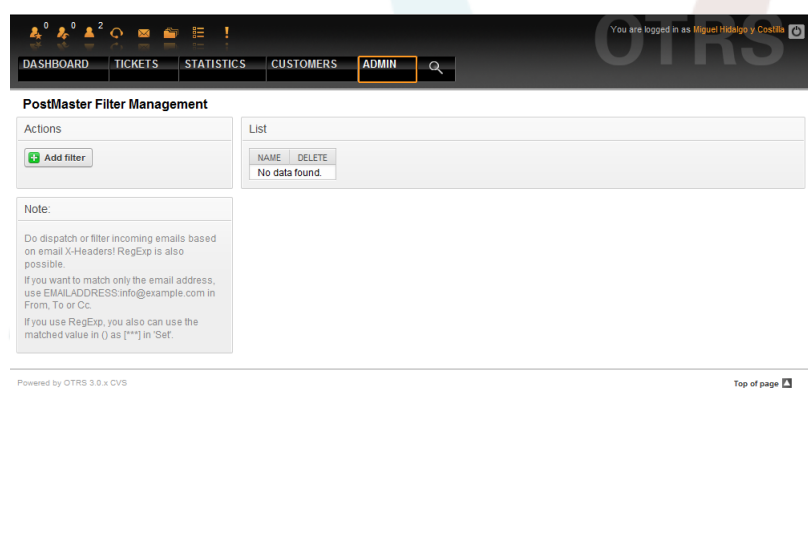


Рисунок: Управление фильтрами PostMaster.

Фильтр состоит из одного или нескольких критериев, которые применяются для определения порядка действий над письмом. Фильтры могут быть определены для заголовка или тела письма, например, для определенных данных заголовка, как то адрес отправителя или определенных строк в теле письма. Можно использовать регулярные выражения для расширенного разбора. Если критерий фильтра выполняется, вы можете заполнить поля, называемые X-OTRS заголовки, в данном интерфейсе. Эти значения будут использованы при создании заявки или регистрации отклика клиента в OTRS. Таблица 5-4 содержит список различных X-OTRS заголовков и их назначение.

Примечание: Вы также можете использовать X-OTRS-Followup-* заголовки для писем с ответами/дополнениями клиентов.

Таблица 4.5. Функции различных X-OTRS-заголовков

Название	Возможные значения:	Описание
X-OTRS-Priority:	1 очень низкий, 2 низкий, 3 нормальный, 4 высокий, 5 очень высокой	Устанавливает приоритет заявки.
X-OTRS-Queue:	Название очереди в системе.	Устанавливает очередь, из которой заявка должна быть распределена. Если установлены X-OTRS-заголовки, все остальные пра-

Название	Возможные значения:	Описание
		вила фильтрации, пытающиеся распределить заявку в определенной очереди игнорируются. Если вы используете подочереди, указывайте их как "Parent::Sub".
X-OTRS-Lock:	блокировать, разблокировать	Устанавливает состояние блокирования заявки.
X-OTRS-Ignore:	Yes или True	Если этот X-OTRS заголовок установлен в "Yes", входящие сообщения будут полностью игнорироваться и никогда не попадут в систему.
X-OTRS-State:	новая, открытая, закрытая успешно, закрытая неуспешно, ...	Устанавливает следующее состояние заявки.
X-OTRS-State-PendingTime:	например 2010-11-20 00:00:00	Задаёт время ожидания для заявки (вы также должны установить состояние ожидания, используя X-OTRS-State). Можно задать абсолютное значение типа "2010-11-20 00:00:00" или относительную дату, основанную на времени поступления письма. Используйте формат "+ \$Number \$Unit", где \$Unit должно быть 's' (секунды), 'm' (минуты), 'h' (часы) или 'd' (дни). Только одна единица измерения может быть задана. Примеры правильного ввода: "+50s" (отложить на 50 секунд), "+30m" (30 минут), "+12d" (12 дней). Значение типа "+1d 12h" недопустимо. Для такого случая правильным будет "+36h".
X-OTRS-Type:	по умолчанию (в зависимости от настроек)	Устанавливает тип заявки (если активировано Ticket::Type).
X-OTRS-Service:	(в зависимости от настроек)	Указывает сервис для заявки (если Ticket::Service включен). Если вы желаете задать подсервис сервиса, он указывается в формате "Parent::Sub".

Название	Возможные значения:	Описание
X-OTRS-SLA:	(в зависимости от настроек)	Устанавливает SLA для заявки (если активирована поддержка Ticket::Service).
X-OTRS-CustomerUser:	CustomerUser	Устанавливает клиентского пользователя для заявки.
X-OTRS-CustomerNo:	CustomerNo	Устанавливает ID пользователя для заявки.
X-OTRS-SenderType:	агент, система, клиент	Устанавливает тип отправителя заявки.
X-OTRS-ArticleType:	email-external, email-internal, email-notification-ext, email-notification-int, phone, fax, sms, webrequest, note-internal, note-external, note-report	Устанавливает тип статьи для входящих заявок.
X-OTRS-DynamicField-<DynamicFieldName>:	В зависимости от конфигурации Динамических Полей (Текст: Notebook, Дата: 2010-11-20 00:00:00, Количество: 1)	Сохраняет дополнительную информацию для заявки в динамическом поле <DynamicFieldName>.
X-OTRS-Loop:	Истина	Если этот X-OTRS заголовок установлен, то отправителю сообщение не доставляется ни одного автоматического ответа (защита от зацыкливания).

Для каждого правила фильтрации нужно указать имя. Критерии фильтрации можно указать в разделе "Условия Фильтрации". Выберите из списка "Заголовок 1", "Заголовок 2" и так далее для части сообщения, где нужно производить поиск и в правой части укажите значения необходимые для фильтрации. В разделе "Установки Email Заголовков" можно указать действия, которые будут выполняться при выполнении правил фильтрации. Также можно выбрать "Заголовок 1", "Заголовок 2" и так далее для X-OTRS-заголовков и установить их значения (см. Рисунок ниже). Правила применяются в алфавитном порядке и выполняются все, за исключением случая, когда параметр "Прекратить после совпадения" установлен в "Да" в одном из правил (в этом случае применение остальных фильтров отменяется).

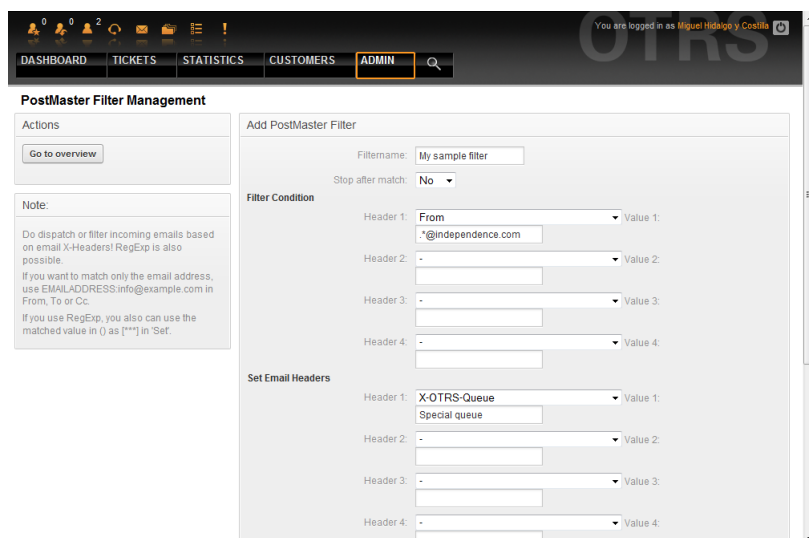


Рисунок: Добавление фильтров PostMaster.

Пример 4.1. Сортирование спама в определенную очередь

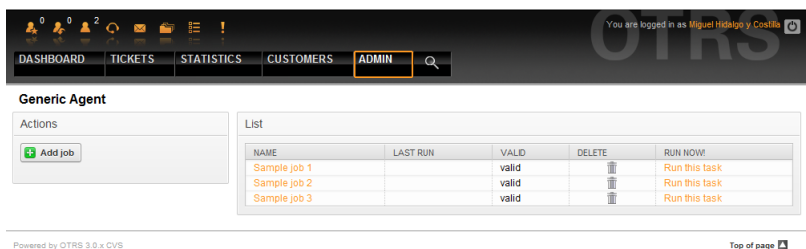
Полезные правила фильтрации помогают OTRS автоматически сортировать почту помеченную как спам с инструментом обнаружения спама, таких как SpamAssassin, в очередь "Junk". SpamAssassin добавляет "X-Spam-Flag" заголовок для каждого проверяемого письма. Если письмо помечено как спам, заголовок устанавливается в "Да". Так критерием фильтра будет "X-Spam-Flag: Yes". Чтобы создать правило фильтрации с этим критерием можно задать имя, например "спам-почты". Также выбрать в разделе "Условия фильтра" для "Заголовок 1" из списка "X-Spam-Flag:". Вставить "Да" в качестве значения для этого заголовка. Теперь критерий фильтра указан. Чтобы убедиться, что все спам-письма попадут в "Junk" очередь выберите в разделе "Установить заголовки письма" для "Заголовок 1" - "X-OTRS-Queue:". Укажите "Junk" в качестве значения для этого заголовка. Наконец добавьте новые правила фильтрации, чтобы активировать его для следующих новых писем в системе.

Есть дополнительные модули, которые можно использовать для более подробной фильтрации входящих сообщений. Эти модули будут полезны для больших и сложных систем.

1.15. Выполнение автоматизированных заданий с помощью GenericAgent

GenericAgent является инструментом для автоматического выполнения задач. GenericAgent, например, может закрывать или перемещать заявки, отправлять уведомления для эскалированных заявок и т.д.

Перейдите по ссылке "GenericAgent" в Панели Управления (см. Рисунок ниже). Откроется таблица со списком автоматизированных заданий. Их можно редактировать, переключив способ вызова на ручной, или удалить.



Dashboard: TICKETS STATISTICS CUSTOMERS ADMIN

You are logged in as Miguel Hidalgo y Costas

Generic Agent

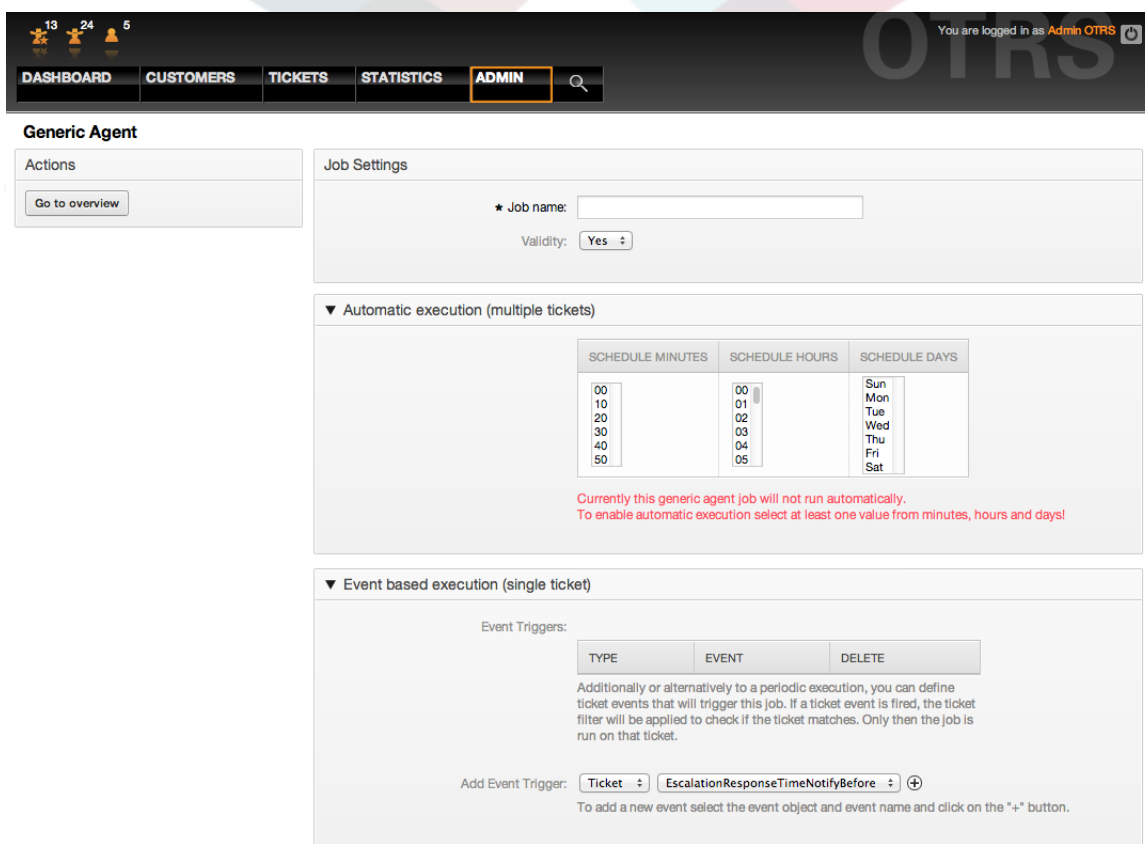
Actions: Add job

NAME	LAST RUN	VALID	DELETE	RUN NOW
Sample job 1		valid		Run this task
Sample job 2		valid		Run this task
Sample job 3		valid		Run this task

Powered by OTRS 3.0.x CVS Top of page

Рисунок: Список заданий для GenericAgent.

Нажмите "Добавить задание" для создания новой задачи. В первую очередь укажите имя. Затем, вы можете указать как она должна выполняться: автоматически в фиксированное время (как cronjob, этот режим будет оперировать всеми заявками, найденными с помощью фильтра заявок) или базируясь на событиях (сразу после изменения отдельной заявки, если она удовлетворяет фильтру). Помните, что если вы вручную запускаете задание, основанное на событии, с экрана обзора заданий, оно будет оперировать всеми заявками прошедшими фильтр.



Dashboard: CUSTOMERS TICKETS STATISTICS ADMIN

You are logged in as Admin OTRS

Generic Agent

Actions: Go to overview

Job Settings

★ Job name:

Validity: Yes

▼ Automatic execution (multiple tickets)

SCHEDULE MINUTES	SCHEDULE HOURS	SCHEDULE DAYS
00	00	Sun
10	01	Mon
20	02	Tue
30	03	Wed
40	04	Thu
50	05	Fri
		Sat

Currently this generic agent job will not run automatically.
To enable automatic execution select at least one value from minutes, hours and days!

▼ Event based execution (single ticket)

Event Triggers:

TYPE	EVENT	DELETE

Additionally or alternatively to a periodic execution, you can define ticket events that will trigger this job. If a ticket event is fired, the ticket filter will be applied to check if the ticket matches. Only then the job is run on that ticket.

Add Event Trigger: Ticket EscalationResponseTimeNotifyBefore +

To add a new event select the event object and event name and click on the "+" button.

Рисунок: Создание заданий для GenericAgent.

Для каждого задания необходимо задать фильтр заявок, например, отбирать только заявки в определенной очереди. Чтобы задание выполнялось для заявки она должна удовлетворять всем критериям фильтра.

Наконец, данные заявки можно изменить, задавая различные поля, такие как очередь или состояние. Можно создать заметку к заявке(ам) или выполнить пользовательский модуль. Можно также удалить заявку(и) из базы данных. Это может быть полезно для очистки системы от ошибочных или устаревших данных, потерявших актуальность.

Предупреждение

Если вы выбрали функцию удаления, все выбранные заявки и вложения к ним будут удалены из базы данных без возможности восстановления!

После редактирования задания, OTRS вернет вас на экран обзора заданий. Здесь вы можете запустить его вручную. Если вы запустили задание, то вначале вы получите список заявок, которые будут обработаны заданием после старта. Этот список поможет вам проверить, что фильтры были заданы правильно. В этот момент еще никаких действий с заявками не выполнялось. Оно начнет работу только после подтверждения вами продолжения работы.

1.16. Электронная почта администратора

OTRS администраторы могут отсылать сообщения определенному пользователю или группе. Переход по ссылке "Уведомления Администратора" в Панели Администрирования открывает страницу, на которой можно указать агентов и группы, которые должны быть оповещены (см. Рисунок ниже).

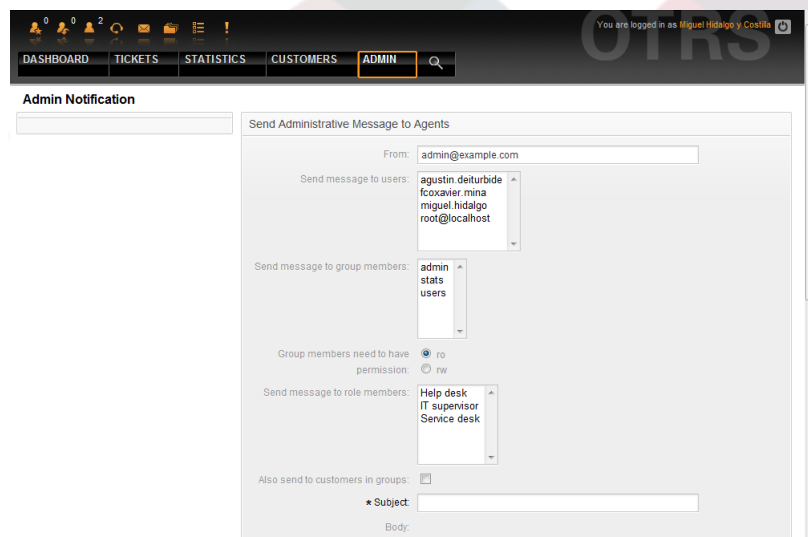


Рисунок: Оповещения Администратором.

Можно указать отправителя, тему и текст уведомления. Также можно выбрать агентов, группы и роли, которые должны получить сообщение.

1.17. Управление сеансами

Перейдя по ссылке "Управление Сеансами" в панели управления вы можете увидеть всех зарегистрированных в системе пользователей и детали их сеансов (см. Рисунок ниже).

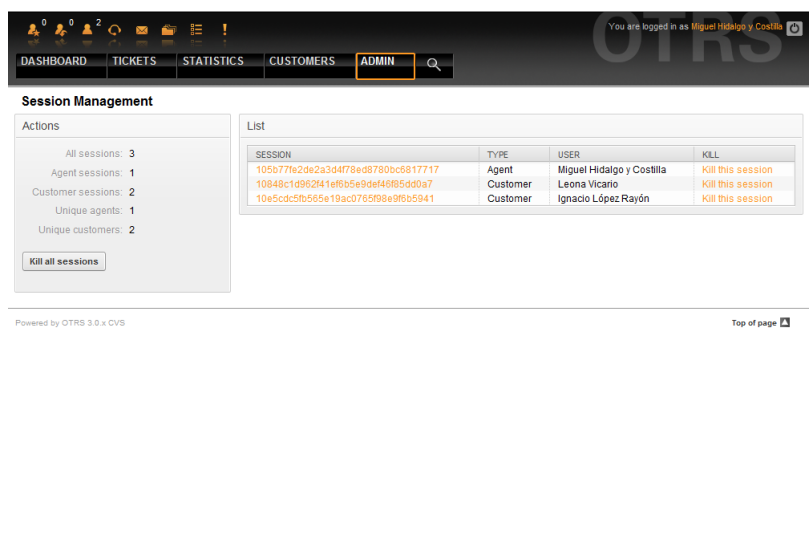


Рисунок: Управление Сеансами.

Выводится статистика о всех активных сессиях, например, сколько агентов и клиентов вошли в систему, количество активных сессий. Кликнув на ссылку "Удалить эту сессию", которая находится в правой части списка, можно удалить ту или иную сессию. Также можно воспользоваться ссылкой *Удалить все сессии*, что может быть полезно чтобы отключить систему. Также можно просмотреть подробную информацию о каждой сессии (см. Рисунок ниже).

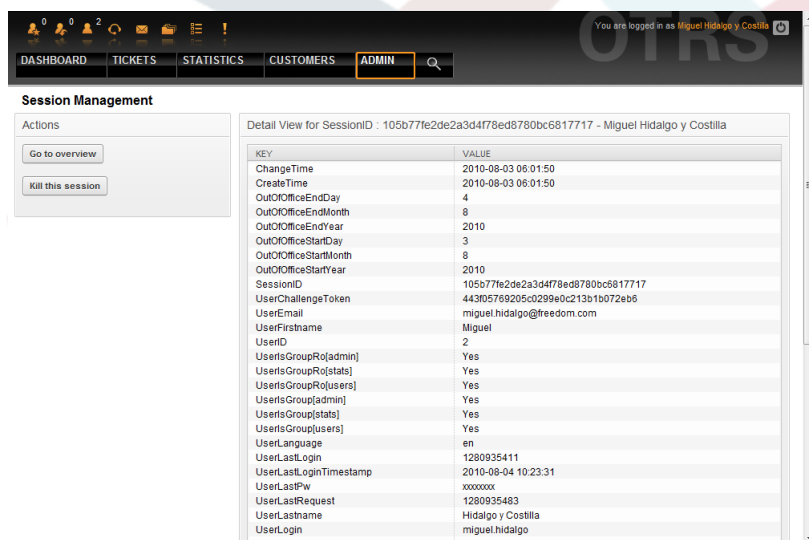
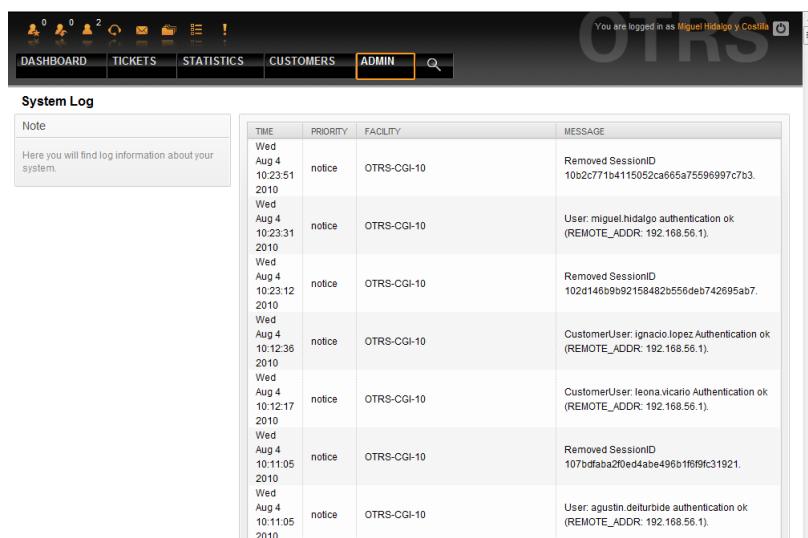


Рисунок: Подробная информация о сессии.

1.18. Системный журнал

Перейдя по ссылке "Системный журнал" в Панели Администрирования можно просмотреть записи в системном журнале, отсортированные в обратном хронологическом порядке (см. Рисунок ниже).



The screenshot shows the OTRS Admin interface with the 'System Log' section active. The log table contains the following entries:

TIME	PRIORITY	FACILITY	MESSAGE
Wed Aug 4 10:23:51 2010	notice	OTRS-CGI-10	Removed SessionID 10b2c771b4115052ca665a75596997c7b3.
Wed Aug 4 10:23:31 2010	notice	OTRS-CGI-10	User: miguel.hidalgo authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:23:12 2010	notice	OTRS-CGI-10	Removed SessionID 102d146b9b92158482b556deb742695ab7.
Wed Aug 4 10:12:36 2010	notice	OTRS-CGI-10	CustomerUser: ignacio.lopez Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:12:17 2010	notice	OTRS-CGI-10	CustomerUser: leona.vicario Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	Removed SessionID 107bdfaba2f0ed4abe496b1f8f9c31921.
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	User: agustin.delburvide authentication ok (REMOTE_ADDR: 192.168.56.1).

Рисунок: Системный журнал.

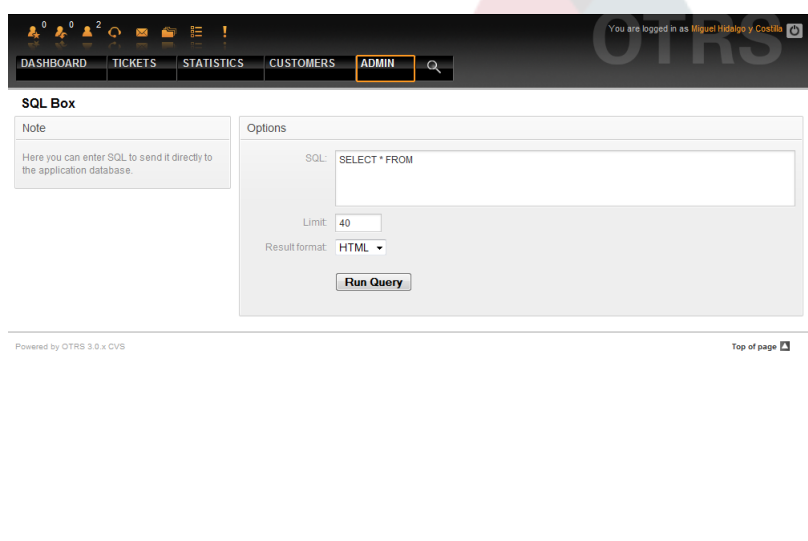
Каждая строка в журнале содержит: время, приоритет, компонент системы и непосредственно саму запись.

Примечание

Системный журнал доступен через веб-интерфейс только на Linux/UNIX системах. В Windows-системах, журнал можно увидеть, открыв в текстовом редакторе файл `[install_dir]otrs\var\log\otrs.log`.

1.19. SQL запросы

Ссылка "SQL Box" в Панели Администрирования открывает окно, которое позволяет напрямую делать запросы к таблицам базы данных системы OTRS (см. Рисунок ниже). Разрешается делать только запросы на выборку данных, изменять данные с помощью запросов нельзя.



The screenshot shows the OTRS Admin interface with the 'SQL Box' section active. The interface includes a text input for SQL queries, a 'Limit' field set to 40, and a 'Result format' dropdown set to HTML. A 'Run Query' button is located below the input fields.

Рисунок: SQL-запросы.

1.20. Менеджер пакетов

Используя ссылку "Менеджер пакетов" в Панели Администрирования, можно устанавливать и управлять пакетами, которые расширяют функциональные возможности OTRS (см. Рисунок ниже). Перейдите по ссылке [Дополнительные приложения](#) чтобы выбрать раздел где находится список и описание расширений, которые доступны из хранилищ OTRS.

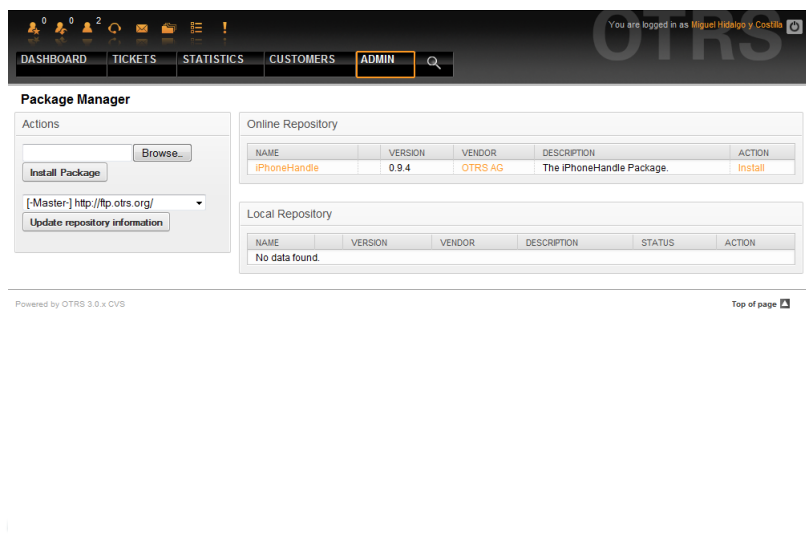


Рисунок: Менеджер пакетов.

Менеджер Пакетов позволяет просмотреть пакеты расширения OTRS вместе с номерами их версий, установленные на сервере в данный момент.

Пакеты можно установить из удаленного хоста, для этого, в разделе *Онлайн Репозиторий* нужно выбрать репозиторий и нажать кнопку *Обновить информацию из репозитория*. Все доступные пакеты будут отображены в соответствующей таблице. В правой части страницы отображаются доступные пакеты. Чтобы установить пакет, нажмите на ссылку *Установить*. После установки пакет будет отображаться в разделе *Локальный Репозиторий*.

Для обновления установленных пакетов можно воспользоваться списком доступных пакетов из он-лайн репозитория. В колонке "действие" для каждого из пакетов, который имеет более высокую версию, чем установленная сейчас, нажмите кнопку *Обновить*. Просто нажмите на кнопку "Обновить", и новый пакет будет обновлен.

В некоторых случаях, например, когда ваша OTRS-система не подключена к Интернету, можно установить пакеты, которые были заранее загружены на локальный диск. Нажмите кнопку *Обзор* на боковой панели и выберите .ortp-файл на жестком диске. Нажмите *Открыть*, а затем *Установить пакет*. После установки пакет отображается в разделе *Локальный Репозиторий*. Прделайте те же действия для того чтобы обновить уже установленный пакет.

В особых случаях, можно настроить Менеджер Пакетов, например, для использования прокси-сервера, или локального репозитория. Перейдя по ссылке `Framework:Core::Package`, можно увидеть на имеющиеся варианты в разделе *Конфигурация системы/SysConfig*.

1.21. Веб-сервисы

Ссылка Веб-сервисы вызывает графический интерфейс (для OTRS Generic Interface), с помощью которого можно создавать и и управлять ими (см. Рисунок ниже).

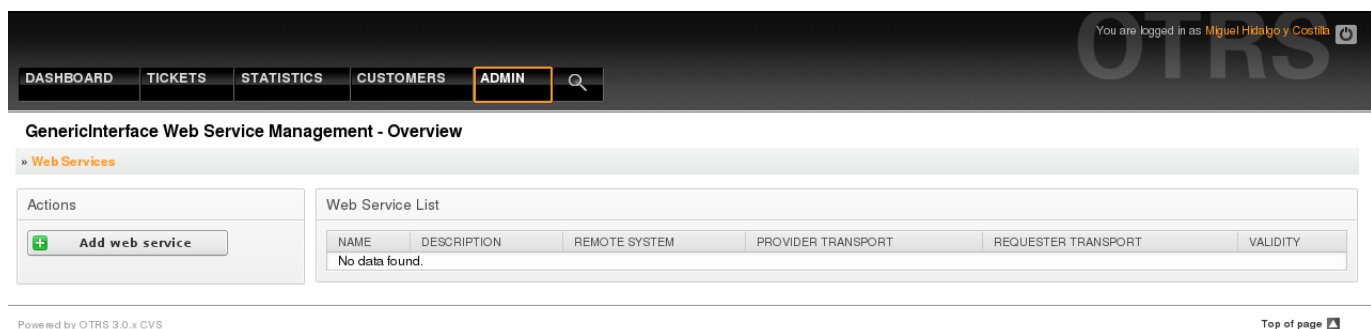


Рисунок: Графический интерфейс для веб-сервисов.

Более подробно графический интерфейс для настройки веб-сервисов рассматривается в главе "Настройка веб-сервисов через веб-интерфейс".

1.22. Динамические Поля

Раздел Динамические поля позволяет создавать, настраивать и управлять пользовательскими полями для заявок и сообщений/заметок (см. рисунок ниже).

DASHBOARD
TICKETS
STATISTICS
CUSTOMERS
ADMIN
🔍

Dynamic Fields Management - Overview

Actions

Article

▼

Add new field for object: Article

Ticket

▼

Add new field for object: Ticket

Hint

To add a new field, select the field type form one of the object's list, the object defines the boundary of the field and it can't be changed after the field creation.

Dynamic Fields List

NAME	LABEL	ORDER	TYPE	OBJECT	VALIDITY
Field1	My Field 1	1	Text	Ticket	valid
Field2	My Field 2	2	Textarea	Ticket	valid
Field3	My Field 3	3	Checkbox	Ticket	valid
Field4	My Field 4	4	Dropdown	Ticket	valid
Field5	My Field 5	5	Multiselect	Ticket	valid
Field6	My Field 6	6	Date	Ticket	valid
Field7	My Field 7	7	Date / Time	Ticket	valid

Powered by OTRS 3.1.x CVS

Рисунок: Страница с некоторыми динамическими полями.

Более подробно описание настройки динамических полей содержится в разделе "Настройка Динамических полей".

Каждый тип динамического поля имеет свои настройки и соответственно свои собственные экраны настройки.

Примечание

При настройке в Панели Администрирования, динамические поля могут быть привязаны только к заявкам или заметкам/сообщениям по умолчанию, но их использование также может быть распространено и на другие объекты.

2. Настройка Системы

2.1. Конфигурационные файлы OTRS

Все конфигурационные файлы OTRS хранятся в директории Kernel и ее поддиректориях. Нет необходимости изменять другие файлы кроме Kernel/Config.pm, потому что остальные файлы будут изменены, после обновления системы. Просто скопируйте конфигурационные параметры из других файлов в Kernel/Config.pm и измените их в соответствии с вашими потребностями. Этот файл никогда не будет затронут во время обновления, так что настройки, которые вы произвели вручную будут сохранены.

В каталоге Kernel/Config/Files есть некоторые другие файлы, которые передаются, когда запускается страница входа в OTRS. Если установлены дополнительные приложения, такие как FAQ (ЧАВО) или Менеджер Файлов, то их конфигурационные файлы также могут быть найдены по указанному пути.

Если веб-интерфейс OTRS доступен - все .xml-файлы из директории Kernel/Config/Files анализируются в алфавитном порядке, затем подгружаются настройки глав-

ного фреймворка и дополнительных приложений. Затем, будут применены настройки в двух файлах Kernel/Config/Files/ZZZAAuto.pm и Kernel/Config/Files/ZZZProcessManagement.pm (если он существует). Эти файлы используются графическим веб-интерфейсом конфигурации системы и никогда не должны редактироваться вручную. Последним применяется файл Kernel/Config.pm, который содержит индивидуальные настройки и параметры конфигурации, изменяемые вручную. Чтение конфигурационных файлов в таком порядке гарантирует, что специфические параметры вашей конфигурации будут использованы системой.

2.2. Настройка системы через графический веб-интерфейс

Начиная с версии OTRS 2.0 и в последующих версиях, с помощью графического веб-интерфейса для настройки системы можно легко изменить почти все конфигурационные параметры центрального фреймворка и дополнительно установленных модулей. Для его использования, войдите в систему как OTRS-администратор и перейдите по ссылке Конфигурация системы/SysConfig в Панели Администрирования (см. Рисунок ниже).

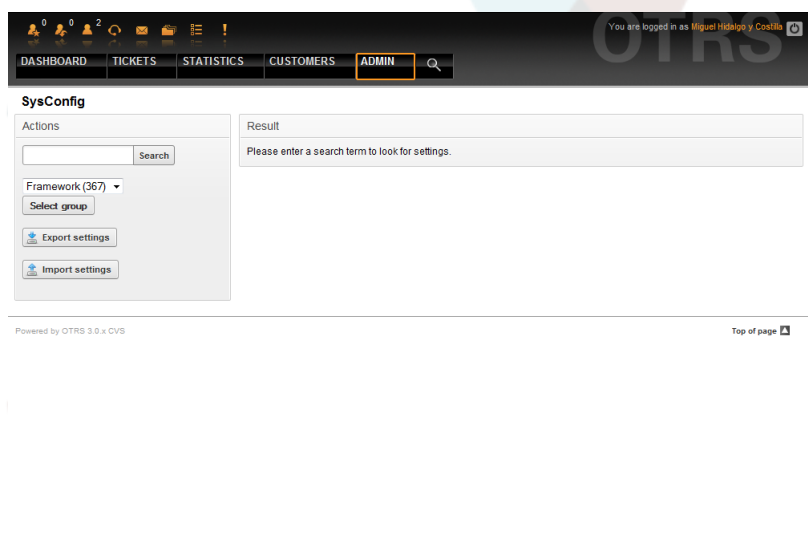


Рисунок: Графический интерфейс для настройки системы.

Поскольку в настоящее время в OTRS имеется больше 600 параметров конфигурации, есть несколько различных способов быстрого доступа к специфическим настройкам. Используя полнотекстовый поиск можно просмотреть все конфигурационные параметры на предмет совпадения с определенным ключевым словом. Полнотекстовый поиск ищет совпадения не только в названиях конфигурационных параметров, но также и в их описаниях. Это позволяет найти тот или иной элемент даже в том случае если его название неизвестно.

Кроме этого, все параметры конфигурации отсортированы по основным группам и подгруппам. Главная группа представляет собой приложение состоящие из подгрупп "Framework" для главного OTRS-фреймворка, "Ticket" для системы обработки заявок, "FAQ" - для системы ответов и вопросов и так далее. Подгруппы могут быть доступны если приложение выбирается из списка групп "Выбор групп" кнопка нажата.

Каждый параметр конфигурации может быть "включен" или "выключен" с помощью флажка. Если параметр "выключен" (флажок не выбран) система будет игнориро-

вать этот параметр или использовать настройки по умолчанию. Также, воспользовавшись кнопкой "Восстановить" можно переключится между измененным значением конфигурационного параметра и его значением по умолчанию.

Если нужно сохранить все изменения, внесенные в конфигурацию системы, например, для быстрой настройки новой установки, можно воспользоваться кнопкой "Экспортировать настройки", которая создаст .pm-файл. Чтобы восстановить свои собственные настройки, нажмите кнопку "Импортировать настройки" и выберите .pm-файл с настройками созданными ранее.

Примечание

Из за соображений безопасности, параметры конфигурации соединения с базой данных не могут быть изменены в SysConfig. Их нужно установлены вручную в файле Kernel/Config.pm.

3. Резервное копирование системы

В этой главе описывается резервное копирование и восстановление данных OTRS.

3.1. Резервное копирование

Есть два типа данных для резервного копирования: файлы приложения (например, файлы в /opt/otrs) и данные, хранящиеся в базе данных.

Для упрощения создания резервных копий, сценарий scripts/backup.pl включен в каждую инсталляцию OTRS. Он позволяет создавать резервную копию всех важных данных (см. Сценарий ниже).

```
linux:/opt/otrs# cd scripts/  
linux:/opt/otrs/scripts# ./backup.pl --help  
backup.pl <Revision 1.1> - backup script  
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>  
usage: backup.pl -d /data_backup/ [-c bzip2|gzip] [-r 30] [-t nofullbackup]  
linux:/opt/otrs/scripts#
```

Сценарий: Получене помощи о механизме резервного копирования OTRS.

Для создания резервной копии выполните команду, приведенную в Сценарии ниже:

```
linux:/opt/otrs/scripts# ./backup.pl -d /backup/  
Backup /backup//2010-09-07_14-28/Config.tar.gz ... done  
Backup /backup//2010-09-07_14-28/Application.tar.gz ... done  
Dump MySQL rdbms ... done  
Compress SQL-file... done  
linux:/opt/otrs/scripts#
```

Сценарий:Создание резервной копии.

Все данные хранятся в директории /backup/2010-09-07_14-28/ (см. Сценарий ниже). Кроме того данные сохранены в файле с расширением .tar.gz.

```
linux:/opt/otrs/scripts# ls /backup/2010-09-07_14-28/  
Application.tar.gz Config.tar.gz DatabaseBackup.sql.gz  
linux:/opt/otrs/scripts#
```

Сценарий: Проверка файлов резервной копии.

3.2. Восстановление

Для восстановления резервной копии, сохраненные данные приложения должны быть записаны обратно в каталог установки, например /opt/otrs. База данных также должна быть восстановлена.

Сценарий `scripts/restore.pl` (см. Сценарий ниже), который упрощает процесс восстановления и поставляется с каждой install-версией OTRS. Поддерживаются СУБД MySQL и PostgreSQL.

```
linux:/opt/otrs/scripts# ./restore.pl --help
restore.pl <Revision 1.1> - restore script
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>
usage: restore.pl -b /data_backup/<TIME>/ -d /opt/otrs/
linux:/opt/otrs/scripts#
```

Сценарий: Получение справки о механизме восстановления.

Данные, которые хранятся в директории `/backup/2010-09-07_14-28/`, могут быть восстановлены с помощью команды, приведенной в приведенном ниже Сценарии при условии что OTRS установлен в `/opt/otrs`.

```
linux:/opt/otrs/scripts# ./restore.pl -b /backup/2010-09-07_14-28 -d /opt/otrs/
Restore /backup/2010-09-07_14-28//Config.tar.gz ...
Restore /backup/2010-09-07_14-28//Application.tar.gz ...
create MySQL
decompresses SQL-file ...
cat SQL-file into MySQL database
compress SQL-file...
linux:/opt/otrs/scripts#
```

Сценарий: Восстановление данных OTRS.

4. Настройки электронной почты

4.1. Отправка / получение электронной почты

4.1.1. Отправка почты

4.1.1.1. через Sendmail (по умолчанию)

OTRS может отправлять сообщения электронной почты через [Sendmail](#), [Postfix](#), [Qmail](#) or [Exim](#)). По умолчанию используется Sendmail, он должен работать при установке системы "из коробки".

Параметры sendmail можно настроить через графический веб-интерфейс для конфигурации (Framework::Core::Sendmail)

4.1.1.2. Через SMTP-сервер или smarthost

OTRS может отсылать сообщения электронной почты через SMTP ([Simple Mail Transfer Protocol / RFC 821](#)) или Secure SMTP. Возможно вы захотите использовать эту опцию на не-unix-системах (например на Windows).

Параметры настройки SMTP-сервера могут быть сконфигурированы через SysConfig (Framework::Core::Sendmail). Если у вас нету SMTPS в качестве опции, зна-

чит вы пропустили установку соответствующих Perl-модулей. В таком случае, обратитесь пожалуйста к этой "Установка Perl-модулей необходимых для работы OTRS" инструкции.

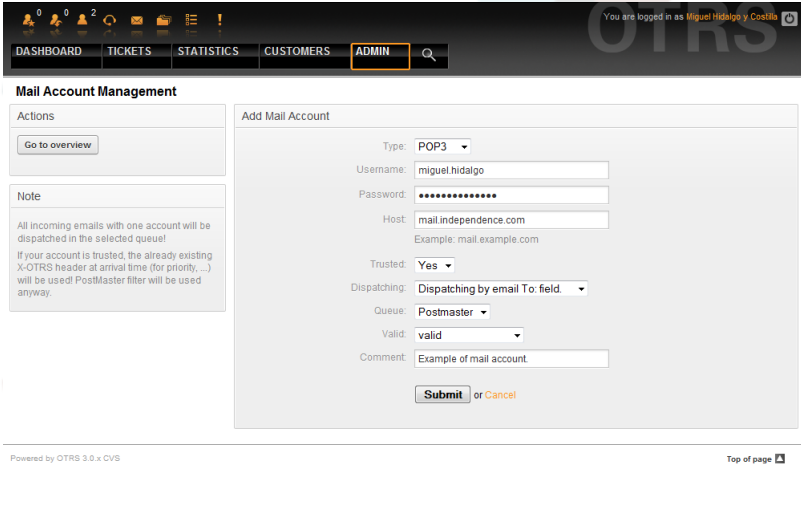
4.1.2. Получение сообщений электронной почты

4.1.2.1. Учетные записи электронной почты настраиваемые через графический пользовательский интерфейс OTRS

OTRS позволяет получать сообщения электронной почты через почтовые аккаунты POP3, POP3S, IMAP, и IMAPS.

Для настройки почтовых аккаунтов воспользуйтесь ссылкой "Почтовые Аккаунты PostMaster" на странице администрирования.

Если создан новый почтовый аккаунт/ящик (см. нижеприведенный сценарий), то нужно еще указать имя его почтового сервера, логин и пароль. Также, необходимо выбрать тип почтового сервера: POP3, POP3S, IMAP или IMAPS. Если в опциях не видно типа сервера, который нужно использовать, значит вы не установили в системе все необходимые Perl-модули. В таком случае, за более подробной информацией обратитесь к инструкции "Установка Perl-модулей, необходимых для работы OTRS".



The screenshot shows the OTRS Admin interface for adding a mail account. The top navigation bar includes Dashboard, Tickets, Statistics, Customers, and Admin. The main content area is titled "Mail Account Management" and contains an "Add Mail Account" form. The form fields are: Type (POP3), Username (miguel.hidalgo), Password (masked with dots), Host (mail.independencia.com), Trusted (Yes), Dispatching (Dispatching by email To: field), Queue (Postmaster), Valid (valid), and Comment (Example of mail account). There are "Submit" and "Cancel" buttons at the bottom of the form. A note on the left side of the form states: "All incoming emails with one account will be dispatched in the selected queue! If your account is trusted, the already existing X-OTRS header at arrival time (for priority, ...) will be used! PostMaster filter will be used anyway."

Рисунок Добавление учетной записи электронной почты.

Если для опции "Trusted" выбрано значение "Да", будут оцениваться и выполняться любые X-OTRS-заголовки, присоединенные к входящему сообщению. Поскольку X-OTRS-заголовок может выполнять некоторые действия в системе обработки заявок, то для известных отправителей опцию Trusted нужно установить только в значение Да. X-OTRS-заголовки используются в OTRS модулем фильтрации. Более подробно X-OTRS заголовки рассматриваются в этой таблице. Любые созданные и выполненные правила фильтрации независимы от параметров настройки Trusted.

Можно контролировать распределением входящих сообщений, если они должны сортироваться по очередям или содержимому поля "Кому". Если для Диспетчеризации выбрана опция "Диспетчеризация по выбранной очереди", то все входящие сообщения будут отсортированы в указанной очереди. В таком случае адрес, с которого отправили сообщение не учитывается. Если для диспетчеризации выбрана опция "Диспетчеризация писем по полю Кому", система для входящих писем проверяет, связана ли эта очередь с адресом электронной почты в поле Кому. Перейдя по ссылке Управление E-mail-адресами в Панели Администрирования, можно связать определенный адрес с очередью. Если не найдено никакой связи между адресом

в поле "Кому" то сообщение будет сохраняться в системе в очереди "Raw" , которая задана в PostmasterDefaultQueue по умолчанию после установки системы.

Все данные для учетных записей электронной почты сохраняются в базе данных OTRS. Скрипт `otrs.PostMasterMailbox.pl`, который находится в директории `bin` вашей системы OTRS, использует настройки в базе данных и получает почту. Вы можете выполнить файл `./bin/otrs.PostMasterMailbox.pl` вручную чтобы проверить что все ваши настройки почты работают правильно.

Для обычной установки, с помощью файла заданий `cron job postmaster_mailbox` выборка емейлов будет выполняться каждые 10 минут. Для получения дополнительной информации об изменении заданий планировщика задач `cron jobs`, обратитесь к главе "Настройка пралировщика задач `cron jobs` для OTRS".

Примечание

При получении почты OTRS удаляет почту с POP или IMAP сервера. Нету такой опции, которая бы позволяла хранить копию сообщения на сервере. Если вы все же хотите чтобы такая возможность присутствовала, скорей всего нужно воспользоваться правилами переадресации на почтовый сервер. Оратитесь пожалуйста к документации по вашему почтовому серверу.

4.1.2.2. Через командную строку программы и например, `procmail (otrs.PostMaster.pl)`

Если для получения электронной почты в OTRS нету возможности использовать учетные записи, то эту проблему можно решить с помощью программы командной строки `bin/otrs.PostMaster.pl` Она принимает сообщения электронной почты через STDIN и направляет их непосредственно в OTRS. Это значит что емейлы будут доступны в OTRS когда MDA (Mail Delivery Agent - Агент доставки почты) выполняет эту программу.

Для тестирования `bin/otrs.PostMaster.pl` без MDA, выполните команду приведенную в листинге нижеприведенного сценария.

```
linux:/opt/otrs# cd bin
linux:/opt/otrs/bin# cat ../doc/sample_mails/test-email-1.box | ./otrs.PostMaster.pl
linux:/opt/otrs/bin#
```

Сценарий: Тестирование PostMaster без MDA.

Если сообщения электронной почты отображаются в QueueView, значит ваши настройки работают.

Procmail - это очень распространенный фильтр электронной почты в среде Linux. Он устанавливается на большинстве систем. Если нет, перейдите по ссылке [procmail homepage](#).

Для настройки `procmail` для OTRS (требуется сконфигурированный транспортный агент MTA, например `sendmail`, `postfix`, `exim` or `qmail`), используйте файл `~otrs/.procmailrc.dist`, скопируйте его в `.procmailrc` а затем добавьте строки из нижеприведенного сценария.

```
SYS_HOME=$HOME
PATH=/bin:/usr/bin:/usr/local/bin
# --
# Pipe all email into the PostMaster process.
# --
:0 :
| $SYS_HOME/bin/otrs.PostMaster.pl
```

Сценарий: Настройка *prosmail* для OTRS.

Все емейлы отсылаемые локальному OTRS-пользователю будут обрабатываться `bin/otrs.PostMaster.pl` и потом отображаться в QueueView.

4.1.2.3. Получение электронной почты по протоколу POP3 или IMAP и обработка для `otrs.PostMaster.pl`

Для того чтобы получить электронную почту с вашего почтового сервера через POP3 или IMAP и сохранить ее на компьютере на котором установлен OTRS, для локального аккаунта или в *prosmail*, перейдите по ссылке [fetchmail](#).

Примечание

Работающий и сконфигурированный SMTP необходим для работы OTRS.

Можно использовать файл `.fetchmailrc.dist` в домашней директории OTRS и скопировать его в `.fetchmailrc`. Изменить его в соответствии с вашими требованиями (см. ниже Пример 7-1).

Пример 4.2. `.fetchmailrc`

```
#poll (mailserver) protocol POP3 user (user) password (password) is (localuser)
poll mail.example.com protocol POP3 user joe password mama is otrs
```

Не забудьте установить `.fetchmailrc` to 710 ("chmod 710 `.fetchmailrc`")!

Из Листинга 7-1 выше, `.fetchmailrc`, все емейлы будут перенаправлены в локальный OTRS-аккаунт, если выполнена команда **fetchmail -a**. Установите эту команду в планировщике задач `cronjob` если хотите извлекать емейлы постоянно.

4.1.2.4. Фильтрация/рассылка модулями OTRS/PostMaster (для более сложной диспетчеризации)

Если вы используете метод `bin/otrs.PostMaster.pl` или `bin/otrs.PostMasterMailbox.pl`, то с модулем фильтрации PostMaster можно вставить или модифицировать X-OTRS заголовок. С помощью X-OTRS-заголовков, система обработки заявок может вызывать некоторые действия для входящих сообщений, сортировать их в определенные очереди, или, например, изменять приоритет или ID-клиента. Более подробную информацию о X-OTRS-заголовках можно найти в главе добавление аккаунтов электронной почты в Панели Администрирования.

Есть некоторые предустановленные модули фильтрации:

Примечание

Название задания (например `$Self->{'PostMaster::PreFilterModule'}->{'JobName'}`) должно быть уникальным!

`Kernel::System::PostMaster::Filter::Match` модуль по умолчанию для проверки совпадения заголовков определенных емейлов (например "От", "Кому", "Тема", ...). Он может устанавливать новые заголовки email (например X-OTRS-Ignore: да или X-OTRS-Queue: `spam`) если совпадают правила совпадения. Задания из Примера 7-2 могут быть прописаны в `Kernel/Config.pm`

Пример 4.3. Пример задания для модуля фильтрации `Kernel::System::PostMaster::Filter::Match`

```
# Job Name: 1-Match
# (block/ignore all spam email with From: noreply@)
$self->{'PostMaster::PreFilterModule'}->{'1-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    From => 'noreply@',
  },
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
};
# Job Name: 2-Match
# (sort emails with From: sales@example.com and Subject: **ORDER**
# into queue 'Order')
$self->{'PostMaster::PreFilterModule'}->{'2-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    To => 'sales@example.com',
    Subject => '**ORDER**',
  },
  Set => {
    'X-OTRS-Queue' => 'Order',
  },
};
```

Kernel::System::PostMaster::Filter::CMD модуль по умолчанию для получения емейлов для внешних команд. Вывод передается в STDOUT и если результат истинна, то устанавливается новый заголовок (например X-OTRS-Ignore: да или X-OTRS-Queue: spam). Пример 7-3 может быть использован в Kernel/Config.pm

Пример 4.4. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::CMD

```
# Job Name: 5-SpamAssassin
# (SpamAssassin example setup, ignore spam emails)
$self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = {
  Module => 'Kernel::System::PostMaster::Filter::CMD',
  CMD => '/usr/bin/spamassassin | grep -i "X-Spam-Status: yes"',
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
};
```

Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition это модуль по умолчанию, который дает возможность разбора/парсинга внешних идентификаторов в теме письма, в его теле или в обоих полях с помощью регулярных выражений. Затем результат разбора сохраняется в заданном динамическом поле. При поступлении письма, OTRS сначала отыскивает внешний идентификатор и, при нахождении, запрашивает OTRS о заранее заданном значении динамического поля. Если найдена существующая заявка, она обновляется, в противном случае создается новая заявка со ссылкой на внешний номер в отдельном поле.

OTRS SysConfig предлагает 4 разных варианта настройки внешних номеров заявки. Если требуются другие варианты, они должны быть добавлены вручную. Следующий пример может быть использован для расширения имеющихся настроек в Kernel/Config.pm.

Пример 4.5. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition

```
# Job Name: ExternalNumber
# External Ticket Number Reconition, check for Incident-<number> in incoming mails
subject and
```



```
# body from the addressees <sender>@externalticket.com, if number is found it will be
stored in
# the dynamic field 'ExternalNumber' (that need to be setup in the Admin Panel).
$self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition5'} = {
  'FromAddressRegExp' => '\\s*@externalticket.com',
  'NumberRegExp'      => 'Incident-(\\d.*)',
  'SearchInSubject'   => '1',
  'SearchInBody'      => '1',
  'TicketStateTypes' => 'new;open',
  'DynamicFieldName' => 'ExternalNumber',
  'Module'            =>
'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
  'Name'              => 'Test External Ticket Number',
  'SenderType'        => 'system',
};
```

Параметры конфигурации

- FromAddressRegExp

Это настройка необязательна. Только письма с заданным значением поля "From:" будут рассмотрены этим фильтром. Вы можете изменить эту настройку на адрес отправителя вашей внешней системы, используемой для исходящей почты. В случае, если адреса различны, вы можете оставить этот параметр пустым. Тогда OTRS не будет проверять адрес отправителя.

- NumberRegExp

Этот параметр обязателен. Он содержит регулярное выражение, которое OTRS будет использовать для извлечения номера заявки из темы и/или тела письма. Выражение по умолчанию будет проверять соответствие, например, 'Incident-12354' и помещать часть значения между скобками в динамическое поле, в данном случае '12354'.

- SearchInSubject

Если установлено в '1', в поле Тема письма будет осуществляться поиск номера заявки.

- SearchInBody

Если установлено в '1', поиск номера заявки будет осуществляться в тексте письма.

- TicketStateTypes

Это необязательный параметр. Если задан, OTRS будет искать открытые внешние заявки с заданными типами состояний. Типы состояний в списке разделяются точкой с запятой.

- DynamicField

Обязательный параметр. Задает динамическое поле, используемое для хранения внешнего номера заявки (имя поля должно существовать в системе и быть действительным).

- SenderType

Этот параметр задает тип отправителя и используется при создании сообщений/заметок в OTRS.

Конечно, также есть возможность разработки своих собственных PostMaster-модулей фильтрации.

4.2. Безопасность электронной почты с PGP

OTRS имеет возможность подписывать или шифровать исходящие сообщения с помощью PGP. Также, могут быть расшифрованы входящие зашифрованные сообщения. Шифрование и дешифрование осуществляется с помощью GPL-инструмента GnuPG. Чтобы установить GnuPG для OTRS, нужно проделать следующие действия:

1. Установка GnuPG с помощью менеджера пакетов вашей операционной системы.
2. Настройте GnuPG для использования с OTRS. Для GnuPG и частного ключа нужно создать необходимые директории. В командной строке для пользователя 'otrs' нужно выполнить команду из нижеприведенного сценария.

```
linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --gen-key
gpg (GnuPG) 1.4.2; Copyright (C) 2005 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: directory `/opt/otrs/.gnupg' created
gpg: new configuration file `/opt/otrs/.gnupg/gpg.conf' created
gpg: WARNING: options in `/opt/otrs/.gnupg/gpg.conf' are not yet active during t
his run
gpg: keyring `/opt/otrs/.gnupg/secring.gpg' created
gpg: keyring `/opt/otrs/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) DSA and Elgamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection? 1
DSA keypair will have 1024 bits.
ELG-E keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y

You need a user ID to identify your key; the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
  "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: Ticket System
Email address: support@example.com
Comment: Private PGP Key for the ticket system with address support@example.com
You selected this USER-ID:
"Ticket System (Private PGP Key for the ticket system with address support@examp
le.com) <support@example.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 0
You need a Passphrase to protect your secret key.

Passphrase: secret
Repeat passphrase: secret

We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
```

```

generator a better chance to gain enough entropy.
+++++++.....+++++.....+++++
+++++++.....+++++.....+++++>+++++>+++++
.....>+++++<+++++.....+++++

Not enough random bytes available. Please do some other work to give
the OS a chance to collect more entropy! (Need 280 more bytes)

+++++++.....+++++.....+++++.....+++++.....+++++.....+++++.....+++++
+++++++.....+++++.....+++++.....+++++.....+++++.....+++++.....+++++
..+++++>.....>+++++.....>+++++<+++++<+++++.....>+++++<+++++.....
.....>+++++<+++++.....
.....+++++^^^
gpg: /opt/otrs/.gnupg/trustdb.gpg: trustdb created
gpg: key 7245A970 marked as ultimately trusted
public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
pub 1024D/7245A970 2006-02-03
    Key fingerprint = 2ED5 BC36 D2B6 B055 7EE1 5833 1D7B F967 7245 A970
uid          Ticket System (Private pgp key for ticket system with addre
ss support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

Сценарий: Настройка GnuPG.

Как показано в нижеприведенном сценарии, настройки по умолчанию могут быть применены для большинства из необходимых параметров. Правильно должны быть указаны только пароль и значения для владельца ключа.

3. На странице с PGP-настройками PGP, PGP должен быть активирован для OTRS (первая опция). Также нужно задать и проверить путь к gpg-программе.

Следующим нужно изменить конфигурационный параметр (PGP::Options). Используя эти конфигурационные настройки можно определить параметры, которые будут использоваться для каждого вызова gpg пользователем 'otrs'. Каталог с конфигурационными файлами для GnuPG является очень важным. В примере используется /opt/otrs/.gnupg. Эта директория была создана на первом шаге конфигурации PGP.

Используя следующий конфигурационный параметр (PGP::Key::Password), можно указать пары для ключей ID и их пароли для собственных закрытых ключей. Поскольку партнеры извне пишут зашифрованные сообщения в систему заявок используя ваш публичный ключ, OTRS может расшифровать эти сообщения с помощью ID/паролей указанных здесь.

Как получить идентификатор вашего собственного закрытого ключа? ID вашего собственного закрытого идентификатора отображается во время генерации (см. выше шаг 1). Также есть возможность получить ID, указав команду из нижеприведенного сценария вызванную с правами пользователя 'otrs':

```

linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --list-keys
/opt/otrs/.gnupg/pubring.gpg
-----
pub 1024D/7245A970 2006-02-03
uid          Ticket System (Private pgp key for ticket system with
address support@example.com) <support@example.com>

```

```
sub 2048g/52B97069 2006-02-03
```

```
linux:~$
```

Сценарий: Получение ID вашего собственного приватного ключа.

Идентификатор (ID) закрытого ключа можно найти в строке, которая начинается с "sub". Это шестнадцатеричное представление строки, длиной в восемь символов, например "52B97069". Пароль, который нужно указать для этого ключа в системе обработки заявок такой же как данные при генерации ключей.

После того, как эти данные были введены, нажмите кнопку "Обновить" чтобы сохранить настройки. Теперь OTRS готов принимать и расшифровывать зашифрованные сообщения.

4. И наконец, нужно импортировать открытый ключ клиента. Это гарантирует, что зашифрованные сообщения могут быть отправлены этому клиенту. Есть два способа импортировать открытый ключ клиента.

Первый способ - указать открытый ключ клиента в веб-интерфейсе управления клиентами.

Вторая возможность заключается в определении ключей посредством настройки PGP, которая доступна через Панель Администрирования. В правой стороне этой страницы отображаются все импортированные открытые ключи клиентов. После того как PGP активирован и сконфигурирован, здесь также должен отображаться ваш собственный открытый ключ. В левой части страницы настроек PGP можно производить поиск по ключам. Также, новый открытый ключ может быть загружен в систему из файла.

Файлы с открытым ключем, которые нужно импортировать в OTRS должны соответствовать файлам ключей GnuPGP. В большинстве случаев, ключ сохраняется в файле как "ASCII armored key". OTRS позволяет работать с этим форматом.

4.3. Безопасность электронной почты с S/MIME

На первый взгляд, S/MIME-шифрование кажется более сложным чем PGP. Прежде всего для OTRS нужно создать Центр Сертификации - Certification Authority (CA). Последующие процедуры очень похожи на те, которые вы выполняли для PGP: настроить OTRS, установить собственные сертификаты, импортировать другие открытые сертификаты, если необходимо, и т. д.

В большинстве случаев настройка S/MIME производится за пределами веб-интерфейса OTRS, и должна осуществляться в оболочке операционной системы с правами пользователя 'otrs'. MIME конфигурация под Linux основана на SSL (OpenSSL). Поэтому, сначала проверьте установлен ли в вашей системе пакет OpenSSL. OpenSSL-пакет содержит скрипт, который называется CA.pl, для создания сертификата, все наиболее важные шаги, нужно проделать именно с ним. Для упрощения процедуры узнайте, где в файловой системе находится сценарий CA.pl и введите путь к его местоположению в переменную путей оболочки (см. нижеприведенный сценарий).

```
otrs@linux:~> rpm -ql openssl | grep CA
/usr/share/ssl/misc/CA.pl
otrs@linux:~> export PATH=$PATH:/usr/share/ssl/misc
otrs@linux:~> which CA.pl
/usr/share/ssl/misc/CA.pl
otrs@linux:~> mkdir tmp; cd tmp
otrs@linux:~/tmp>
```

Сценарий: Настройка S/MIME.

Сценарий выше показывает, что была создана новая временная директория ~/tmp, в которой хранятся сгенерированные сертификаты.

Для создания сертификата, проделайте следующие операции в командной строке (предположим, что OTRS-администратор должен создать SSL-сертификат с целью тестирования и обучения. В случае, если у вас уже есть SSL-сертификат для шифрования, используйте его и пропустите следующие шаги):

1. Создание собственного Центра Сертификации (Certification Authority) для SSL. Вы должны подтвердить запрос на собственный SSL-сертификат (см. нижеприведенный сценарий).

```
otrs@linux:~/tmp> CA.pl -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
...+++++
.....+++++
writing new private key to './demoCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:OTRS-state
Locality Name (eg, city) []:OTRS-town
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your company
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:OTRS Admin
Email Address []:otrs@your-domain.tld
otrs@linux:~/tmp> ls -la demoCA/
total 8
-rw-r--r--  1 otrs otrs 1330 2006-01-08 17:54 cacert.pem
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 certs
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 crl
-rw-r--r--  1 otrs otrs   0 2006-01-08 17:53 index.txt
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 newcerts
drwxr-xr-x  2 otrs otrs  80 2006-01-08 17:54 private
-rw-r--r--  1 otrs otrs  17 2006-01-08 17:54 serial
otrs@linux:~/tmp>
```

Сценарий: Создание Центра Сертификации (Certification Authority) для SSL.

2. Создание запроса сертификата (см. Сценарий ниже).

```
otrs@linux:~/tmp> CA.pl -newreq
Generating a 1024 bit RSA private key
.....+++++
....+++++
writing new private key to 'newreq.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
```

```

For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE\keyreturn
State or Province Name (full name) [Some-State]:OTRS-state
Locality Name (eg, city) []:OTRS-town
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your company
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:OTRS admin
Email Address []:otrs@your-domain.tld

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Request (and private key) is in newreq.pem
otrs@linux:~/tmp> ls -la
total 4
drwxr-xr-x  6 otrs otrs  232 2006-01-08 17:54 demoCA
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
otrs@linux:~/tmp>

```

Сценарий: Создание запроса сертификата.

3. Подпись запроса сертификата. Запрос сертификата может быть подписан, и таким образом сертифицирован вашим собственным центром сертификации (CA), что более правдоподобно, чем использование других, внешних CA (см. нижеприведенный сценарий).

```

otrs@linux:~/tmp> CA.pl -signreq
Using configuration from /etc/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
  Serial Number:
    fd:85:f6:9f:14:07:16:c8
  Validity
    Not Before: Jan  8 17:04:37 2006 GMT
    Not After  : Jan  8 17:04:37 2007 GMT
  Subject:
    countryName           = DE
    stateOrProvinceName   = OTRS-state
    localityName          = OTRS-town
    organizationName      = Your Company
    commonName            = OTRS administrator
    emailAddress          = otrs@your-domain.tld
  X509v3 extensions:
    X509v3 Basic Constraints:
      CA:FALSE
    Netscape Comment:
      OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:
      01:D9:1E:58:C0:6D:BF:27:ED:37:34:14:D6:04:AC:C4:64:98:7A:22
    X509v3 Authority Key Identifier:
      keyid:10:4D:8D:4C:93:FD:2C:AA:9A:B3:26:80:6B:F5:D5:31:E2:8E:DB:A8
      DirName:/C=DE/ST=OTRS-state/L=OTRS-town/O=Your Company/
      CN=OTRS admin/emailAddress=otrs@your-domain.tld
      serial:FD:85:F6:9F:14:07:16:C7

Certificate is to be certified until Jan  8 17:04:37 2007 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Signed certificate is in newcert.pem
otrs@linux:~/tmp>

```

Сценарий: Подпись запроса сертификата.

4. Создание своего собственного сертификата, и все данные собираются с ним, используя запрос сертификата (см. Сценарий ниже).

```
otrs@linux:~/tmp> CA.pl -pkcs12 "OTRS Certificate"
Enter pass phrase for newreq.pem:
Enter Export Password:
Verifying - Enter Export Password:
otrs@linux:~/tmp> ls -la
total 12
drwxr-xr-x  6 otrs otrs  328 2006-01-08 18:04 demoCA
-rw-r--r--  1 otrs otrs 3090 2006-01-08 18:13 newcert.p12
-rw-r--r--  1 otrs otrs 3791 2006-01-08 18:04 newcert.pem
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
otrs@linux:~/tmp>
```

Сценарий: Создание нового сертификата.

Теперь эти операции выполнены, установка S/MIME в OTRS должна быть завершена.

Эта часть установки осуществляется в Панели Администрирования, выбрав ссылку "SMIME". В случае, если в OTRS нету поддержки S/MIME, маска указывает администратору удобные ссылки чтобы активировать поддержку S/MIME.

С помощью группы параметров SysConfig "Crypt::SMIME", вы можете также включить и настроить общую поддержку протоколов S/MIME.

Здесь можно активировать поддержку S/MIME, определить пути для команд OpenSSL, и директории для сертификатов. Файл ключей должен храниться в директории, указанной здесь. В противном случае OpenSSL не будет их использовать.

Следующим шагом, который следует выполнить - перейти по ссылке Настройка S/MIME в Панели Администрирования. Здесь можно импортировать закрытые ключи OTRS-системы и открытые ключи других партнеров. Введите публичный ключ, который был создан и добавлен в OTRS в начале этого раздела.

Перейдя по ссылке Инструменты администрирования клиентов можно импортировать все открытые S/MIME ключи партнеров по общению.

5. Использование внешних хранилищ данных

5.1. Данные клиентов

OTRS работает с различными атрибутами данных клиентов, такими как имя пользователя, адрес электронной почты, номер телефона и т.д. Эти атрибуты отображаются в обеих фронтэндах, для Агентов и Клиентов. Они также используются для проверки подлинности клиентов.

Данные клиентов, которые используются и отображаются в OTRS очень легко настраиваются. Несмотря на это следующая информация будет всегда необходима для проверки подлинности клиента:

- Вход пользователей
- Адрес электронной почты

- ID Клиента

Используйте следующие параметры SysConfig, если желаете отобразить информацию о клиенте в интерфейсе агента.

```
# Ticket::Frontend::CustomerInfo*
# (show customer info on Compose (Phone and Email), Zoom and
# Queue view)
$self->{'Ticket::Frontend::CustomerInfoCompose'} = 1;
$self->{'Ticket::Frontend::CustomerInfoZoom'} = 1;
```

Скрипт: Параметры конфигурации SysConfig.

5.2. Пользовательский бэк-енд

Вы можете использовать два типа хранилища информации клиентов: Базу Данных DB и LDAP. Если у вас уже есть другой бэк-енд для хранения пользовательской информации (например SAP), также есть возможность написать модуль для использования этой функции.

5.2.1. База Данных (По умолчанию)

В Примере 11-1 приведена конфигурация базы данных, которая использует данные клиента, хранящиеся в базе данных OTRS.

Пример 4.6. Настройка клиентского хранилища базы данных (DB)

```
# CustomerUser (customer database backend and settings)
$self->{'CustomerUser'} = {
  Name => 'Database Datasource',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the required settings
    # DSN => 'DBI:odbc:yourdsn',
    # Type => 'mssql', # only for ODBC connections
    # DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
    # User => '',
    # Password => '',
    Table => 'customer_user',
    # if your frontend is unicode and the charset of your
    # customer database server is iso-8859-1, use these options.
    # SourceCharset => 'iso-8859-1',
    # DestCharset => 'utf-8',

    # CaseSensitive will control if the SQL statements need LOWER()
    # function calls to work case insensitively. Setting this to
    # 1 will improve performance dramatically on large databases.
    CaseSensitive => 0,
  },
  # customer unique id
  CustomerKey => 'login',

  # customer #
  CustomerID => 'customer_id',
  CustomerValid => 'valid_id',
  CustomerUserListFields => ['first_name', 'last_name', 'email'],
  CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
  CustomerUserSearchPrefix => '',
  CustomerUserSearchSuffix => '*',
  CustomerUserSearchListLimit => 250,
  CustomerUserPostMasterSearchFields => ['email'],
  CustomerUserNameFields => ['title', 'first_name', 'last_name'],
  CustomerUserEmailUniqCheck => 1,
  # # show not own tickets in customer panel, CompanyTickets
```



```
# CustomerUserExcludePrimaryCustomerID => 0,
# # generate auto logins
# AutoLoginCreation => 0,
# AutoLoginCreationPrefix => 'auto',
# # admin can change customer preferences
# AdminSetPreferences => 1,
# # cache time to live in sec. - cache any database queries
# CacheTTL => 0,
# # just a read only source
# ReadOnly => 1,
# Map => [
#   # note: Login, Email and CustomerID needed!
#   # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target
  [ 'UserTitle',      'Title',      'title',      1, 0, 'var', '', 0 ],
  [ 'UserFirstname', 'Firstname',  'first_name', 1, 1, 'var', '', 0 ],
  [ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var', '', 0 ],
  [ 'UserLogin',     'Username',  'login',      1, 1, 'var', '', 0 ],
  [ 'UserPassword',  'Password',  'pw',         0, 0, 'var', '', 0 ],
  [ 'UserEmail',     'Email',     'email',      1, 1, 'var', '', 0 ],
#   [ 'UserEmail',    'Email',    'email',      1, 1, 'var', '$Env{"CGIHandle"}?
Action=AgentTicketCompose&ResponseID=1&TicketID=$Data{"TicketID"}&ArticleID=
$Data{"ArticleID"}', 0 ],
  [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
#   [ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],
  [ 'UserPhone',     'Phone',     'phone',      1, 0, 'var', '', 0 ],
  [ 'UserFax',       'Fax',       'fax',        1, 0, 'var', '', 0 ],
  [ 'UserMobile',    'Mobile',    'mobile',     1, 0, 'var', '', 0 ],
  [ 'UserStreet',    'Street',    'street',     1, 0, 'var', '', 0 ],
  [ 'UserZip',       'Zip',       'zip',        1, 0, 'var', '', 0 ],
  [ 'UserCity',      'City',      'city',       1, 0, 'var', '', 0 ],
  [ 'UserCountry',   'Country',   'country',    1, 0, 'var', '', 0 ],
  [ 'UserComment',   'Comment',   'comments',   1, 0, 'var', '', 0 ],
  [ 'ValidID',       'Valid',     'valid_id',   0, 1, 'int', '', 0 ],
],
# default selections
Selections => {
  UserTitle => {
    'Mr.' => 'Mr.',
    'Mrs.' => 'Mrs.',
  },
},
};
```

Если нужно настроить данные клиентов, измените заголовки столбцов или добавьте новые к таблице customer_user базы данных OTRS. В качестве примера используйте нижеприведенный сценарий, в котором показано как добавить новое поле для номера комнаты.

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 116 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (250);
Query OK, 1 rows affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

Сценарий: Добавление поля "комната" в таблицу customer_user table.

Теперь добавьте новый столбец в MAP-массив в Kernel/Config.pm, как это показано в нижеприведенном сценарии.

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[...]
[ 'UserRoom',      'Room',      'room',      0, 1, 'var', '', 0 ],
```

Сценарий: Добавление поля "комната" в файл Kernel/Config.pm file.

Всю эту информацию о клиенте можно также отредактировать воспользовавшись ссылкой Клиенты в интерфейсе агента.

5.2.1.1. Клиент с несколькими идентификаторами (Заявки Компании)

Одному клиенту можно назначить больше одного клиентского идентификатора (Customer ID). Это может быть полезно, если клиенту необходимо получить доступ к заявкам других клиентов, например, руководитель хочет посмотреть заявки своих помощников. Если клиент может получить доступ к заявкам другого клиента, то используется особенность OTRS "заявки компании". Заявки компании могут быть доступны перейдя по ссылке "Заявки Компании" в клиентской панели управления.

Для использования заявок компании, новый столбец с идентификатором (IDs), который должен быть доступен для клиента, должен быть добавлен в таблицу customer_user базы данных OTRS (см. Сценарий ниже).

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 124 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD customer_ids VARCHAR (250);
Query OK, 1 rows affected (0.02 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

Сценарий: Добавление поля customer_ids в таблицу customer_user.

Теперь новый столбец должен быть добавлен в MAP-массив в Kernel/Config.pm, как это показано в нижеприведенном Сценарии.

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[...]
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],
```

Сценарий: Добавление поля UserCustomerIDs в Kernel/Config.pm

Теперь, новый столбец для мульти-идентификатора (IDs) клиента можно редактировать с помощью веб-интерфейса Агента, в разделе управления клиентами.

Для того, чтобы убедиться что один клиент имеет доступ к заявкам других клиентов нужно добавить идентификаторы (IDs) этих пользователей в новое поле для нескольких идентификаторов клиента. Каждый идентификатор (ID) должен быть отделен точкой с запятой (см. ниже Пример 11-2).

Пример 4.7. Хранение Заявок Компании в базе данных DB

Клиенты А, Б и Ц созданы в вашей системе и А хочет иметь доступ к заявкам Б и Ц используя клиентскую панель. Б и Ц не должны иметь доступа к заявкам других пользователей.

Для реализации этой структуры, измените таблицу `customer_user` и маппинг (преобразование) в `Kernel/Config.pm` как это показано выше. С помощью Панели Администрирования или используя ссылку Клиенты в веб-интерфейсе Агента загрузите настройки пользователя А. Если настройки отображаются, добавьте значения "Б;Ц;" в поле для `CustomerIDs`.

5.2.2. LDAP

Если у вас есть LDAP-каталог, в котором хранятся данные о клиентах, его можно использовать в OTRS, в качестве хранилища данных о клиентах, как это показано в Примере 11-3.

Пример 4.8. Настройка LDAP в качестве клиентского бэк-енда

```
# CustomerUser
# (customer ldap backend and settings)
$self->{CustomerUser} = {
    Name => 'LDAP Data Source',
    Module => 'Kernel::System::CustomerUser::LDAP',
    Params => {
        # ldap host
        Host => 'bay.csuhayward.edu',
        # ldap base dn
        BaseDN => 'ou=seas,o=csuh',
        # search scope (one|sub)
        SSCOPE => 'sub',
        # The following is valid but would only be necessary if the
        # anonymous user does NOT have permission to read from the LDAP tree
        UserDN => '',
        UserPw => '',
        # in case you want to add always one filter to each ldap query, use
        # this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
        '(objectclass=user)'
        AlwaysFilter => '',
        # if both your frontend and your LDAP are unicode, use this:
        SourceCharset => 'utf-8',
        DestCharset => 'utf-8',
        # if your frontend is unicode and the charset of your
        # ldap server is iso-8859-1, use these options.
        # SourceCharset => 'iso-8859-1',
        # DestCharset => 'utf-8',
        # Net::LDAP new params (if needed - for more info see perl doc Net::LDAP)
        Params => {
            port => 389,
            timeout => 120,
            async => 0,
            version => 3,
        },
    },
    # customer unique id
    CustomerKey => 'uid',
    # customer #
    CustomerID => 'mail',
    CustomerUserListFields => ['cn', 'mail'],
    CustomerUserSearchFields => ['uid', 'cn', 'mail'],
}
```

```

CustomerUserSearchPrefix => '',
CustomerUserSearchSuffix => '*',
CustomerUserSearchListLimit => 250,
CustomerUserPostMasterSearchFields => ['mail'],
CustomerUserNameFields => ['givenname', 'sn'],
# show not own tickets in customer panel, CompanyTickets
CustomerUserExcludePrimaryCustomerID => 0,
# add an ldap filter for valid users (expert setting)
# CustomerUserValidFilter => '!(description=locked)',
# administrator can't change customer preferences
AdminSetPreferences => 0,
# # cache time to live in sec. - cache any database queries
# CacheTTL => 0,
Map => [
  # note: Login, Email and CustomerID are mandatory!
  # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly
  [ 'UserTitle',      'Title',      'title',          1, 0, 'var', '', 0 ],
  [ 'UserFirstname', 'Firstname', 'givenname',     1, 1, 'var', '', 0 ],
  [ 'UserLastname',  'Lastname',  'sn',            1, 1, 'var', '', 0 ],
  [ 'UserLogin',     'Username',  'uid',           1, 1, 'var', '', 0 ],
  [ 'UserEmail',     'Email',     'mail',          1, 1, 'var', '', 0 ],
  [ 'UserCustomerID', 'CustomerID', 'mail',          0, 1, 'var', '', 0 ],
#   [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1, 0, 'var', '', 0 ],
  [ 'UserPhone',     'Phone',     'telephonenumber', 1, 0, 'var', '', 0 ],
  [ 'UserAddress',   'Address',   'postaladdress',  1, 0, 'var', '', 0 ],
  [ 'UserComment',   'Comment',   'description',    1, 0, 'var', '', 0 ],
],
};

```

Если дополнительные атрибуты клиента хранятся в LDAP-каталоге, например, имя руководителя, номер мобильного телефона, или отдела, и если эту информацию нужно отобразить в OTRS, просто расширьте MAP-массив в файле `Kernel/Config.pm` записями для этих атрибутов, как показано в нижеприведенном сценарии.

```

# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[...]
[ 'UserPhone',      'Phone',      'telephonenumber', 1, 0, 'var', '', 0 ],

```

Сценарий: Добавление нового поля в файл `Kernel/Config.pm`.

5.2.2.1. Клиент с несколькими идентификаторами (Заявки Компании)

При использовании LDAP-бэкэнда клиенту можно присвоить больше одного клиентского айди (Customer ID). Для использования заявок компании, в LDAP-директорию нужно добавить новое поле, которое содержит доступные агенту идентификаторы (IDs).

Если в LDAP-каталоге было создано новое поле, его также нужно добавить в MAP-массив в `Kernel/Config.pm`, как это показано в нижеприведенном сценарии.

```

# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[...]
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],

```

Сценарий: Подстановка новых полей в `Kernel/Config.pm` file.

Клиентские идентификаторы (IDs) можно редактировать напрямую в LDAP-директории. OTRS может только считать информацию из LDAP, но не записывать.

Чтобы убедиться что клиент имеет доступ к заявкам других клиентов, добавьте идентификаторы (IDs) клиентов, к заявкам которых нужен доступ, к новому полю

вашей LDAP-директории. Каждый идентификатор ID должен быть отделен точкой с запятой (см. ниже Рисунок 11-4).

Пример 4.9. Использование заявок Компании с LDAP-бэкендом

Клиенты А, Б и Ц созданы в вашей системе и А хочет иметь доступ к заявкам Б и Ц через панель клиента. Б и Ц не должны иметь доступа к другим пользователям.

Для реализации этой установки измените LDAP-директорию и маппинг (mapping) в Kernel/Config.pm, как это показано выше. Затем добавьте в вашей LDAP-директории значения "Б;Ц;" в поле для CustomerIDs, для клиента "А".

5.2.3. Использование больше чем одного пользовательского хранилища данных с OTRS

Если в OTRS нужно использовать больше одного источника данных о клиентах (например LDAP и базу данных), конфигурационный параметр CustomerUser должен быть расширен числом, например "CustomerUser1", "CustomerUser2" (см. ниже Пример 11-5).

Пример 4.10. Использование больше чем одного пользовательского хранилища данных с OTRS

В следующем примере показано применение конфигурации как для LDAP так и для базы данных клиентского бэкенда с OTRS.

```
# 1. Customer user backend: DB
# (customer database backend and settings)
$self->{CustomerUser1} = {
    Name => 'Customer Database',
    Module => 'Kernel::System::CustomerUser::DB',
    Params => {
        # if you want to use an external database, add the
        # required settings
        DSN => 'DBI:odbc:yourdsn',
        Type => 'mssql', # only for ODBC connections
        DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
        User => '',
        Password => '',
        Table => 'customer_user',
    },
    # customer unique id
    CustomerKey => 'login',
    # customer #
    CustomerID => 'customer_id',
    CustomerValid => 'valid_id',
    CustomerUserListFields => ['first_name', 'last_name', 'email'],
    CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
    CustomerUserSearchPrefix => '',
    CustomerUserSearchSuffix => '*',
    CustomerUserSearchListLimit => 250,
    CustomerUserPostMasterSearchFields => ['email'],
    CustomerUserNameFields => ['title', 'first_name', 'last_name'],
    CustomerUserEmailUniqCheck => 1,
    # show not own tickets in customer panel, CompanyTickets
    CustomerUserExcludePrimaryCustomerID => 0,
    # generate auto logins
    AutoLoginCreation => 0,
    AutoLoginCreationPrefix => 'auto',
    # admin can change customer preferences
    AdminSetPreferences => 1,
    # cache time to live in sec. - cache any database queries
    CacheTTL => 0,
    # just a read only source
    ReadOnly => 1,
    Map => [
```

```

# note: Login, Email and CustomerID needed!
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target
[ 'UserTitle',      'Title',      'title',      1, 0, 'var', '', 0 ],
[ 'UserFirstname', 'Firstname',  'first_name', 1, 1, 'var', '', 0 ],
[ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var', '', 0 ],
[ 'UserLogin',     'Username',  'login',      1, 1, 'var', '', 0 ],
[ 'UserPassword',  'Password',  'pw',         0, 0, 'var', '', 0 ],
[ 'UserEmail',     'Email',     'email',      1, 1, 'var', '', 0 ],
[ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
[ 'UserPhone',     'Phone',     'phone',      1, 0, 'var', '', 0 ],
[ 'UserFax',       'Fax',       'fax',        1, 0, 'var', '', 0 ],
[ 'UserMobile',    'Mobile',    'mobile',     1, 0, 'var', '', 0 ],
[ 'UserStreet',    'Street',    'street',     1, 0, 'var', '', 0 ],
[ 'UserZip',       'Zip',       'zip',        1, 0, 'var', '', 0 ],
[ 'UserCity',      'City',      'city',       1, 0, 'var', '', 0 ],
[ 'UserCountry',   'Country',   'country',    1, 0, 'var', '', 0 ],
[ 'UserComment',   'Comment',   'comments',   1, 0, 'var', '', 0 ],
[ 'ValidID',       'Valid',     'valid_id',   0, 1, 'int', '', 0 ],
],
# default selections
Selections => {
  UserTitle => {
    'Mr.' => 'Mr.',
    'Mrs.' => 'Mrs.',
  },
},
};

# 2. Customer user backend: LDAP
# (customer ldap backend and settings)
$self->{CustomerUser2} = {
  Name => 'LDAP Datasource',
  Module => 'Kernel::System::CustomerUser::LDAP',
  Params => {
    # ldap host
    Host => 'bay.csuhayward.edu',
    # ldap base dn
    BaseDN => 'ou=seas,o=csuh',
    # search scope (one|sub)
    SSCOPE => 'sub',
    # The following is valid but would only be necessary if the
    # anonymous user does NOT have permission to read from the LDAP tree
    UserDN => '',
    UserPw => '',
    # in case you want to add always one filter to each ldap query, use
    # this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
    '(objectclass=user)'
    AlwaysFilter => '',
    # if both your frontend and your LDAP are unicode, use this:
    SourceCharset => 'utf-8',
    DestCharset => 'utf-8',
    # if your frontend is e. g. iso-8859-1 and the character set of your
    # ldap server is utf-8, use these options:
    SourceCharset => 'utf-8',
    DestCharset => 'iso-8859-1',

    # Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
    Params => {
      port => 389,
      timeout => 120,
      async => 0,
      version => 3,
    },
  },
  # customer unique id
  CustomerKey => 'uid',
  # customer #
  CustomerID => 'mail',
  CustomerUserListFields => ['cn', 'mail'],
  CustomerUserSearchFields => ['uid', 'cn', 'mail'],
};

```

```

CustomerUserSearchPrefix => '',
CustomerUserSearchSuffix => '*',
CustomerUserSearchListLimit => 250,
CustomerUserPostMasterSearchFields => ['mail'],
CustomerUserNameFields => ['givenname', 'sn'],
# show not own tickets in customer panel, CompanyTickets
CustomerUserExcludePrimaryCustomerID => 0,
# add a ldap filter for valid users (expert setting)
# CustomerUserValidFilter => '!(description=locked)',
# admin can't change customer preferences
AdminSetPreferences => 0,
Map => [
  # note: Login, Email and CustomerID needed!
  # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly
  [ 'UserTitle',      'Title',      'title',          1, 0, 'var', '', 0 ],
  [ 'UserFirstname', 'Firstname', 'givenname',     1, 1, 'var', '', 0 ],
  [ 'UserLastname',  'Lastname',  'sn',           1, 1, 'var', '', 0 ],
  [ 'UserLogin',     'Username',  'uid',          1, 1, 'var', '', 0 ],
  [ 'UserEmail',    'Email',    'mail',         1, 1, 'var', '', 0 ],
  [ 'UserCustomerID', 'CustomerID', 'mail',        0, 1, 'var', '', 0 ],
#  [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1, 0, 'var', '', 0 ],
  [ 'UserPhone',    'Phone',    'telephonenumber', 1, 0, 'var', '', 0 ],
  [ 'UserAddress',  'Address',  'postaladdress',  1, 0, 'var', '', 0 ],
  [ 'UserComment',  'Comment',  'description',    1, 0, 'var', '', 0 ],
],
};

```

Есть возможность интегрировать до 10 различных пользовательских бэк-эндов. Используйте интерфейс управления клиентами в OTRS, чтобы просматривать и редактировать данные о них (при условии наличия прав для записи).

5.2.4. Сохранение информации о клиенте в динамических полях

Иногда бывает полезным также сохранить данные клиента заявки непосредственно в динамических полях заявки, для создания некоторых отчетов по этим данным. Например, по заявкам компании, когда клиент в течение отчетного периода перешел в другую компанию того же провайдера услуг техподдержки.

Динамические поля заполняются при создании заявки или когда клиент заявки изменяется. Значения полей берутся из данных клиента. Это работает для всех бэк-эндов, но наиболее полезно для LDAP-бэкэндов.

Для включения этой опции OTRS, активируйте параметр "Ticket::EventModulePost###930-DynamicFieldFromCustomerUser" и "DynamicFieldFromCustomerUser::Mapping". Последний параметр содержит настройку, какие поля данных клиента должны сохраняться в каждом из динамических полей. Поля должны быть заранее созданы в системе и включены в AgentTicketFreeText, так чтобы они были доступны для ручного заполнения. Они должны быть недоступны для ввода в AgentTicketPhone, AgentTicketEmail и AgentTicketCustomer, во избежание перекрытия автоматический установленных значений.

5.3. Хранилища (бэк-енды) для аутентификации Агентов и Клиентов

OTRS предлагает опцию для проверки подлинности агентов и клиентов с использованием различных хранилищ данных (бэкэндов).

5.3.1. Хранилища данных (бэк-енды) для аутентификации Агентов

5.3.1.1. База Данных (DB, по умолчанию)

В качестве бэк-энда для аутентификации агентов в OTRS, по умолчанию, используется база данных. Чтобы добавлять агентов, редактировать данные о них, перейдите на страницу Панель Администрирования и нажмите ссылку Интерфейс для управления агентами (см. ниже Пример 11.6).

Пример 4.11. Проверка подлинности агентов путем использования Базы Данных (DB) в качестве хранилища информации.

```
$Self->{'AuthModule'} = 'Kernel::System::Auth::DB';
```

5.3.1.2. LDAP

Если данные всех ваших агентов хранятся в LDAP-директории, то для аутентификации пользователей в OTRS можно использовать LDAP-модуль (см. ниже Пример 11-7). Этот модуль имеет права только на чтение дерева LDAP-каталогов, что означает что нету возможности редактировать данные пользователей используя веб-интерфейс для управления пользователями .

Пример 4.12. Проверка подлинности агентов при использовании LDAP в качестве хранилища данных

```
# This is an example configuration for an LDAP auth. backend.
# (Make sure Net::LDAP is installed!)
$Self->{'AuthModule'} = 'Kernel::System::Auth::LDAP';
$Self->{'AuthModule::LDAP::Host'} = 'ldap.example.com';
$Self->{'AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$Self->{'AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$Self->{'AuthModule::LDAP::GroupDN'} = 'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$Self->{'AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (with full user dn)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user do NOT have permission to read from the LDAP tree
$Self->{'AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'AuthModule::LDAP::SearchUserPw'} = '';

# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter => '(objectclass=user)'
$Self->{'AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.
# $Self->{'AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
$Self->{'AuthModule::LDAP::Params'} = {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
```



```
};
```

Параметры конфигурации приведенные в сценарии приведенном ниже могут быть использованы для синхронизации пользовательских данных из LDAP-каталога в локальную базу данных OTRS. Это уменьшает количество запросов к LDAP-серверу и ускоряет проверку подлинности в OTRS. Синхронизация данных завершена, когда агент аутентифицируется впервые. Несмотря на то, что данные могут быть синхронизированы в локальной базе данных OTRS, LDAP-каталог является последней инстанцией для проверки подлинности пользователей, таким образом пользователи, не активные для дерева каталогов LDAP не могут пройти аутентификацию в OTRS, даже в том случае, когда данные о пользователе уже хранятся в базе данных OTRS. Данные агентов в LDAP-директории не могут быть отредактированы с помощью веб-интерфейса OTRS, так что управление данными должно осуществляться непосредственно в дереве каталогов LDAP.

```
# defines AuthSyncBackend (AuthSyncModule) for AuthModule
# if this key exists and is empty, there won't be a sync.
# example values: AuthSyncBackend, AuthSyncBackend2
$self->{'AuthModule::UseSyncBackend'} = 'AuthSyncBackend';

# agent data sync against ldap
$self->{'AuthSyncModule'} = 'Kernel::System::Auth::Sync::LDAP';
$self->{'AuthSyncModule::LDAP::Host'} = 'ldap://ldap.example.com/';
$self->{'AuthSyncModule::LDAP::BaseDN'} = 'dc=otrs, dc=org';
$self->{'AuthSyncModule::LDAP::UID'} = 'uid';
$self->{'AuthSyncModule::LDAP::SearchUserDN'} = 'uid=sys, ou=user, dc=otrs, dc=org';
$self->{'AuthSyncModule::LDAP::SearchUserPw'} = 'some_pass';
$self->{'AuthSyncModule::LDAP::UserSyncMap'} = {
    # DB -> LDAP
    UserFirstname => 'givenName',
    UserLastname  => 'sn',
    UserEmail     => 'mail',
};
[...];

# AuthSyncModule::LDAP::UserSyncInitialGroups
# (sync following group with rw permission after initial create of first agent
# login)
$self->{'AuthSyncModule::LDAP::UserSyncInitialGroups'} = [
    'users',
];
```

Сценарий: Синхронизация пользовательских данных из LDAP-директории в базу данных OTRS.

5.3.1.3. HTTPBasicAuth-аутентификация для Агентов

Если вы хотите реализовать решение "single sign on" для всех агентов, вы можете использовать базовую аутентификацию (для всех систем) и HTTPBasicAuth-модуль для OTRS (см. ниже Пример 11-8).

Пример 4.13. Аутентификация Агентов с помощью HTTPBasic

```
# This is an example configuration for an apache ($ENV{REMOTE_USER})
# auth. backend. Use it if you want to have a single login through
# apache http-basic-auth
$self->{'AuthModule'} = 'Kernel::System::Auth::HTTPBasicAuth';

# Note:
#
# If you use this module, you should use as fallback
# the following configuration settings if the user is not authorized
# apache ($ENV{REMOTE_USER})
$self->{'LoginURL'} = 'http://host.example.com/not-authorized-for-otrs.html';
$self->{'LogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';
```

5.3.1.4. Radius

Параметры конфигурации приведенные в Примере 11-9 могут быть использованы для аутентификации агентов с использованием Radius-сервера.

Пример 4.14. Аутентификация (проверка подлинности) агентов с использованием Radius-сервера в качестве хранилища информации

```
# This is example configuration to auth. agents against a radius server
$self->{'AuthModule'} = 'Kernel::System::Auth::Radius';
$self->{'AuthModule::Radius::Host'} = 'radiushost';
$self->{'AuthModule::Radius::Password'} = 'radiussecret';
```

5.3.2. Хранилища информации для аутентификации (проверки подлинности) клиентов

5.3.2.1. База Данных (По умолчанию)

Для аутентификации клиентов в OTRS, по умолчанию, используется база данных. Используя базу данных в качестве хранилища, все данные клиентов можно редактировать через веб-интерфейс OTRS (см. ниже Пример 11-10).

Пример 4.15. Аутентификация Клиента в Базе Данных

```
# This is the auth. module against the otrs db
$self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';
$self->{'Customer::AuthModule::DB::Table'} = 'customer_user';
$self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';
$self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';
$self->{'Customer::AuthModule::DB::DSN'} =
  "DBI:mysql:database=customerdb;host=customerdbhost";
$self->{'Customer::AuthModule::DB::User'} = "some_user";
$self->{'Customer::AuthModule::DB::Password'} = "some_password";
```

5.3.2.2. LDAP

Если у вас есть LDAP-каталог со всеми данными о клиентах, можно использовать модуль LDAP для аутентификации клиентов в OTRS (см. Пример 11-11 ниже). Поскольку этот модуль имеет права только для чтения для данных из LDAP-бэкенда, то нету возможности изменить данные клиента через веб.

Пример 4.16. Аутентификация пользователей с помощью LDAP-бэкенда

```
# This is an example configuration for an LDAP auth. backend.
# (make sure Net::LDAP is installed)
$self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::LDAP';
$self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';
$self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$self->{'Customer::AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$self->{'Customer::AuthModule::LDAP::GroupDN'} =
  'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
$self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (full user dn)
```

```

#$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user does NOT have permission to read from the LDAP tree
$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = '';

# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter => '(objectclass=user)'
$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each customer login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.
#$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
$Self->{'Customer::AuthModule::LDAP::Params'} = {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
};

```

5.3.2.3. HTTPBasicAuth аутентификация для клиентов

Если для пользователей нужно внедрить "single sign on"-решение, можно использовать базовую, HTTPBasic аутентификацию (для всех систем) и использовать модуль HTTPBasicAuth с OTRS (больше не нужно логиниться в OTRS). См. ниже Пример 11-12.

Пример 4.17. Аутентификация клиентов с помощью HTTPBasic

```

# This is an example configuration for an apache ($ENV{REMOTE_USER})
# auth. backend. Use it if you want to have a single login through
# apache http-basic-auth
$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::HTTPBasicAuth';

# Note:
# If you use this module, you should use the following
# config settings as fallback, if user isn't login through
# apache ($ENV{REMOTE_USER})
$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/not-authorized-for-otrs.html';
$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';

```

5.3.2.4. Radius

Настройки приведенные в примере 11-13 могут быть использованы для аутентификации ваших клиентов с помощью Radius-сервера.

Пример 4.18. Аутентификация клиентов с использованием Radius

```

# This is a example configuration to auth. customer against a radius server
$Self->{'Customer::AuthModule'} = 'Kernel::System::Auth::Radius';
$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';
$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';

```

5.4. Настройка самостоятельной регистрации клиента

Есть возможность настроить самостоятельную регистрацию новых клиентов, доступную через панель customer.pl. Можно добавить новые дополнительные или обязательные поля, такие как номер комнаты, адрес или состояние.

В следующем примере показано каким образом можно указать обязательные для заполнения поля в базе данных клиентов, в данном случае, для хранения номера клиента.

5.4.1. Настройка веб-интерфейса

Для отображения нового поля для номера комнаты в веб-интерфейсе пользователя (customer.pl), нужно внести изменения в .dtl-файл, который отвечает за шаблон этого интерфейса. Отредактируйте файл Kernel/Output/HTML/Standard/CustomerLogin.dtl добавив новые поля после строки 80 (см. нижеприведенный сценарий).

```
[...]
<div class="NewLine">
  <label for="Room">$Text{"Room{CustomerUser}"</label>
  <input title="$Text{"Room Number"}" name="Room" type="text" id="UserRoom"
  maxlength="50" />
</div>
[...]
```

Сценарий: Отображение новых полей в веб-интерфейсе.

5.4.2. Отображения клиентов

На следующем этапе, нужно внести изменения в маппинг (mapping), расширив его новым элементом - номером комнаты. Чтобы убедиться что изменения не потеряются после обновления, установите настройки "CustomerUser" из файла Kernel/Config/Defaults.pm в файл Kernel/Config.pm. Теперь измените MAP-массив, добавив новое поле - номер комнаты, как это показано в нижеприведенном сценарии.

```
# CustomerUser
# (customer database backend and settings)
$self->{CustomerUser} = {
  Name => 'Database Backend',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the
    # required settings
    DSN => 'DBI:odbc:yourdsn',
    Type => 'mssql', # only for ODBC connections
    DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
    User => '',
    Password => '',
    Table => 'customer_user',
  },
  # customer unique id
  CustomerKey => 'login',
  # customer #
  CustomerID => 'customer_id',
  CustomerValid => 'valid_id',
  CustomerUserListFields => ['first_name', 'last_name', 'email'],
  # CustomerUserListFields => ['login', 'first_name', 'last_name', 'customer_id', 'email'],
  CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
  CustomerUserSearchPrefix => '',
  CustomerUserSearchSuffix => '*',
  CustomerUserSearchListLimit => 250,
  CustomerUserPostMasterSearchFields => ['email'],
  CustomerUserNameFields => ['title', 'first_name', 'last_name'],
  CustomerUserEmailUniqCheck => 1,
  # show not own tickets in customer panel, CompanyTickets
  # CustomerUserExcludePrimaryCustomerID => 0,
  # generate auto logins
  # AutoLoginCreation => 0,
  # AutoLoginCreationPrefix => 'auto',
  # admin can change customer preferences
  # AdminSetPreferences => 1,
```

```
# # cache time to live in sec. - cache database queries
# CacheTTL => 0,
# # just a read only source
# ReadOnly => 1,
# Map => [
#
# note: Login, Email and CustomerID needed!
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target
[ 'UserTitle',      'Title',      'title',      1, 0, 'var', '', 0 ],
[ 'UserFirstname', 'Firstname',  'first_name', 1, 1, 'var', '', 0 ],
[ 'UserLastname',  'Lastname',   'last_name',  1, 1, 'var', '', 0 ],
[ 'UserLogin',     'Username',   'login',      1, 1, 'var', '', 0 ],
[ 'UserPassword',  'Password',   'pw',         0, 0, 'var', '', 0 ],
[ 'UserEmail',     'Email',      'email',      1, 1, 'var', '', 0 ],
[ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
[ 'UserPhone',     'Phone',      'phone',      1, 0, 'var', '', 0 ],
[ 'UserFax',       'Fax',        'fax',        1, 0, 'var', '', 0 ],
[ 'UserMobile',    'Mobile',     'mobile',     1, 0, 'var', '', 0 ],
[ 'UserRoom',      'Room',       'room',       1, 0, 'var', '', 0 ],
[ 'UserStreet',    'Street',     'street',     1, 0, 'var', '', 0 ],
[ 'UserZip',       'Zip',        'zip',        1, 0, 'var', '', 0 ],
[ 'UserCity',      'City',       'city',       1, 0, 'var', '', 0 ],
[ 'UserCountry',   'Country',    'country',    1, 0, 'var', '', 0 ],
[ 'UserComment',   'Comment',    'comments',   1, 0, 'var', '', 0 ],
[ 'ValidID',       'Valid',      'valid_id',   0, 1, 'int', '', 0 ],
],
# default selections
Selections => {
  UserTitle => {
    'Mr.' => 'Mr.',
    'Mrs.' => 'Mrs.',
  },
},
};
```

Сценарий: Внесение изменений в тар-массив.

5.4.3. Настройка таблицы customer_user в Базе Данных OTRS DB

Последним шагом является добавление нового столбца с номером комнаты в таблицу customer_user базы данных OTRS (см. нижеприведенный сценарий). В этой колонке будет храниться информация для номера комнаты.

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (200);
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

Сценарий: Добавление новой колонки в таблицу customer_user.

Теперь новое поле для номера комнаты должно отображаться в панели информации о клиенте и экранах управления клиентами. Когда новые клиенты будут регистри-

ровать аккаунт, им нужно будет вписать номер комнаты. Если для работы OTRS используется Microsoft IIS, то чтобы новые изменения вступили в силу в Config.pm нужно перезапустить веб-сервер.

6. Настройки Заявки

6.1. Состояния заявок

6.1.1. Предопределенные состояния

OTRS позволяет изменять предопределенные состояния заявок и их типы, а также добавлять новые. Для состояния важны два атрибута: имя (state-name) и тип (state-type).

Предустановленные состояния в OTRS: "закрыто успешно", "закрыто неудачно", "объединено", "новая", "открытая", "в ожидании с автозакрытием+", "в ожидании с автозакрытием-", "в ожидании с напоминанием", "удаленная".

6.1.1.1. Новая

Заявки находятся в этом состоянии, когда они создаются на основе входящих сообщений электронной почты.

6.1.1.2. Открытая

Это состояние по умолчанию для заявок, которые присвоены очередям или агентам.

6.1.1.3. Ожидание с напоминанием

После того как время ожидания истекло, владелец заявки будет получать напоминание на электронную почту. Если заявка не закрыта, то напоминание о заявке будет отправлено всем агентам в очереди. Напоминание о заявках будет отправлено только в рабочее время и будет повторяться каждые 24-часа, пока агент не изменит состояние заявки. Время, которое заявка проведет с таким статусом будет добавлено к времени эскалации.

6.1.1.4. Ожидание авто-закрытие -

Если время ожидания вышло, заявки с этим статусом будут установлены в "Закрытые неуспешно". Время, проведенное заявкой в этом статусе будет добавлено к времени эскалации.

6.1.1.5. Ожидание авто-закрытие+

Если вышло время ожидания, заявки с этим статусом будут установлены в "Закрыто Успешно". Время, проведенное заявкой в этом статусе будет добавлено к времени эскалации.

6.1.1.6. Объединенные

Это состояние для заявок, которые были объединены с другими заявками.

6.1.1.7. Закрыта Успешно

Это конечное состояние для заявок, которые были решены успешно. В зависимости от конфигурации, у вас будет или не будет возможности заново открыть ранее закрытые заявки.

6.1.1.8. Закрыта Неудачно

Это конечное состояние для заявок которые НЕ были успешно решены. В зависимости от конфигурации, у вас будет или не будет возможности заново открыть ранее закрытые заявки.

6.1.2. Настраиваемы состояния

Каждое состояние имеет название (state-name) и тип (state-type). Чтобы создать новое состояние перейдите по ссылке Состояния на Панели Администрирования и нажмите кнопку "Добавить состояние". Можно свободно выбирать имя нового состояния. Типы состояний не могут изменяться посредством веб-интерфейса. Если нужно добавить новые типы или изменить существующие, - все изменения придется делать напрямую в базе данных. Предусмотренные типы состояний не могут быть изменены, поскольку это может привести к непредсказуемым результатам. Например, расчет эскалации и функция разблокирования основаны на конкретных типах состояний.

Имя/название уже существующего состояния может быть изменено или новые состояния добавлены на этом экране. Если состояние "new" было изменено в веб-интерфейсе, это состояние также должно быть настроено в Kernel/Config.pm или через SysConfig. Параметры заданные в сценарии приведенном ниже должны быть изменены, чтобы убедиться в том, что OTRS работает с измененным состоянием для бывшего "new".

```
[...]  
# PostmasterDefaultState  
# (The default state of new tickets.) [default: new]  
$Self->{PostmasterDefaultState} = 'new';  
  
# CustomerDefaultState  
# (default state of new customer tickets)  
$Self->{CustomerDefaultState} = 'new';  
[...]
```

Сценарий: Изменение параметров настройки в Kernel/Config.pm.

Если нужно добавить новый тип состояния, то это можно сделать с помощью клиентской программы управления базами данных, изменив таблицу ticket_state_type базы данных OTRS, как это показано в нижеприведенном сценарии

```
linux:~# mysql -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 23 to server version: 5.0.16-Debian_1-log  
  
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.  
  
mysql> use otrs;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> insert into ticket_state_type (name,comments) values ('own','Own  
state type');  
Query OK, 1 row affected (0.00 sec)  
  
mysql> quit  
Bye  
linux:~#
```

Script: Изменение базы данных OTRS.

На данный момент можно использовать новый тип состояния, который вы только что создали. Как только состояние будет связано с этим новым типом состояния, то чтобы убедиться что новое состояние используется и работает корректно нужно также изменить настройки OTRS. Используя SysConfig внесите изменения в следующие опции:

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateDefault - задает следующее состояние по умолчанию для новых заявок созданных на основе телефонного звонка.

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateType - задает список последующих доступных состояний для новых заявок, созданных на основе телефонного звонка.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateDefault - задает следующее состояние по умолчанию для новых заявок созданных на основе сообщений электронной почты.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateType - задает список последующих доступных состояний для новых заявок созданных на базе сообщений электронной почты.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###State - задает значение по умолчанию следующего состояния заявки при регистрации новых сообщений в заявке, полученных от клиента по телефону.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###StateType - задает доступные значения следующего состояния заявки при регистрации новых сообщений в заявке, полученных от клиента по телефону.

Ticket -> Frontend::Agent::Ticket::ViewMove > AgentTicketMove###State - задает следующее состояние по умолчанию при перемещении заявки в другую очередь.

Ticket -> Frontend::Agent::Ticket::ViewMove > AgentTicketMove###StateType - задает список последующих доступных состояний при перемещении заявки в другую очередь.

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateDefault - to define the default next state after bouncing a ticket.

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateType - to define the available next states in the bounce screen.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateDefault - для указания следующего состояния по умолчанию при массовом действии.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateType - задает список последующих доступных состояний при массовом действии.

Ticket -> Frontend::Agent::Ticket::ViewClose > StateDefault - для указания следующего состояния по умолчанию после закрытия заявки.

Ticket -> Frontend::Agent::Ticket::ViewClose > StateType - задает список последующих доступных состояний после закрытия заявки.

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateDefault - задает значение по умолчанию следующего состояния заявки на странице/экране создания сообщения (ответа) агентом.

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateType - задает список доступных следующих состояний заявки на странице/экране создания сообщения (ответа) агентом.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - to define the default next state after forwarding a ticket.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - to define the available next states in the Forward screen.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - задает следующее состояние по умолчанию после редактирования свободных полей заявки

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - задает список доступных состояний после редактирования свободных полей заявки

Ticket -> Core::PostMaster > PostmasterDefaultState - задает состояние заявки созданной из электронного письма.

Ticket -> Core::PostMaster > PostmasterFollowUpState - задает состояние заявки после получения отклика клиента.

Ticket -> Core::PostMaster > PostmasterFollowUpStateClosed - задает состояние заявок после получения отклика клиента на уже закрытую заявку.

Ticket -> Core::Ticket > ViewableStateType - задает типы состояний заявок, которые отображаются в различных обзорах системы, например, в Обзоре очередей.

Ticket -> Core::Ticket > UnlockStateType - задает список типов состояний для разблокированных заявок.

Ticket -> Core::Ticket > PendingReminderStateType - задает тип состояния для заявок с напоминанием.

Ticket -> Core::Ticket > PendingAutoStateType - задает тип состояния для отложенных заявок.

Ticket -> Core::Ticket > StateAfterPending - задает тип состояния для заявок с автоожиданием после истечения срока ожидания.

6.2. Приоритеты заявок

OTRS поставляется с пятью предустановленными уровнями приоритетов, которые можно изменить перейдя по ссылке "Приоритеты" на Панели Администрирования. При создании настраиваемого списка приоритетов, пожалуйста помните, что они сортируются в алфавитном порядке. Также OTRS сортирует заявки в QueueView по их внутреннему номеру (ID).

Примечание

Как и в случае с другими сущностями OTRS, приоритеты не могут быть удалены, а только деактивированы путем установки параметра Действительный в значение *не действительный* или *не действительный-временно*.

Важно

Если был создан новый приоритет, или был изменен уже существующий, то можно также произвести изменения некоторых параметров в SysConfig:

- Ticket:Core::Postmaster::PostmasterDefaultPriority - задает предустановленный приоритет для всех входящих сообщений электронной почты.

- Ticket:Frontend::Agent:Ticket::ViewPhoneNew:Priority - задает приоритет по умолчанию для новых Заявок созданных на основе Телефонных Звонков.
- Ticket:Frontend::Agent:Ticket::ViewEmailNew:Priority - задает приоритет по умолчанию на странице создания новой Email-Заявки в интерфейсе агента.
- Ticket:Frontend::Customer:Ticket::ViewNew:PriorityDefault - задает приоритет по умолчанию при создании заявки клиентом в интерфейсе клиента.

6.3. Ответственность за Заявку & Наблюдение за Заявкой

Начиная с OTRS 2.1 и выше, в дополнение к владельцу заявки можно определить ответственного за нее агента. Кроме того, все мероприятия, связанные с заявкой могут просматриваться не только владельцем но другими людьми. Эти две возможности системы реализованы с помощью функций TicketResponsible и TicketWatcher и также позволяют работать в рамках иерархической структуры команды.

6.3.1. Ответственность за Заявку

Функция ответственности за заявку способствует ее полной обработки еагентом, который не является владельцем заявки. Таким образом агент, который заблокировал заявку может передать ее другому агенту, который не является владельцем заявки, для того, чтобы второй просто дал ответ на вопрос. После того как запрос был рассмотрен, первый агент может снять ответственность за заявку с второго агента.

С помощью параметров конфигурацииTicket::Responsible, можно активировать функцию ответственности за заявку. Это приведет к появлению в верхней части экрана 3-х дополнительных иконок/значков.

Ответственность за заявку может быть назначена после открытия ее содержимого, нажав ссылку "Ответственность", соответствующего меню в шаблоне просмотра подробной информации о заявке агентского веб-интерфейса (см. нижеприведенный сценарий).

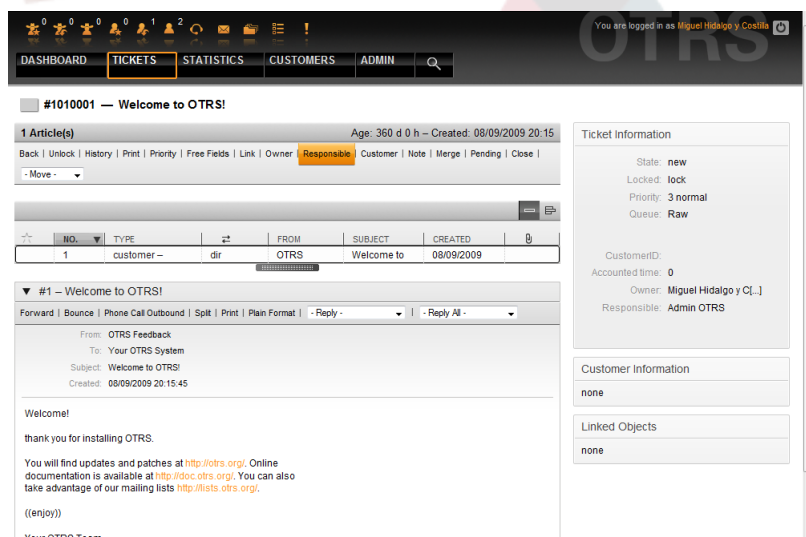


Рисунок: Изменение Ответственного за заявку в шаблоне просмотра подробной инф. о заявке.

После нажатия на кнопку "Ответственность", откроется всплывающее окно для изменения ответственности этой заявки (см. нижеприведенный Рисунок). Этот шаб-

лон/диалог также может быть использован для отправки сообщения новому агенту, который будет нести ответственность за эту заявку.

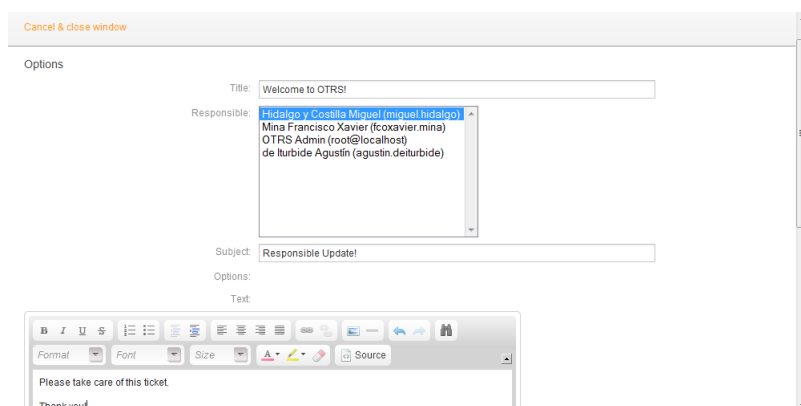


Рисунок: Всплывающий диалог для изменения ответственного за заявку.

Если активирована функция ответственности за заявку, то в шаблоне Ответственность, агентского веб-интерфейса OTRS можно просмотреть список всех заявок, за которые агент несет ответственность.

6.3.2. Просмотр Заявок

Начиная с OTRS 2.1 и выше с помощью функции TicketWatcher, выбранные агенты, такие как, например, руководители могут просматривать определенные заявки без их обработки.

Функция TicketWatcher может быть активирована с помощью параметра Ticket::Watcher, после чего в панели инструментов появятся новые ссылки/иконки. Используя Ticket::WatcherGroup можно определить одну или несколько групп пользователей с правами просмотра/наблюдения заявок.

Для того чтобы смотреть заявку, перейдите к шаблону просмотра подробной информации о заявке и нажмите ссылку "Подписаться" в меню заявки (см. нижеприведенный Рисунок).

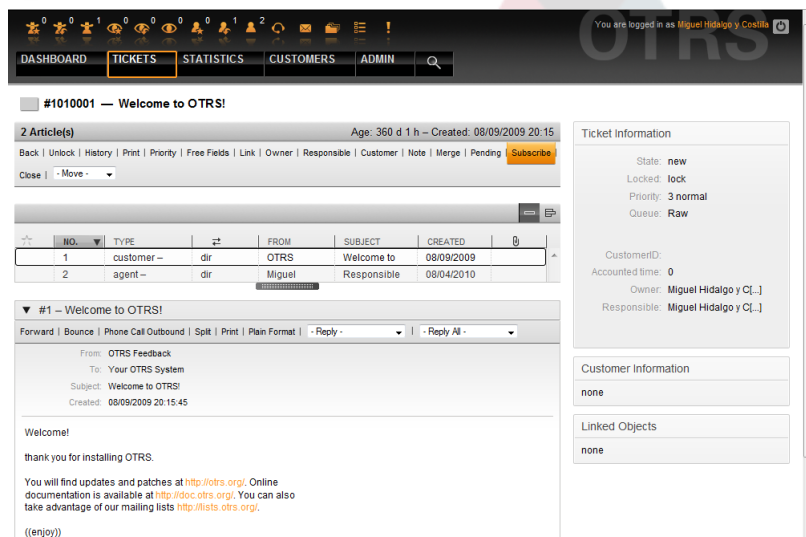


Figure: Подписка на просмотр подробной информации о заявке.

Если вы больше не хотите просматривать определенную заявку, перейдите к шаблону просмотра подробной информации заявки и нажмите кнопку "Отменить подписку" в меню заявки (см. нижеприведенный рисунок).

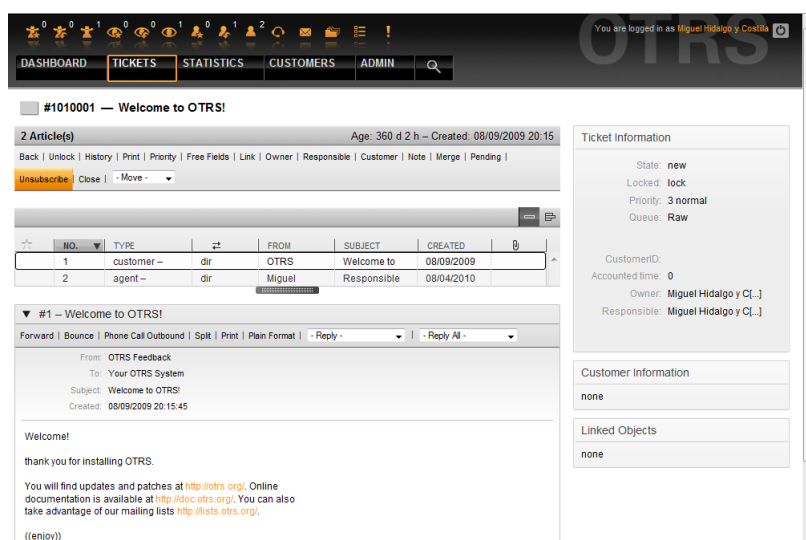


Рисунок: Отказ от подписи на заявку в шаблоне просмотра подробной информации о заявке.

Как только активирована функция "просмотр заявок", то список всех просматриваемых заявок будет доступен в шаблоне Watched View (см. нижеприведенный Рисунок).

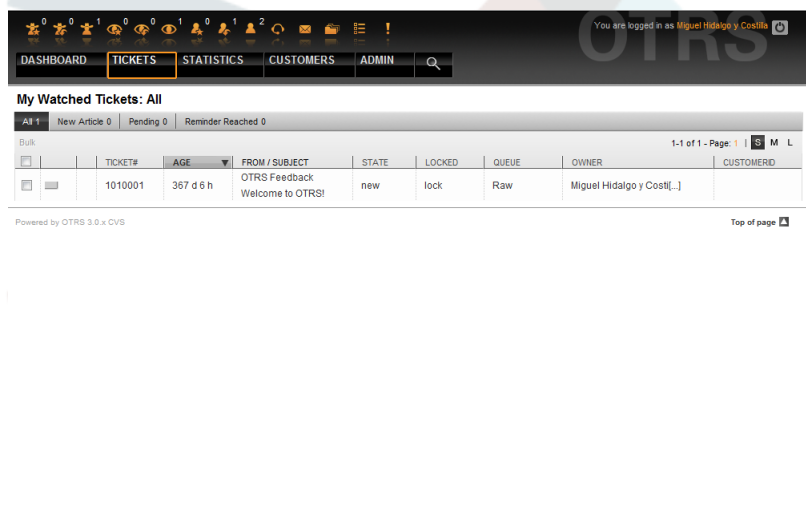


Рисунок: Шаблон для просматриваемых заявок.

7. Функции связанные с временем

7.1. Настройка периода рабочего времени, праздников и часовых поясов

Некоторые функции в OTRS, например, как эскалация и автоматическая разблокировка заявок зависят от правильной настройки параметра "рабочее время", часовых поясов и праздников. Все эти настройки можно определить перейдя по ссылке интерфейс SysConfig в Framework > Core::Time. Перейдя по ссылке Framework > Core::Time::Calendar1 через Framework > Core::Time::Calendar9 можно также указать различные, заранее установленные "наборы рабочего времени", праздников и часовых поясов. Перейдя по ссылке настройки очереди можно определить Календари.

Для этого также можно использовать SLA. Например можно создать различные SLA-схемы: 5 x 8 рабочих часов для стандартного SLA, но и вместе с тем создать другой календарь: 7 x 24 - "Золотой SLA"; также можно иметь отдельный календарь для очереди 'Support-USA', с часовыми поясами, различающимися с очередью 'Support-Japan'. В OTRS можно создавать до 99 различных календарей.

7.1.1. Рабочее время

Задаёт рабочие часы в вашей системе в SysConfig Framework > Core::Time::TimeWorkingHours, или ваши специальные календари в настройке календарей. OTRS манипулирует отрезками времени в один час. Пометка рабочих часов 8, 9, 10 ... 17 соответствует диапазону времени с 8 AM - 6 PM.

Заявки эскалируются, уведомления для эскалированных и ожидающих заявок отправляются и заявки разблокируются только в рабочее время.

7.1.2. Праздники с фиксированными датами

Праздники, которые с каждым годом не приходятся на фиксированный день, такие как Новый год или четвертое Июля могут быть указаны в Отпускные дни, или в соответствующем разделе календаря 1-9.

Заявки не будут эскалироваться, ни разблокироваться в сроки, определенные как "Специально зарезервированные дни".

Примечание

По умолчанию, OTRS работает по *Немецким* выходным.

7.1.3. TimeVacationDaysOneTime

Праздники, такие как Пасха, которые ежегодно не имеют фиксированной даты могут быть указаны в Отпускные дни/One-Time .

Заявки не будут эскалироваться и не будут разблокироваться в период времени определенный как TimeVacationDaysOneTime.

Примечание

OTRS поставляется без единого предустановленного One-Time-праздника. Это означает, что во время настройки системы OTRS, нужно самостоятельно добавить праздники, такие как Пасха, День Благодарения.

7.2. Автоматические Разблокировки

Заблокированные заявки могут быть разблокированы системой автоматически. Эта опция может быть полезной, если, например, агент заблокировал заявки, которые нужно обработать, но не работал с ними по определенной причине, например был в отпуске. Чтобы убедиться в том, что заблокированные заявки не будут забыты и их обработают другие агенты, автоматическая разблокировка разблокирует заблокированные заявки при истечении определенного времени.

Перейдя по ссылке настройки очереди для каждой заявки можно установить количество времени, при истечении которого она будет разблокирована. Модуль, bin/otrs.UnlockTickets.pl, который периодически вызывается как задание планировщика задач cron job, выполняет автоматическую разблокировку заявок.

Уведомления о разблокированных заявках отсылаются только тем агентам, которые добавили очередь с разблокированными заявками в "Мои очереди", и активировали в своих настройках функцию уведомлений о разблокированных заявках.

Заявки будут заблокированы в том случае, если выполняются следующие условия:

- Для очереди в которой хранится заявка определяется *время разблокировки* .
- Заявке присвоен статус *заблокирована*.
- Заявке присвоен статус *открытая*.

Если агент добавляет в заявку новую статью - таймер разблокировки будет сброшен. Он может быть любым из следующих типов: *внешний-email, телефон, факс, смс, или внешнее примечание*.

Кроме этого, если последняя статья в заявке создана агентом, и клиент добавляет еще одну, либо через веб-интерфейс либо по электронной почте, таймер разблокировки также будет сброшен.

Последнее событие, которое будет сбрасывать таймер разблокировки, когда заявка передается другому агенту.

8. Настройка вывода в формате PDF

В этом разделе приводится настройка опций для экспорта данных из OTRS в PDF.

При использовании функции Печать в любом месте интерфейса OTRS будет создаваться файл в формате PDF. Вы можете отключить это изменением параметра PDF для вывода в HTML формате.

Чтобы изменить внешний вид файлов, сгенерированных системой OTRS, можно создать свой собственный логотип и добавить его перейдя по ссылке PDF::LogoFile. Чтобы определить стандартный размер страниц генерируемых pdf-файлов (DIN-A4 или Letter) воспользуйтесь ссылкой PDF::PageSize. Чтобы указать максимальное количество страниц pdf-файла используйте ссылку PDF::MaxPage. Эта настройка может быть полезна когда пользователь ошибочно генерирует файл большого размера.

Для генерации pdf-файлов должны быть проинсталированы Perl-модули CPAN: PDF::API2 и Compress::Zlib. В большинстве дистрибутивов они доступны в виде пакетов и могут быть легко установлены с помощью соответствующего менеджера пакетов. В случае, если это не возможно, они должны быть установлены с помощью CPAN. За дополнительной информацией об установке Perl-модулей, обратитесь к главе "Установка Perl-модулей".

9. Stats-модуль

Модуль отчетов OTRS дает возможность оперативно отслеживать статистику и создавать пользовательские отчеты, связанные с использованием OTRS. OTRS использует общепотребляемый термин "stat" для обозначения отчетов с различными показателями.

Правильная настройка модуля статистики OTRS связана с множеством различных условий. К ним относятся различные модули системы OTRS, подлежащие оценке, настройки прав доступа пользователей, признаки, которые должны быть рассчитаны и уровень их сложности, легкость настройки модуля статистики, скорость и эффективность вычислений, поддержка множества вариантов вывода информации.

Статистические элементы, такие как например файлы, которые дополняют функциональные возможности модуля статистики для конкретных требований, могут быть интегрированы для подсчета сложной статистики.

9.1. Обработка модуля агентом

Когда вы вошли в систему как агент, в меню интерфейса будет отображаться пункт "Отчеты". Этот раздел имеет множество различных подменю, как это показано на Рисунке.

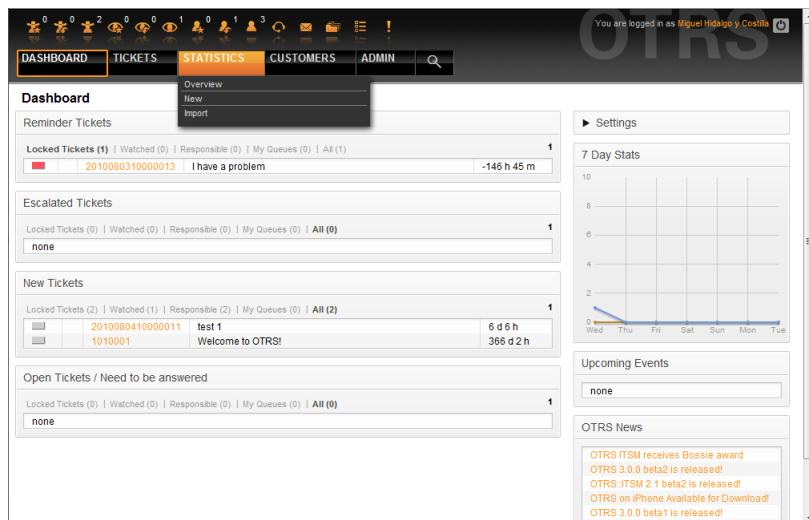


Рисунок: Опции меню Статистика.

Различные варианты представленные в меню статистики:

- *Overview*. Предоставляет список различных предварительно сконфигурированных отчетов.
- *Новая*. Требуется права gw.
- *Импорт*. Требуется gw-права.

9.1.1. Обзор

Выберите ссылку "Отчеты" в меню, а затем подменю "Обзор". На открывшейся странице будет представлен список всех предварительно настроенных отчетов, которые агент уже может использовать (см. Рисунок ниже).

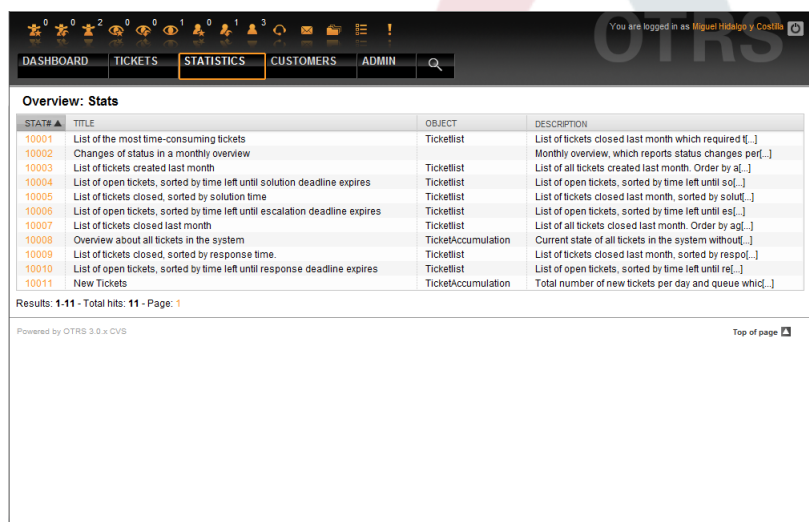


Рисунок: Просмотр стандартных запросов.

Для каждого из отчетов, перечисленных в Обзоре предоставлена следующая информация:

- **Stat#.** Уникальный номер отчета.
- **Заголовок.** Заголовок отчета.
- **Объект.** Объект, который используется для генерации статистики. В случае статической статистики не отображается ни один объект, поскольку в ее генерации динамические объекты не участвуют.
- **Описание.** Краткое описание отчета.

Когда модуль статистики установлен, он поставляется с несколькими предустановленными отчетами, выборочно встроенными в систему. Они приведены в виде списка на странице "Обзор". Если список слишком большой и не помещается на одну страницу, у агента есть возможность просматривать другие страницы. Список отчетов может быть отсортирован по своему вкусу, путем нажатия на желаемый заголовок столбца в списке. Чтобы создать конкретный отчет, нажмите на соответствующие stat-номер, связанный с отчетом. В результате откроется интерфейс просмотра отчета.

9.1.2. Создание и просмотр отчетов

Пользовательский интерфейс предоставляет параметры настройки отчетов (см. Рисунок ниже).

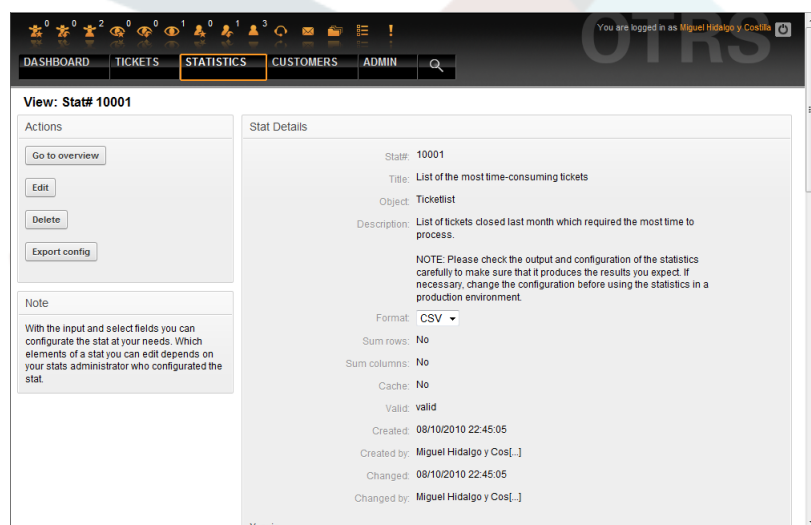


Рисунок: Просмотр определенного отчета.

Параметры конфигурации для конкретного отчета могут быть установлены путем выбора ряда возможных опций. Как автор отчета так и другие пользователи с соответствующими правами доступа могут создавать настройки.

На странице отображается следующее:

- **Возможные действия:**
 - **Перейти к обзору.** Ссылка, ведущая обратно к Просмотру списка отчетов.
 - **Редактировать.** Редактировать имеющуюся структуру отчета (требуется права для чтения и записи, rw).
 - **Удалить.** Удалить текущий отчет (требуется права для чтения и записи, rw).

- *Экспортировать настройки.* Экспортировать настройки отчета с помощью загрузки файла (требуется права для чтения и записи, rw).

Традиционно: функции Экспорт и Импорт позволяют создавать и тестировать отчеты та тестовых системах с их легкой дальнейшей интеграцией в производственную систему.

- **Подробности отчета:**

- *Отчет#.* Номер отчета.
- *Заголовок.* Заголовок отчета.
- *Объект.* Объект, который используется для создания отчета.
- *Описание.* Описание цели отчета.
- *Формат.* Выходной формат отчета. В зависимости от конфигурации, можно выбрать один из следующих форматов вывода:
 - CSV.
 - Печать.
 - Графические линии.
 - Столбцовая диаграмма.
 - Диаграмма.
 - Точечная диаграмма.
 - Точечно-линейная диаграмма.
 - Диаграмма
 - Круговая "пирожок"-диаграмма.
- *Размер в пикселях.* Размер графиков/диаграмм в пикселях. Эта опция доступна только в том случае, если в настройках отчета разрешены графики. Все часто используемые графические размеры настраиваются OTRS-администратором в SysConfig. Затем, во время настройки отчета, агент может выбрать необходимый формат из всех доступных.
- *Количество строк.* Указывает когда отчет заменяется колонкой, чьи ячейки содержат сумму соответствующих строк.
- *Количество колонок.* Указывает когда отчет заменяется строкой, чьи ячейки содержат сумму соответствующих колонок.
- *Кэш.* Указывает когда сгенерированный отчет кэшируется в файловой системе.
- *Действительный.* Может быть установлен в значение "недействительный" если по какой то причине отчет временно не должен запускаться. Кнопка "Старт" в нижней части правой панели больше не отображается. Таким образом отчет не может быть создан.
- *Создан.* Время создания отчета.
- *Создан (кем).* Имя агента, который создал отчет.

- *Изменен.* Время последней модификации отчета.
- *Изменен кем.* Имя агента, который последним внес изменения в отчет.
- *X-axis.* Используя эту функцию, агент может переключать x и y axes (только в том случае, если эта опция активирована OTRS-администратором).
- Общая информация сопровождается информацией о самом отчете. Есть два различных шаблона просмотра отчета (или статистики):
 - *Static stat view.* Static report generators can be integrated into the stats module (see Figure below).

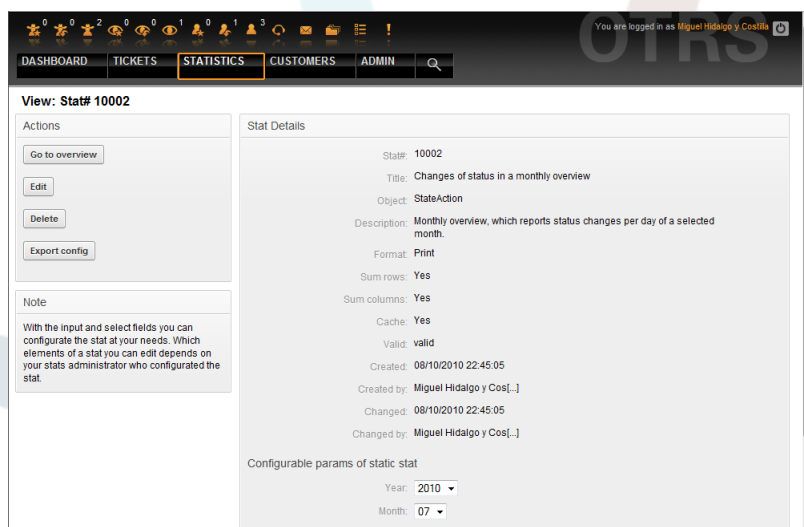


Figure: Viewing a static report.

- *Dynamic stat view* (see Figure above). They can be displayed in two different ways:
 - *Неизменяемые настройки.* Создатель отчета не имеет прав для изменения этих полей.
 - *Изменяемые настройки.* Параметры конфигурации таких отчетов могут быть изменены агентом.

Нажатие кнопки "Пуск" (в нижней части экрана) является последним шагом при создании отчета. Есть две возможные причины, когда эта кнопка не будет отображаться:

1. Отчет был установлен недействительным и, таким образом отключен.
2. Отчет был настроен не аккуратно и, следовательно, не может выполняться. В этом случае необходимую информацию можно найти в разделе уведомлений OTRS (внизу, под навигационной панелью).

Если настройки на странице Просмотр неверны, эта страница отображается опять, после нажатия кнопки "Пуск" и информация о введенных неверных данных будет выведена в разделе уведомлений.

9.1.3. Редактировать / Создать новый

Агенты, которые имеют права на запись, могут редактировать настройки существующего отчета, запустив пользовательский веб-интерфейс модуля статистики. В ка-

честве альтернативы они могут создать новый отчет. Соответствующие страницы могут быть запущены следующим образом:

1. Редактировать: Используя кнопку "Редактировать" в шаблоне просмотра статистики.
2. Новый: Используя ссылку "Новый" в меню Статистика навигационной панели, или кнопку "Добавить" на странице Просмотр.

Статистика редактируется с помощью мастера в четыре этапа:

1. Общие технические условия.
2. Определение элементов для X-axis.
3. Спецификация значений ряда.
4. Выбор ограничений для отчета.

Шаги с 2-4 необходимы только для создания отчетов с динамической статистикой. Для статической статистики требуется только общая информация (пункт 1).

Информация о том, как обрабатывать страницу приводится на каждом из этих экранов, ниже панели "Действия" панели Подсказок.

Если введены неправильные данные, предыдущий пользовательский интерфейс отображается заново, на этот раз уже с информацией о неправильных входных данных. Эта информация отображается в OTRS в разделе уведомлений. Следующий пользовательский веб-интерфейс для ввода отображается только в том случае, если все поля формы заполнены правильно.

1. *Общие спецификации.* Это первая страница мастера Редактирования (см. Рисунок ниже).

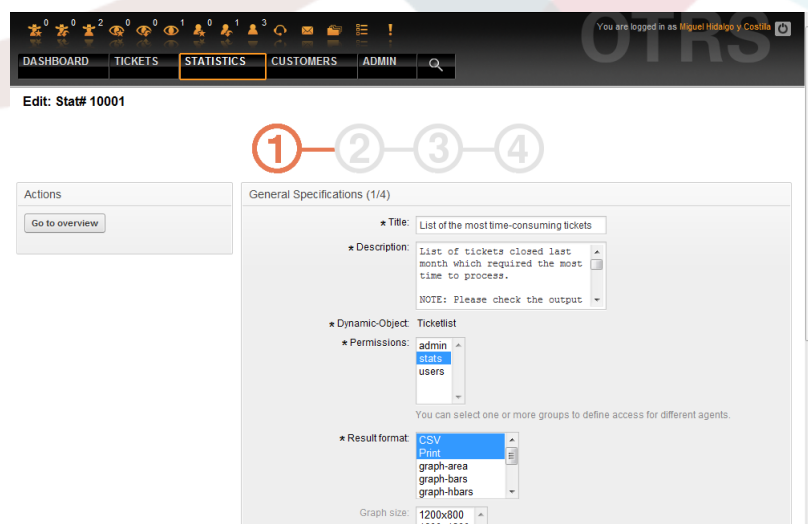


Рисунок: Редактирование общих спецификаций отчета.

На странице приведенной на Рисунке доступно большое количество общих спецификаций и настроек, которые можно редактировать:

- *Заголовок.* Должны в сжатой форме отражать назначение (цель).
- *Описание.* Более подробная информация об отчете, типах конфигурационных параметров и т.д.

- *Динамический объект.* Если установленная система OTRS предоставляет различные динамические объекты, один из них может быть выбран. Объекты отвечают требованиям отдельных модулей.
- *Static file.* Обычно эта возможность не отображается до тех пор, пока отображаются статические файлы, назначенные отчету. Если "Статический файл" отображается, позже, очень важно поставить галочку в поле и выбрать режим генерации (динамический с динамический объектом или статический с файлом). Если выбран статический файл, пользовательский веб-интерфейс 2-4 не отображается как статический файл содержащий все необходимые настройки.
- *Настройки прав доступа.* Содействие ограничению групп (и, следовательно, агентов), которые позже могут просматривать и генерировать предварительные отчеты. Таким образом, различные отчеты могут быть выделена для различных отделов и рабочих групп, которые в них нуждаются. Можно выделить одну статистику для различных групп.

Пример 1: Выбрана "stats" группа. Отчет доступен для просмотра для всех пользователей имеющих по крайней мере права на группу "stats". Этот доступ доступен по умолчанию.

Пример 2: Была выбрана группа под названием "продажи". Все пользователи, имеющие права ro (read only) к этой группе могут видеть статистику в режиме просмотра и генерировать ее. Однако отчет не будет доступен для просмотра другим пользователям.

- *Формат.* Формат отображения статистики. В зависимости от конфигурации, могут быть выбраны следующие один или несколько форматов:
 - CSV.
 - Печать.
 - graph-lines.
 - graph-bars.
 - graph-hbars.
 - graph-points.
 - graph-lines-points.
 - graph-area.
 - graph-pie.
- *Размер графика.* Выберите размер диаграммы в пикселях. Этот выбор необходим только в том случае, если графический формат вывода был выбран в "Формат". Все графические размеры, которые можно использовать определяются в OTRS в SysConfig. При настройке отчета, агент может предварительно выбрать все соответствующие форматы.
- *Количество строк.* Указывает когда отчет заменяется колонкой, чьи ячейки содержат сумму соответствующих строк.
- *Сумма столбцов.* Указывает когда отчет заменяется строкой, чьи ячейки содержат сумму соответствующих строк.

- **Кэш.** Указывает на то, что созданный отчет должен кэшироваться в файловой системе. Это позволяет существенно экономить вычислительные мощности и время, но должно использоваться только в том случае если содержание отчета больше не будет изменяться.

Кэширование автоматически прекращается, если отчет не содержит временных обозначений, или когда временные обозначения указывают на будущее.

В случае редактирования закэшированного отчета удаляются все закэшированные данные.

- **Действительный.** Устанавливается в значение "недействительный" если ранее сконфигурированный отчет по какой то причине не должен выполняться. Кнопка "Статистика" также не отображается в правом нижнем углу блока. Отчет больше не генерируется.

2. **Определение элементов для X-axis.** Это конфигурация для элементов, которые используются для описания X-axis или, если используются таблицы, имя столбца применяется для X-axis (см. Рисунок ниже).

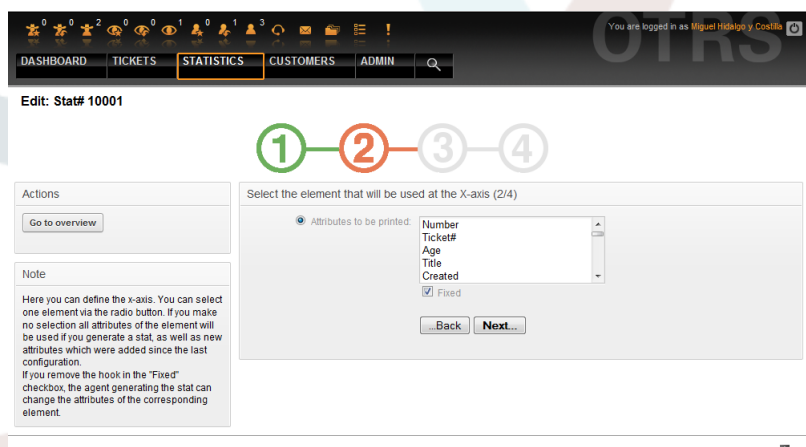


Рисунок: Определение элементов для X-axis.

Во-первых, элемент выбирается с помощью опционального поля. Потом нужно выбрать два или более атрибута элемента. Если выбранных атрибутов нету, используются все атрибуты, в том числе и те, которые были добавлены после конфигурации отчета.

Если отключена настройка "Фиксированный", агент, генерирующий отчет, может изменить атрибуты соответствующего элемента в меню "Вид" пользовательского веб-интерфейса.

Элементы времени отличаются от периодов времени. Тип и число элементов зависит от использующихся динамических объектов и сильно зависит от этого.

Если все входные данные введены правильно, нажатие кнопки "Следующий шаг" открывает форму "Серия значений". Кроме этого, есть возможность заново вернуться к редактированию предыдущих разделов.

3. **Спецификация серии значений.**

На третьем шаге настройки отчета нужно определить серию значений (см. Рисунок ниже). Позже они будут использованы для формирования графиков или различных серий в табличном виде.

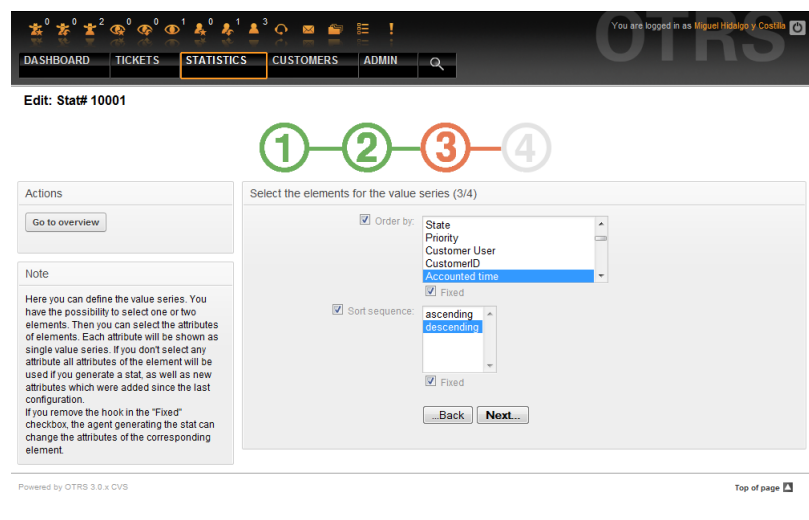


Рисунок: Определение значений серии.

Если выбран определенный элемент, каждый выбранный атрибут будет соответствовать значению серии (см. ниже Пример 19-1).

Пример 4.19. Определение значения серии - один из элементов

Элемент Очередь:

- Серия значений 1 = Raw
- Значение серии 2 = Junk
-

Если для формирования значения ряда, выбрано два элемента, каждый выбранный атрибут первого элемента комбинируется с атрибутом второго элемента (см. ниже Рисунок 19-2).

Пример 4.20. Определение значения серии - один из элементов

Элемент 1 очередь, Элемент 2 статус:

- Значение создания 1 = Raw - открыть
- Серия значений 2 = Raw - успешно закрыта
- Серия значений 3 = Junk - открыто
- Серия значений 4 = Junk - успешно закрыто

Выбор из трех или более элементов не допускается.

Кроме этого эти условия распространяются и на выбор атрибутов и флажок "Основные" как для "X-axis":

- Если для элемента не выбрано никаких атрибутов, то используются все атрибуты, включая и те, которые были добавлены после настройки отчета.
- Если параметр "Фиксированный" отключен, агент, который создает отчет может изменять атрибуты соответствующих элементов.

4. *Установка ограничений отчета.* Это четвертый и последний этап конфигурации (см. Рисунок ниже). Ограничения служат для ограничения результатов выбранных критериев. В ряде случаев не должно быть никаких ограничений.

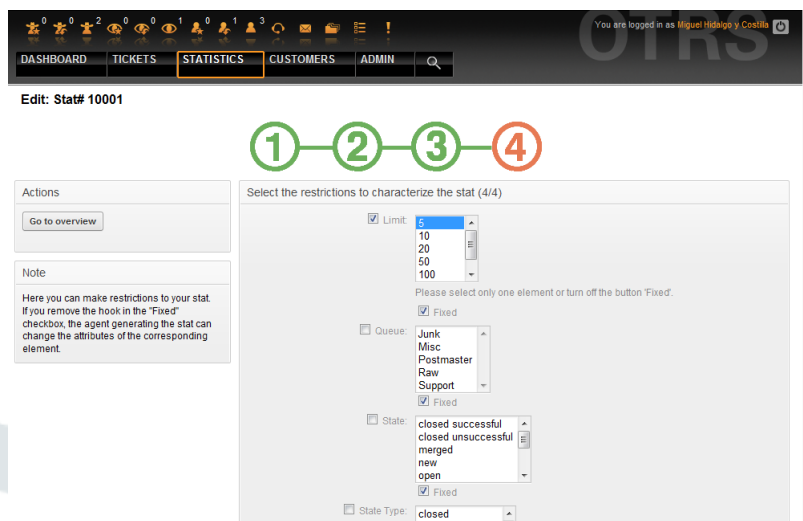


Рисунок: Определение ограничений.

После установки всех ограничений, конфигурация отчета завершается нажатием кнопки "Готово".

9.1.4. Импорт

Для доступа к пользовательскому интерфейсу Импорта (см. Рисунок ниже) перейдите по ссылке "Статистика", а затем выберите "Импорт". Альтернативный способ - на странице "Обзор" нажать кнопку Импорт, это приведет к тому же результату, но для этого действия требуются права для чтения и записи "RW".

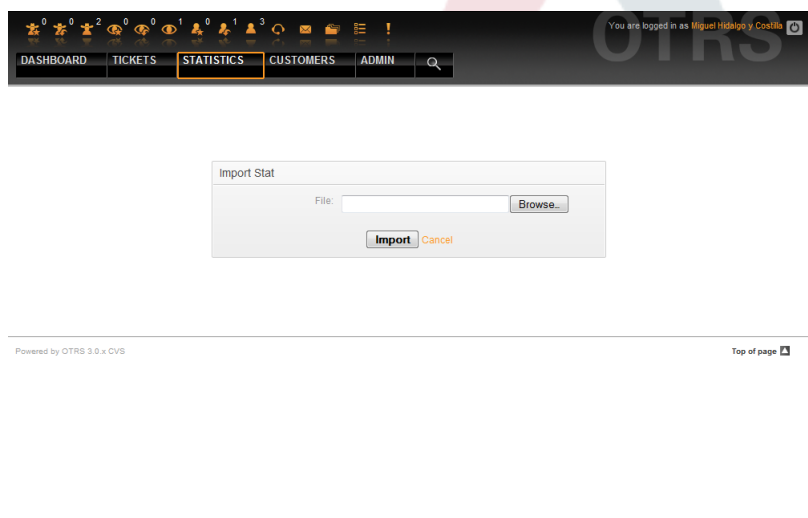


Рисунок: Пользовательский веб-интерфейс для Импорта.

Очень удобная функция, значительно облегчает импорт статистики, особенно в сочетании с возможностями модуля экспорта. Статистику можно создать и протестировать на тестовых системах, а затем импортировать в реальную систему.

Импорт осуществляется путем простой загрузки файла. После этого, для импорта статистики, автоматически открывается пользовательский интерфейс "Вид".

9.2. Администрирование модуля статистики OTRS администратором.

Этот раздел предоставляет информацию о задачах и функциях администратора OTRS, который работает с модулем статистики.

9.2.1. Настройки полномочий, Групп и Ролей

Нет новых очередей и/или групп, созданных при установке модуля статистики.

В конфигурации по умолчанию модуль регистрации дает всем агентам группы "статистика" права доступа к модулю статистики.

Доступ в соответствии с разрешенными настройками:

- *rw*. Разрешает настройку статистики и отчетов.
- *ro*. Разрешает генерацию предварительно сконфигурированной статистики и отчетов.

OTRS-администратор самостоятельно определяет когда агенты с правами для генерации предварительно настроенных отчетов выделяются в *ro*-права для группы "статистика", или если их соответствующие группы добавляются в модуль регистрации в SysConfig.

9.2.2. SysConfig

Группы SysConfig Framework:Core::Stats, Framework:Core::Stats::Graph and Framework:Frontend::Agent::Stats содержат все параметры конфигурации базового набора из модуля статистики. Больше того, конфигурационный параметр `$Self->{'Frontend::Module'}->{'AgentStats'}` управляет организацией и регистрацией модулей и иконками в модуле статистики.

9.3. Администрирование модуля статистики системным администратором

Как правило, системного администратора не требуется для эксплуатации, настройки и обслуживания модуля статистики. Однако, немного справочной информации дано для системного администратора в этой точке.

Примечание

Пути к файлам см. в подкаталогах домашнего каталога OTRS (в большинстве случаев `/opt/otrs`).

9.3.1. Таблица базы данных

Включены все конфигурационные параметры отчета и управляются в XML и, следовательно, хранятся в таблице базы данных "xml_storage". Другие модули, контент которых хранится в xml-формате также используют эту таблицу.

9.3.2. Список файлов

Следующие файлы необходимы для корректной работы модуля статистики:

- Kernel/System/Stats.pm
- Kernel/Modules/AgentStats.pm
- Kernel/System/CSV.pm
- Kernel/Output/HTML/Standard/AgentStatsOverview.dtl
- Kernel/Output/HTML/Standard/AgentStatsDelete.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditSpecification.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditRestrictions.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditXaxis.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditValueSeries.dtl
- Kernel/Output/HTML/Standard/AgentStatsImport.dtl
- Kernel/Output/HTML/Standard/AgentStatsPrint.dtl
- Kernel/Output/HTML/Standard/AgentStatsView.dtl
- Kernel/System/Stats/Dynamic/Ticket.pm
- bin/otrs.GenerateStats.pl

9.3.3. Кэширование

Результаты статистики должны быть сохранены в кэше или не могут быть установлены в конфигурации. Кэшированные результаты отчетов хранятся в файлах в каталоге `var/tmp` установленной системы OTRS (в большинстве случаев `/opt/otrs/var/tmp`).

Кэшированная статистика может быть распознана с помощью префикса "Stats".

В случае потери данных системе не будет нанесено серьезного ущерба. При вызове модуля статистики в следующий раз, в случае если он не обнаружит файл, отчет сгенерируется заново. Конечно, это может занять немного больше времени.

9.3.4. `otrs.GenerateStats.pl`

Этот файл хранится в директории `bin`. Он позволяет генерировать статистику прямо из командной строки.

В качестве примера смотрите вызов Сценария из командной строки, который приведен ниже.

```
bin> perl otrs.GenerateStats.pl -n 10004 -o /output/dir
```

Сценарий: Создание отчета из командной строки.

Отчет из конфигурации статистики "Stat# 10004" генерируется и хранится в csv-формате в директории `/output/dir`.

Сгенерированный отчет также может быть отправлен по электронной почте. Более подробную информацию можно получить выполнив команду из Сценария приведенного ниже.

```
bin> perl otrs.GenerateStats.pl --help
```

Сценарий: Получение информации о файле `otrs.GenerateStats.pl`.

9.3.5. Автоматическое создание статистики с помощью планировщика заданий - Cronjob

Очевидным является тот факт, что в действительности, нету необходимости генерировать отчеты вручную, из командной строки, так как модуль статистики имеет удобный графический веб-интерфейс пользователя. Однако, создание отчетов вручную имеет смысл в сочетании с Cronjob.

Представьте себе следующую ситуацию: Каждый первый день месяца, руководители отделов хотят получать отчет за последний месяц. Объединив cronjob с командой запуска сценария в командной строке, статистика может отправляться им по электронной почте автоматически.

9.3.6. Статические отчеты

Модуль статистики облегчает генерацию статической статистики. Для каждой статической статистики создается отдельный файл, в котором точно определено его содержание.

Таким образом может быть сгенерирована очень сложная статистика. Главным недостатком является то, что этот способ есть не очень гибким.

Файлы хранятся в директории `Kernel/System/Stats/Static/`.

9.3.7. Использование устаревших данных статической статистики

Прежние версии OTRS 1.3 и 2.0 уже позволяли генерировать отчеты. Различные отчеты для OTRS версии 1.3 и 2.0 которые были специально разработаны для удовлетворения потребностей клиентов могут быть использованы в более поздних версиях OTRS.

Файлы должны быть просто перемещены из `Kernel/System/Stats/` в `Kernel/System/Stats/Static/`. Кроме этого имя пакета соответствующего сценария должен быть заменен на `:::Static`.

В следующем примере показано как изменяется первый путь.

```
package Kernel::System::Stats::AccountedTime;
```

```
package Kernel::System::Stats::Static::AccountedTime;
```

9.3.8. Статистика по умолчанию

"Не всегда нужно изобретать велосипед..."

Модуль статистики предоставляет различные предустановленные отчеты. Предустановленные отчеты, которые могут заинтересовать всех OTRS пользователей можно в будущем добавить в модуль статистики. Предустановленные отчеты хранятся в модуле статистики в xml-формате в директории `scripts/test/sample/`.

10. Динамические Поля

10.1. Введение

Динамические поля - это специальный вид полей в OTRS, создаваемых для расширения перечня информации, хранимой в заявке или сообщении/заметке. Эти поля не являются фиксированными в системе и они появляются только на определенных экранах, они могут быть обязательными или нет, и их представление на экране зависит от типа поля, заданного при их создании. Например, поля могут содержать текст, даты, набор из нескольких выбранных значений и т.д.

Динамические поля являются развитием свободных полей TicketFreeText, TicketFreeKey, TicketFreeTime, ArticleFreeText и ArticleFreeKey, ранее использовавшихся в OTRS 3.0 и предыдущих версиях. Ограничением этих "Свободных полей" состояло в том, что можно было задать до 16 (текстовых или выбираемых из списка) полей и 6 полей, содержащих значение времени для заявки и 3 (текстовых или выбираемых из списка) полей для каждого сообщения/заметки, не более.

Теперь, с появлением динамических полей, эти ограничения сняты, вы можете создавать любое количество динамических полей для заявок и сообщений. И, кроме того, интерфейс для управления ими, готов к расширению дальнейшего использования таких полей не только для заявок и сообщений, но и других объектов системы, например, для управления изменениями и пр.

Этот новый интерфейс описания и управления динамическими полями, использует модульную структуру, в которой каждый тип динамических полей может рассматриваться как дополнительный модуль для интерфейса. Это означает, что все множество динамических полей может быть просто расширено для публичных модулей OTRS, OTRS Feature Add-ons, пользовательских разработок и пр.

В этом релизе доступны следующие типы динамических полей:

- Text (одна строка текста)
- Textarea (несколько строк текста)
- Checkbox
- Выпадающий список (единственный выбор, несколько значений)
- Multiselect (множественный выбор, несколько значений)
- Date
- Date / Time

10.2. Настройка Системы

По умолчанию, первоначально установленная система OTRS не содержит никаких динамических полей. Если вы планируете их использование для заявок и сообщений/заметок, вам необходимо их создать. После установки некоторых дополнительных пакетов, например, ITSM, в системе появятся стандартные динамические поля, необходимые для их работы.

Настройка динамических полей разделена на две части. Для создания новых или управления ранее созданными полями перейдите по ссылке "Динамические поля" в Панели Администратора. Для управления показом, обязательностью показа или скрытием полей на конкретных экранах используйте настройки Конфигурации системы/SysConfig для этих экранов.

10.2.1. Добавление Динамического Поля

Откройте Панель Администратора, нажмите на ссылку "Динамические поля" в разделе "Настройки заявок", расположенном в нижней половине экрана, по центру. Список динамических полей выглядит следующим образом:

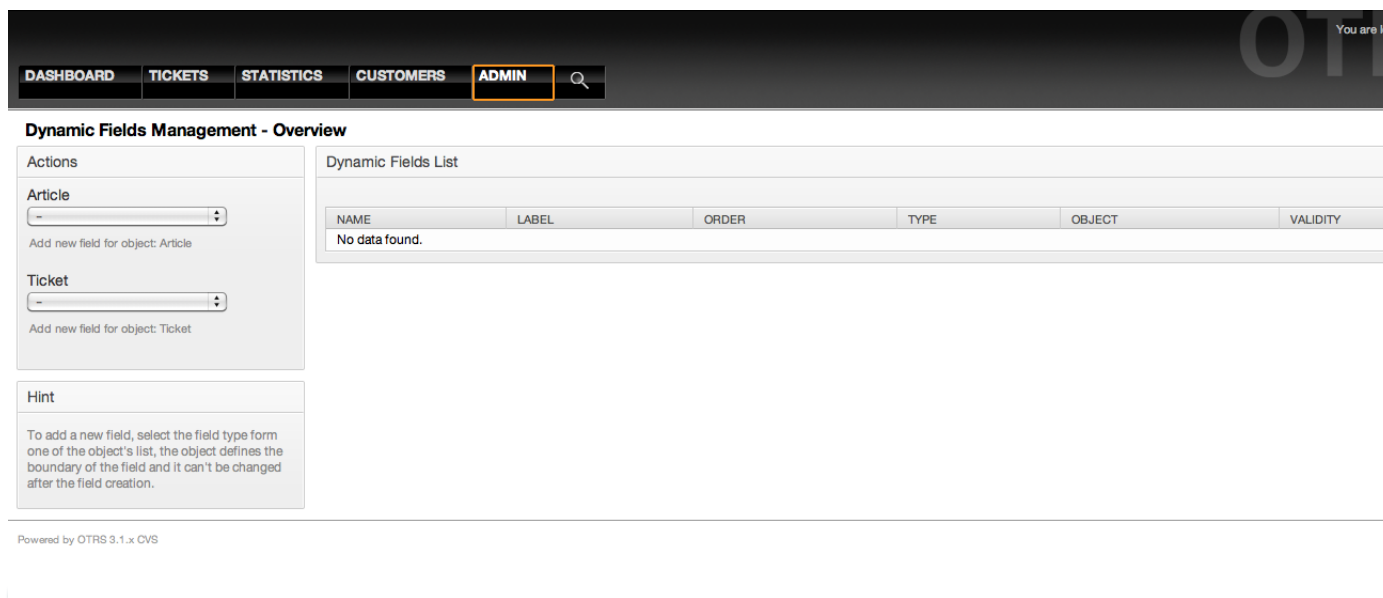


Рисунок: Страница просмотра Динамических полей, пустая

Обратите внимание, что вид экрана изменится при добавлении динамических полей. Он также может уже иметь несколько полей, если вы провели обновление со старых версий OTRS, в которых динамические поля еще не использовались. Они теперь заменили понятие Свободные/Дополнительные поля/FreeText fields.

Раздел "Действия" в левой части экрана позволяет выбрать два варианта объекта для поля: Article/Заметка и Ticket/Заявка, каждый из которых имеет свой собственный набор значений типов полей в выпадающих списках.

Примечание

Установка некоторых дополнительных пакетов OTRS может привести к добавлению типов объектов в разделе "Действия".

Общая процедура для создания динамического поля:

- Кликните на объект выбранного вами динамического поля в выпадающем списке на боковой панели действий.
- Выберите в списке и нажмите на тип динамического поля, которое вы хотите добавить.
- Настройка
- Сохранить.

Диалог настройки динамических полей разделен на две части, верхняя часть является общей для всех полей, а нижняя часть может различаться в зависимости от типа динамического поля.

Общие настройки динамического поля:

- **Название:** Обязательное поле, уникальное, только буквы и числа в качестве допустимых символов.

Это внутреннее имя поля, используемое, например, для показа или скрытия поля на отдельных экранах. Любое изменение имени поля (не рекомендуется) потребует ручного изменения в параметрах "SysConfig", которые ссылаются на это поле.

- **Метка:** Обязательное.

Это имя, под которым поле будет показано на тех экранах, на которых оно активно. Оно может быть переведено на другие языки.

Примечание

Перевод метки должен быть добавлен в файл перевода вручную.

- **Порядок Полей:** Обязательно.

Определяет относительный порядок, в котором поля будут отображаться на экране, по умолчанию каждое новое поле занимает последнюю позицию, изменение этого параметра будет влиять на порядок отображения других динамических полей.

- **Срок действия:** Обязательный.

Не действительное динамическое поле не будет отображаться на любом экране, независимо от того какое значение установлено в конфигурации.

- **Тип поля:** Обязательное, Только для чтения.

Показывает текущий выбранный тип поля.

- **Тип Объекта:** Обязательный, Только для чтения.

Отображает область действия для поля.

Примечание

Для иллюстрации каждого конкретного параметра настройки в нашем примере будет добавлено несколько полей. Ссылки на эти поля будут упоминаться в последующих разделах.

В следующих примерах все динамические поля будут созданы для объекта Заявка, если необходимо создать динамическое поле для объекта Статья, просто выберите поле из выпадающего списка.

Таблица 4.6. В систему будут добавлены следующие поля:

Название	Метка	Type
Поле1	Мое Поле 1	Text
Поле2	Мое Поле 2	Textarea/Длинный текст
Поле3	Мое Поле 3	Checkbox
Поле4	Мое Поле 4	Dropdown

Название	Метка	Тип
Поле5	Мое Поле 5	Multiselect (множественный выбор)
Поле6	Мое Поле 5	Date
Поле7	Мое Поле 6	Date / Time

10.2.2. Настройка Динамического Поля типа Text

Динамические текстовые поля используются для хранения простой строки.

Настройки динамического поля типа Text:

- Значение по умолчанию: Необязательное.

Это значение, которое будет отображаться по умолчанию на экранах редактирования (например, New Phone Ticket/Новая телефонная заявка или Ticket Compose/Создать ответ).

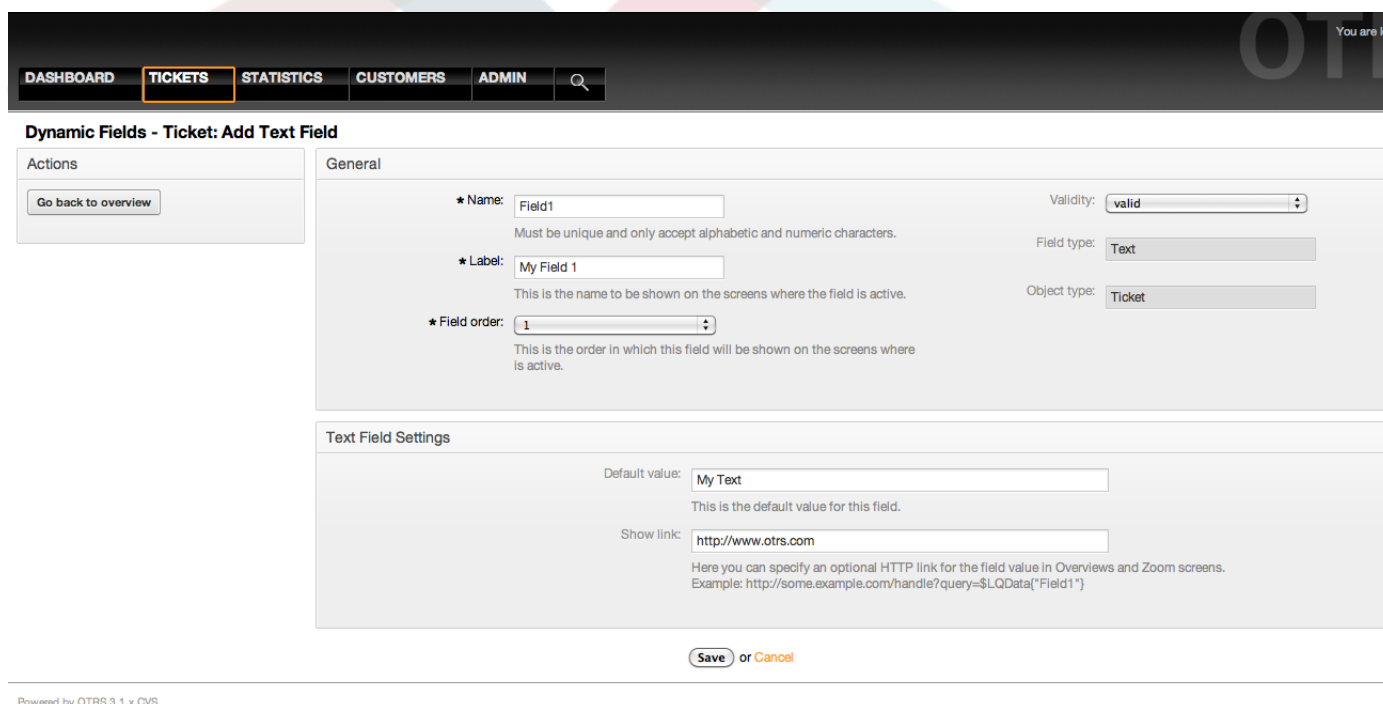
- Показать ссылку: Необязательное.

Если установлено, то значение поля будет преобразовано в кликабельную ссылку, которая будет отображаться на экране.

Например, если "Показать ссылку" установлено в "http://www.otrs.com", нажатие на заполненное значение приведет к открытию веб-страницы OTRS.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.



The screenshot shows the 'Dynamic Fields - Ticket: Add Text Field' configuration window. It is divided into several sections:

- Actions:** Contains a 'Go back to overview' button.
- General:**
 - Name:** Field1 (with a note: 'Must be unique and only accept alphabetic and numeric characters.')
 - Label:** My Field 1 (with a note: 'This is the name to be shown on the screens where the field is active.')
 - Field order:** 1 (with a note: 'This is the order in which this field will be shown on the screens where is active.')
 - Validity:** valid
 - Field type:** Text
 - Object type:** Ticket
- Text Field Settings:**
 - Default value:** My Text (with a note: 'This is the default value for this field.')
 - Show link:** http://www.otrs.com (with a note: 'Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens. Example: http://some.example.com/handle?query=\$LQData{"Field1"}')

At the bottom, there are 'Save' and 'Cancel' buttons.

Рисунок: Диалоговое окно настройки Динамического поля Text.

10.2.3. Настройка Динамического Поля Textarea

Динамическое поле Textarea используется для хранения строк, состоящих из нескольких линий.

Настройки динамического поля Textarea:

- Количество строк: Необязательное, Целое.

Используется для определения высоты поля на странице редактирования.

- Количество столбцов: Необязательное, Целое.

Это значение используется для задания ширины поля при редактировании.

- Значение по умолчанию: Необязательное.

Это значение будет отображаться по умолчанию (может быть текстом, состоящим из нескольких строк).

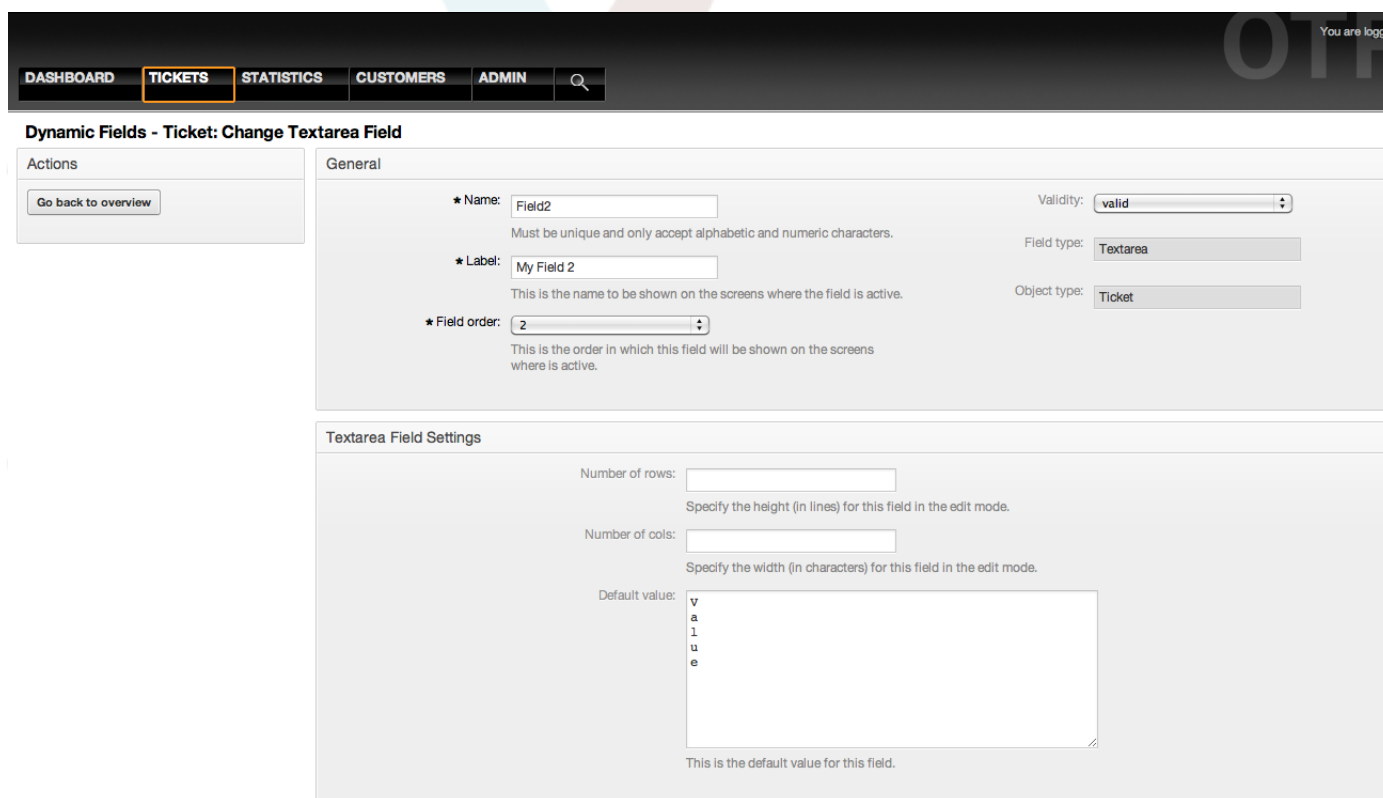


Рисунок: Окно настройки Динамического Поля Textarea.

10.2.4. Конфигурация Динамического Поля Checkbox

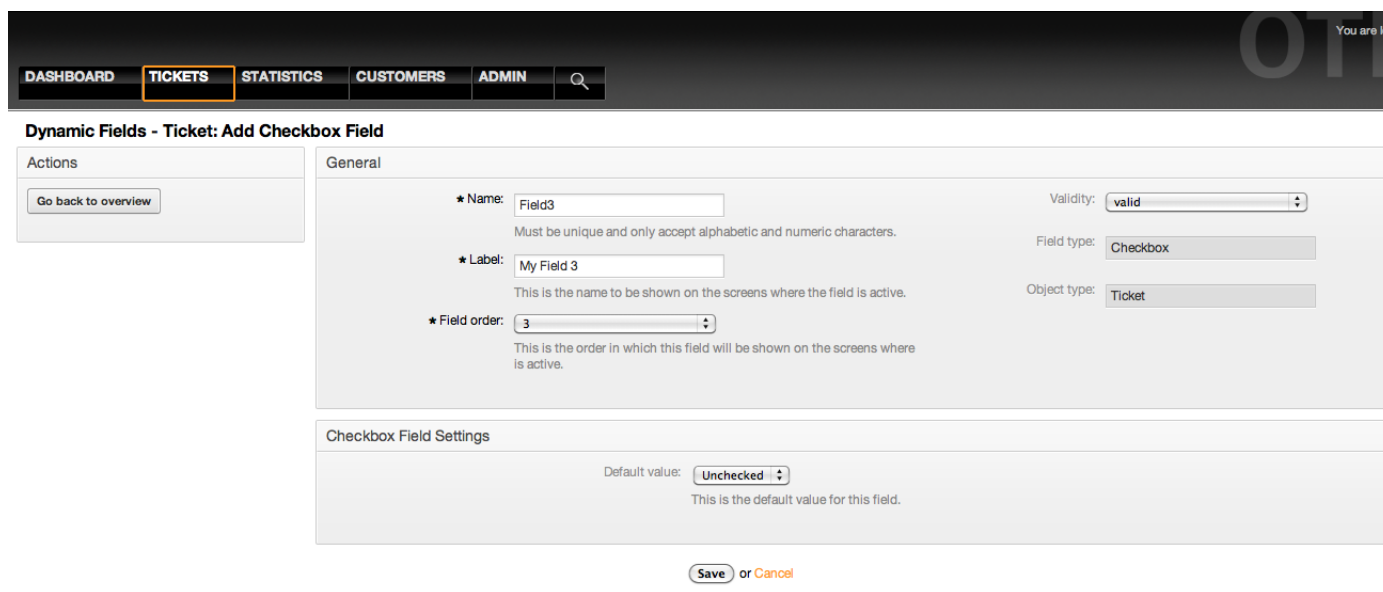
Динамическое поле Checkbox используется для хранения значений true или false, изображено как выбранный или не выбранный флажок.

Настройки динамического поля Checkbox:

- Значение по умолчанию: Обязательное.

Это значение, которое будет показано по умолчанию на страницах редактирования (например заявка, созданная на основе телефонного звонка), значение по

умолчанию для этого поля недоступно для выбора и может принимать значения Выбрано/Checked или Unchecked/Не выбрано.



Powered by OTRS 3.1.x CVS

Рисунок: Окно настройки Динамического Поля Checkbox.

10.2.5. Настройка Динамического Поля Dropdown (Выпадающий Список)

Динамическое поле Выпадающий Список (Dropdown) используется для хранения одного значения, которое выбирается из закрытого списка.

Настройки динамического поля Dropdown:

- Возможные значения: Обязательное.

Список значений для выбора. При добавлении нового значения необходимо указать Ключ (внутреннее значение) и Значение (значение, которое отображается).

- Значение по умолчанию: Необязательное.

Это значение будет отображено по умолчанию на страницах редактирования (например, на странице Новая телефонная заявка, или странице создания ответа), значение по умолчанию для этого поля - определяется параметром "Значение по умолчанию" с возможностью выбора всех возможных значений в списке "Значения".

- Добавить пустое значение: Обязательное, (Да / Нет).

Если эта опция включена, дополнительная стоимость определяется для отображения "-" в списке возможных значений, это специально значение пустоты внутри.

- Переводимые значения: Обязательное, (Да / Нет).

Этот параметр задает возможность перевода на язык системы значений этого поля. Переводятся только отображаемые значения, внутренние значения не подлежат переводу. Перевод полей должен быть занесен вручную в соответствующий пользовательский файл локализации. Для русского языка - ru_custom.pm.

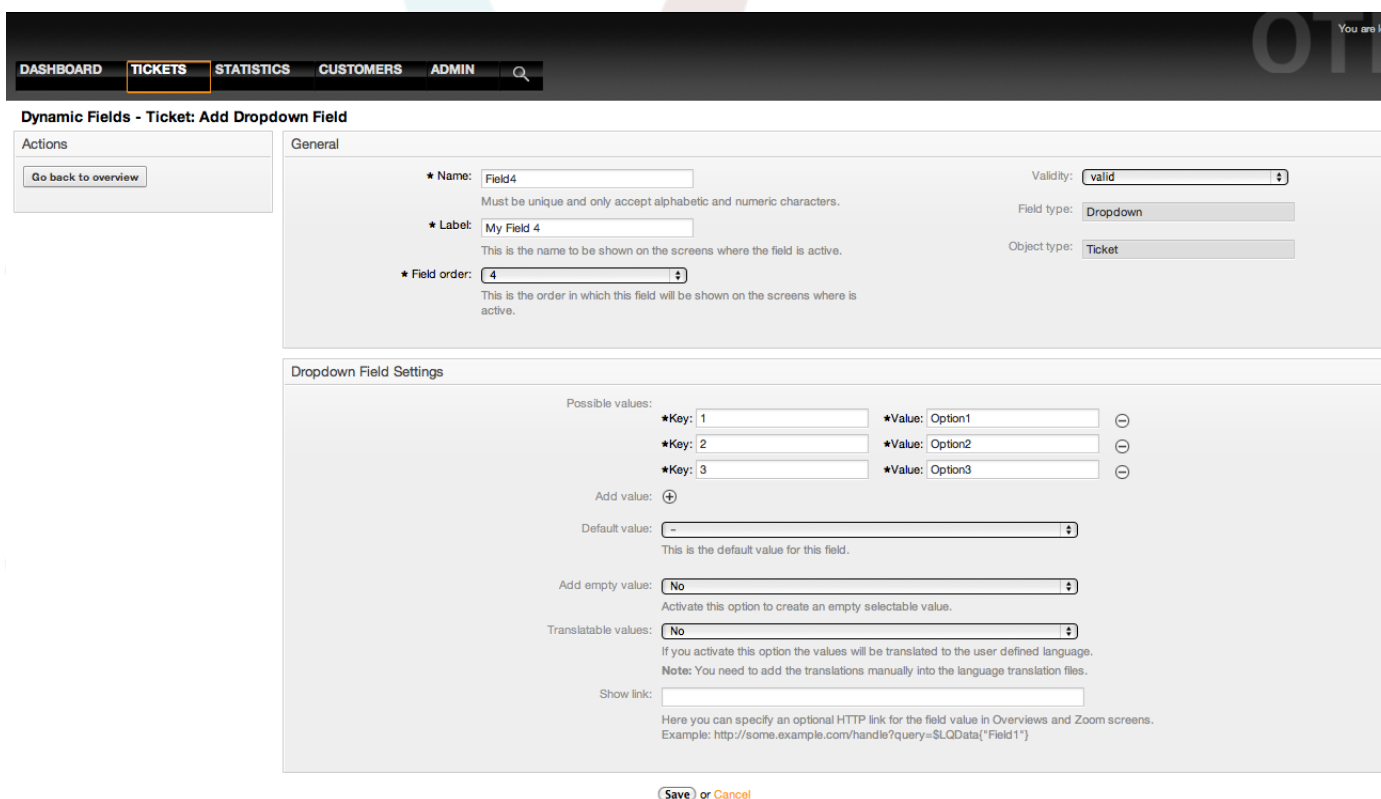
- Показать ссылку: Необязательное.

Если установлено, то значение поля будет установлено в кликабельные ссылки.

Например, если ссылка Показать установлена в значение "http://www.otrs.com", нажатие на значение поля приведет к открытию веб-страницы OTRS.

Примечание

Использование конструкции `$LQData{"NameX"}` в поле Set link/Показывать ссылку, где NameX - имя поля, добавляет значение поля как часть ссылки.



The screenshot shows the 'Dynamic Fields - Ticket: Add Dropdown Field' configuration window. It is divided into several sections:

- Actions:** A button labeled 'Go back to overview'.
- General:**
 - * Name: Field4 (with a note: 'Must be unique and only accept alphabetic and numeric characters.')
 - * Label: My Field 4 (with a note: 'This is the name to be shown on the screens where the field is active.')
 - * Field order: 4 (with a note: 'This is the order in which this field will be shown on the screens where is active.')
 - Validity: valid
 - Field type: Dropdown
 - Object type: Ticket
- Dropdown Field Settings:**
 - Possible values:** A list of three entries:
 - *Key: 1, *Value: Option1
 - *Key: 2, *Value: Option2
 - *Key: 3, *Value: Option3
 - Add value:** A plus sign button.
 - Default value:** A dropdown menu with a minus sign.
 - Add empty value:** A dropdown menu set to 'No'.
 - Translatable values:** A dropdown menu set to 'No'.
 - Show link:** A text input field.

At the bottom right, there are 'Save' and 'Cancel' buttons.

Рисунок: Окно настройки динамического поля Dropdown.

10.2.6. Конфигурация Динамического Поля Multiselect (множественный выбор)

Динамическое поле Multiselect используется для хранения нескольких значений из заданного списка.

Настройка Динамического Поля Multiselect (множественный выбор):

- Возможные значения: Обязательное.

Список значений для выбора. При добавлении нового значения в список, необходимо указать Ключ (внутреннее значение) и Значение (значение, которое отображается).

- Значение по умолчанию: Необязательное.

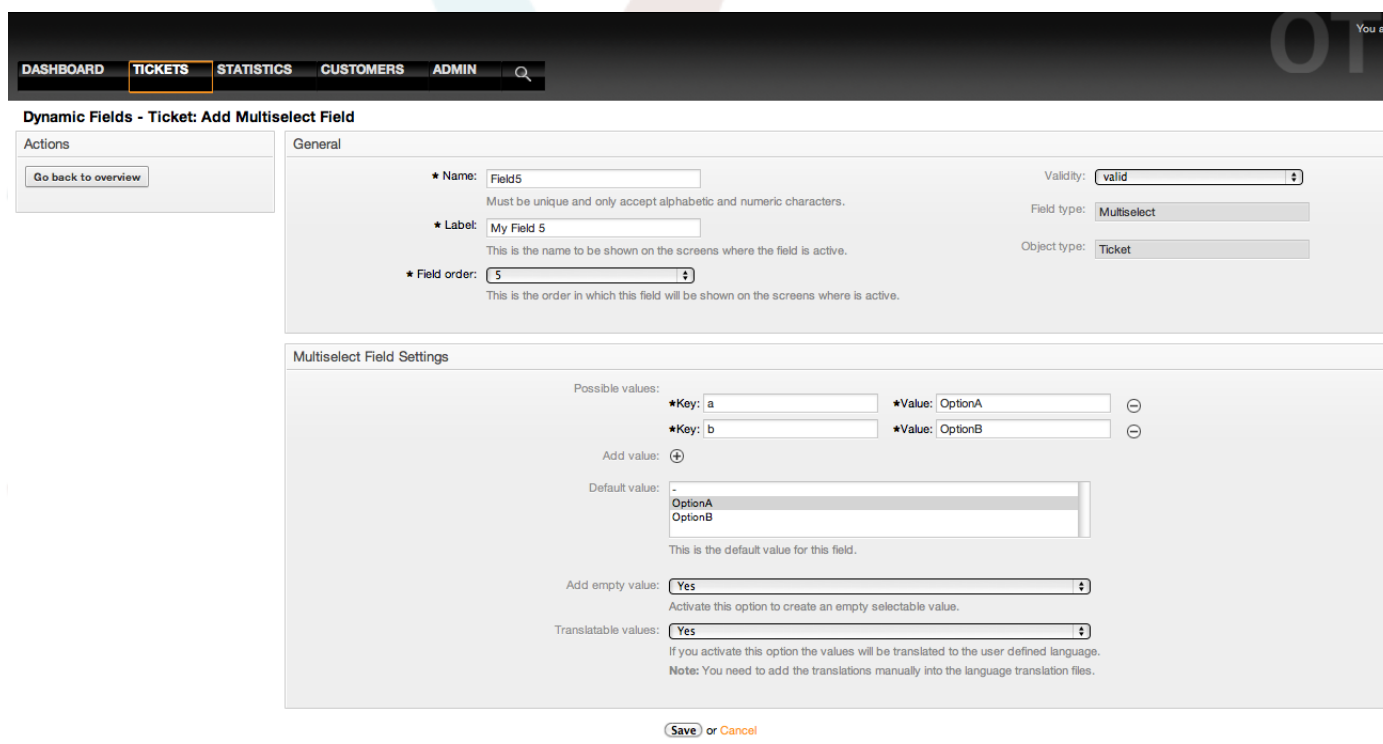
Это значение будет отображено по умолчанию на страницах редактирования (например, на странице Заявки, созданной на основе телефонного звонка или странице создания Заявки), значение по умолчанию для этого поля - задается параметром "Значение по умолчанию" из заданного списка значений.

- Добавить пустое значени: Обязательное, (Да / Нет).

Если эта опция включена, задается возможность для отображения "-" в списке возможных значений, это специальное значение для указания на пустое значение.

- Переводимые значения: Обязательное, (Да / Нет).

Этот параметр задает возможность перевода на язык системы значений этого поля. Переводятся только отображаемые значения, внутренние значения не подлежат переводу. Перевод полей должен быть занесен вручную в соответствующий пользовательский файл локализации. Для русского языка - ru_custom.pm.



The screenshot shows the 'Dynamic Fields - Ticket: Add Multiselect Field' configuration page. It is divided into several sections:

- General:**
 - * Name: Field5 (with a note: 'Must be unique and only accept alphabetic and numeric characters.')
 - * Label: My Field 5 (with a note: 'This is the name to be shown on the screens where the field is active.')
 - * Field order: 5 (with a note: 'This is the order in which this field will be shown on the screens where is active.')
 - Validity: valid
 - Field type: Multiselect
 - Object type: Ticket
- Multiselect Field Settings:**
 - Possible values:**
 - *Key: a, *Value: OptionA
 - *Key: b, *Value: OptionB
 - Add value:** +
 - Default value:** A list box containing OptionA and OptionB, with a note: 'This is the default value for this field.'
 - Add empty value:** Yes (with a note: 'Activate this option to create an empty selectable value.')
 - Translatable values:** Yes (with a note: 'If you activate this option the values will be translated to the user defined language. Note: You need to add the translations manually into the language translation files.')

At the bottom, there are 'Save' and 'Cancel' buttons.

Powered by OTRS 3.1.x CVS

Рисунок: Окно настройки динамического поля Multiselect.

10.2.7. Настройка Динамического Поля Date

Динамическое поле Date используется для хранения значения (День, Месяц и Год).

Настройки динамического поля Date (Дата):

- Разница дат по умолчанию: Опционально, Целое.

Количество секунд (положительное или отрицательное значение) между текущей датой и выбранной датой, которое будет отображаться на страницах редактирования по умолчанию (например на странице Новая Заявка на основе телефонного звонка или Создать Заявку).

- Определить период лет: Обязательно (Да / Нет).

Используется для установки определенного количества лет в прошлом и будущем относительно текущей даты, установленной для этого поля. Если установлено в значение "Да", следующие опции будут доступны:

- Лет в прошлом: Опционально, положительное целое число.

Задаёт количество лет в прошлом относительно текущего дня для отображения при выборе из списка для значения в поле Год.

- Лет в будущем: Опционально, Положительное целое.

Задаёт количество лет в будущем, относительно текущего дня, при выборе значения из списка для отображения в поле Год.

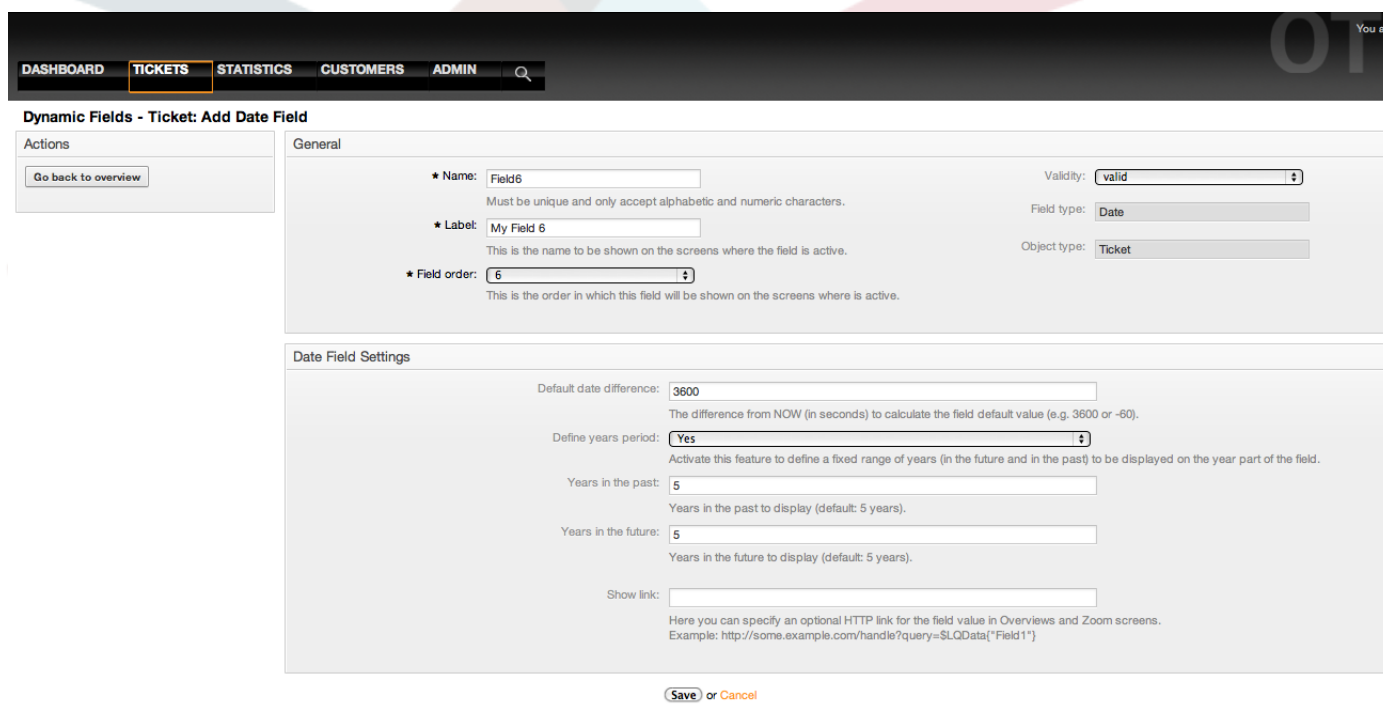
- Показать ссылку: Необязательное.

Если установлено, то значение поля будет установлено в кликабельные ссылки.

Например, если ссылка Показать установлена в значение "http://www.otrs.com", нажатие на значение поля приведет к открытию веб-страницы OTRS.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.



The screenshot shows the 'Dynamic Fields - Ticket: Add Date Field' configuration page. It is divided into several sections:

- Actions:** A button labeled 'Go back to overview'.
- General:**
 - * Name: Field6 (with a note: 'Must be unique and only accept alphabetic and numeric characters.')
 - * Label: My Field 6 (with a note: 'This is the name to be shown on the screens where the field is active.')
 - * Field order: 6 (with a note: 'This is the order in which this field will be shown on the screens where is active.')
 - Validity: valid
 - Field type: Date
 - Object type: Ticket
- Date Field Settings:**
 - Default date difference: 3600 (with a note: 'The difference from NOW (in seconds) to calculate the field default value (e.g. 3600 or -60).')
 - Define years period: Yes (with a note: 'Activate this feature to define a fixed range of years (in the future and in the past) to be displayed on the year part of the field.')
 - Years in the past: 5 (with a note: 'Years in the past to display (default: 5 years).')
 - Years in the future: 5 (with a note: 'Years in the future to display (default: 5 years).')
 - Show link: (with a note: 'Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens. Example: http://some.example.com/handle?query=\$LQData{"Field1"}')

At the bottom, there are 'Save' and 'Cancel' buttons.

Powered by OTRS 3.1.x CVS

Рисунок: Окно настройки динамического поля Date.

10.2.8. Настройка Динамического Поля Date / Time

Динамическое поле Date / Time используется для хранения значения даты и времени (Минута, Час, День, Месяц и Год).

Настройки динамического поля Date / Time:

- Разница дат по умолчанию: Опционально, Целое.

Количество секунд (положительное или отрицательное значение) между текущей датой и выбранной датой, которое будет отображаться на страницах редактирования по умолчанию (например на странице Новая Заявка на основе телефонного звонка или Создать ответ).

- Определить период лет: Обязательно (Да / Нет).

Используется для установки определенного количества лет в прошлом и будущем относительно текущей даты, выбранной для значения поля. Если установлено в значение Yes, следующие опции будут доступны:

- Лет в прошлом: Опционально, положительное целое число.

Задает количество лет в прошлом относительно текущего дня для отображения при выборе из списка для значения в поле Год.

- Лет в будущем: Опционально, Положительное целое.

Задает количество лет в будущем, относительно текущего дня, при выборе значения из списка для отображения в поле Год.

- Показать ссылку: Необязательное.

Если установлено, то значение поля будет установлено в кликабельные ссылки.

Например, если ссылка Показать установлена в значение "http://www.otrs.com", нажатие на значение поля приведет к открытию веб-страницы OTRS.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.

DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

Dynamic Fields - Ticket: Add Date / Time Field

Actions

[Go back to overview](#)

General

* Name: Validity:

Must be unique and only accept alphabetic and numeric characters.

* Label: Field type:

This is the name to be shown on the screens where the field is active.

* Field order: Object type:

This is the order in which this field will be shown on the screens where is active.

Date / Time Field Settings

Default date difference:
The difference from NOW (in seconds) to calculate the field default value (e.g. 3600 or -60).

Define years period:

Activate this feature to define a fixed range of years (in the future and in the past) to be displayed on the year part of the field.

Show link:

Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens.
Example: http://some.example.com/handle?query=\$LQData["Field1"]

[Save](#) or [Cancel](#)

Powered by OTRS 3.1.x CVS

Рисунок: Окно настройки динамического поля Date / Time.

10.2.9. Редактирование Динамического Поля

Страница просмотра заполненного динамического поля (с предыдущими примерами) должна выглядеть так:

DASHBOARD **TICKETS** STATISTICS CUSTOMERS **ADMIN**

Dynamic Fields Management - Overview

Actions

Article

Add new field for object: Article

Ticket

Add new field for object: Ticket

Hint

To add a new field, select the field type from one of the object's list, the object defines the boundary of the field and it can't be changed after the field creation.

Dynamic Fields List

NAME	LABEL	ORDER	TYPE	OBJECT	VALIDITY
Field1	My Field 1	1	Text	Ticket	valid
Field2	My Field 2	2	Textarea	Ticket	valid
Field3	My Field 3	3	Checkbox	Ticket	valid
Field4	My Field 4	4	Dropdown	Ticket	valid
Field5	My Field 5	5	Multiselect	Ticket	valid
Field6	My Field 6	6	Date	Ticket	valid
Field7	My Field 7	7	Date / Time	Ticket	valid

Powered by OTRS 3.1.x CVS

Рисунок: Окно просмотра Динамического поля заполненного тестовыми данными.

Чтобы изменить или редактировать динамическое поле, нужно иметь определенным хотя бы одно динамическое поле, которое находится в списке выбора динамических полей на странице просмотра.

Примечание

Не все настройки динамического поля могут быть изменены, тип Поля и тип Объекта не могут быть изменены.

Не рекомендуется изменять внутреннее имя поля, но внешнее название/метка, может быть изменено в любое время. Если вы все-таки изменили внутреннее имя, необходимо изменить в "SysConfig" все настройки ссылающиеся на это имя, а также в личных настройках агента (если там задано).

10.2.10. Отображение Динамического Поля на Экране

Есть два обязательных условия, которые необходимо выполнить, чтобы отобразить динамическое поле на определенных страницах:

1. Динамическое поле должно быть действительным
2. Динамическое поле должно быть установлено в 1 или 2 на странице с настройками.

Следуйте этим шагам чтобы отобразить динамическое поле на экране

- Убедитесь что значение динамического поля установлено правильно. Убедитесь в правильности значения можно посмотрев на страницу просмотра. Немедленно отредактируйте значение, чтобы сделать его правильным.
- Откройте раздел "SysConfig"/Конфигурация системы выберите подраздел "Ticket" из выпадающего списка в левой верхней части экрана - фрейм "Действия".

Примечание

Вы также можете искать параметры, использующие конкретное динамическое поле через окно поиска или, непосредственно параметр "sysconfig", если знаете его название.

- Найдите подгруппу параметров для экрана, который ищите и нажмите на его название. Например, "Frontend::Agent::Ticket::ViewPhoneNew".
- Поиск параметра, наименование которого заканчивается на "###DynamicField". Например, "Ticket::Frontend::AgentTicketPhone###DynamicField".
- Если параметр пустой или не содержит требуемого имени динамического поля, нажмите кнопку "+" и добавьте необходимую пару значений. Например, Ключ: Поле1, Содержание: 1.

Если параметр уже содержит описание нужного динамического поля, то убедитесь, что "Содержание" установлено в "1" для его отображения, или "2", чтобы сделать его обязательным для ввода.

- Сохраните изменения параметра, нажав кнопку "Обновить" в нижней части экрана и перейдите к описанию экрана, где это поле должно отображаться.

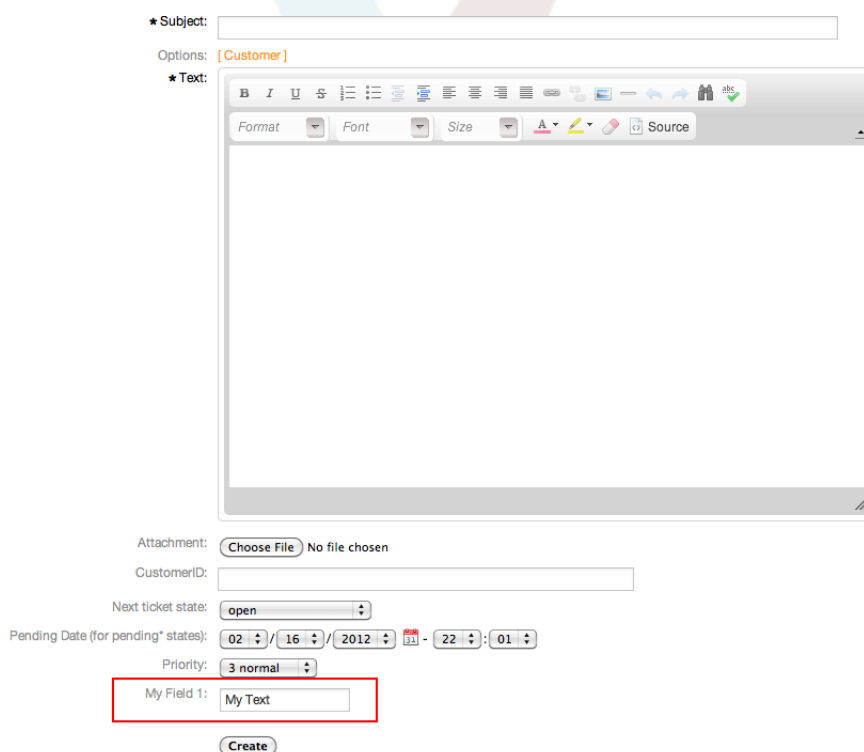
10.2.10.1. Показать Примеры

Далее показаны примеры настройки в "sysconfig" для отображения или скрытия показа динамических полей на разных экранах.

Пример 4.21. Активировать Field1 на странице Новая Заявка на основе телефонного звонка.

- *Группировать:* Заявку
- *Подгруппа:* Frontend::Agent::Ticket::ViewPhoneNew
- *Настройка:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Значение:*

Ключ	Content
Поле1	1



* Subject:
 Options: [Customer]
 * Text:

Format Font Size A Source

 Attachment: No file chosen
 CustomerID:
 Next ticket state:
 Pending Date (for pending* states): / / :
 Priority:
My Field 1:

Powered by OTRS 3.1.x CVS

Рисунок.: Экран приветствия.

Пример 4.22. Сделать Field1 на странице просмотра Заявки созданной на базе телефонного звонка New Phone Ticket Screen as mandatory.

- *Группировать:* Заявку
- *Подгруппа:* Frontend::Agent::Ticket::ViewPhoneNew
- *Настройка:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Значение:*

Attachment: No file chosen

CustomerID:

Next ticket state:

Pending Date (for pending* states): / / :

Priority:

My Field 1:

My Field 2:

My Field 3:

My Field 4:

My Field 5:

My Field 6: / / :

My Field 7: / / - :

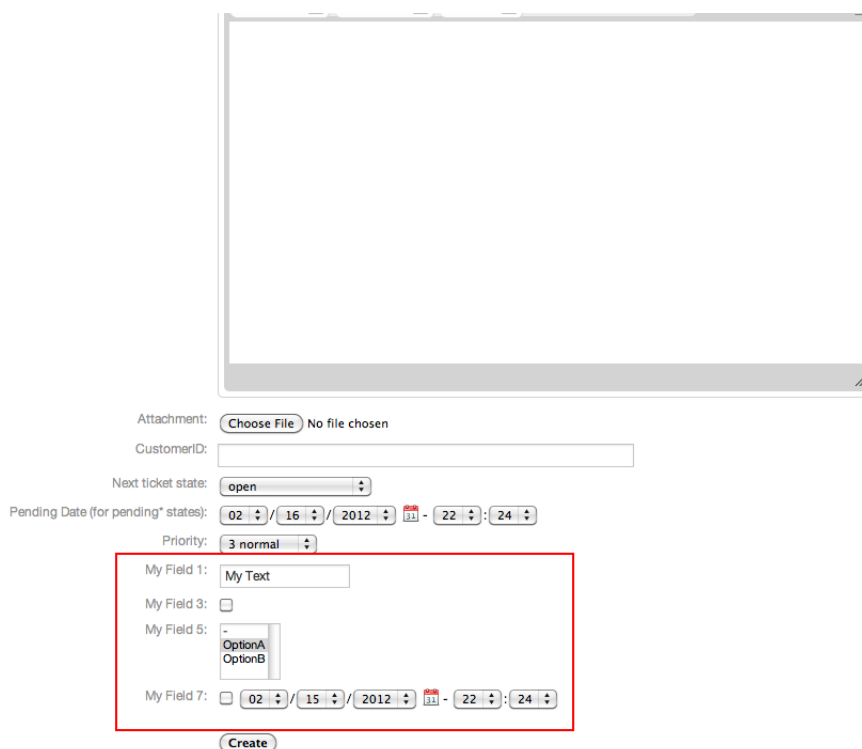
Powered by OTRS 3.1.x CVS

Рисунок: Несколько полей на странице Заявки созданной на основе телефонного звонка обязательны

Пример 4.24. Отключить некоторые поля на странице Новой Заявки, созданной на основе телефонного звонка.

- *Группировать:* Заявку
- *Подгруппа:* Frontend::Agent::Ticket::ViewPhoneNew
- *Настройка:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Значение:*

Ключ	Content
Поле1	1
Поле2	0
Поле3	1
Поле4	0
Поле5	1
Поле6	0
Поле7	1



Powered by OTRS 3.1.x CVS

Рисунок: Несколько неактивных полей на экране Создать новую Заявку на основе телефонного звонка сделать обязательными

Пример 4.25. Активировать поле Field1 на странице просмотра подробной информации о Заявке.

- Группировать: Заявку
- Подгруппа: Frontend::Agent::Ticket::ViewZoom
- Установка: Ticket::Frontend::AgentTicketZoom###DynamicField
- Значение:

Ключ	Content
Поле1	1

1

DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

Ticket#2012021510000016 – Testing Ticket

1 Article(s) Age: 4 m – Created: 02/15/2012 22:30 by My User

Back | Lock | History | Print | Priority | Free Fields | Link | Owner | Responsible | Customer | Note | Merge | Pending | Watch | Close | Change Queue

NO.	TYPE	FROM	SUBJECT	CREATED
1	customer – phone	test@test.com	Testing Ticket	02/15/2012 22:30

▼ #1 – Testing Ticket Created: 02/15/2012 22:30 by My User

Forward | Phone Call Outbound | Phone Call Inbound | Split | Print | Reply

From: test@test.com
 To: Junk
 Subject: Testing Ticket

This is a test

Powered by OTRS 3.1.x CVS

Ticket Information

State: open
 Locked: unlocked
 Priority: 3 normal
 Queue: Junk

CustomerID:
 Owner: My User
 Responsible: My User

My Field 1: My User

Customer Information

none

Linked Objects

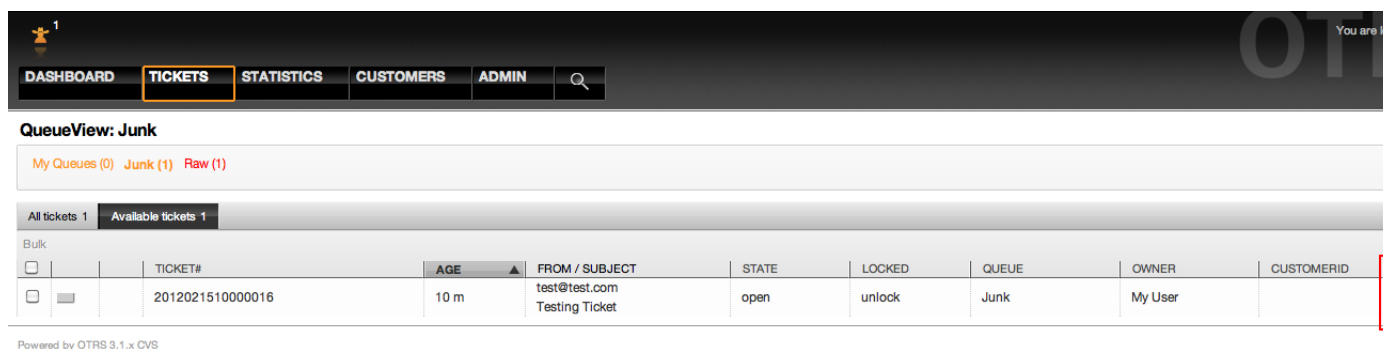
none

Рисунок: поле Field1/Поле1 на Ticket Zoom экране.

Пример 4.26. Активировать поле Field1 на странице просмотра обобщенной информации о Заявке.

- Группировать: Заявку
- Подгруппа: Frontend::Agent::TicketOverview
- Настройка: Ticket::Frontend::OverviewSmall###DynamicField
- Значение:

Ключ	Content
Поле1	1



1

DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

QueueView: Junk

My Queues (0) Junk (1) Raw (1)

All tickets 1 Available tickets 1

Bulk

	TICKET#	AGE	FROM / SUBJECT	STATE	LOCKED	QUEUE	OWNER	CUSTOMERID
<input type="checkbox"/>	2012021510000016	10 m	test@test.com Testing Ticket	open	unlock	Junk	My User	

Powered by OTRS 3.1.x CVS

Рисунок: Поле1 на экране просмотра обобщенной/small информации о Заявке.

Этот параметр влияет на шаблоны: Эскалации, Заблокированные, Очереди, Ответственные, Статус и Watch-шаблон.

10.2.11. Установка Значения по Умолчанию с помощью модуля Ticket Event

Событие Заявки (например TicketCreate) может вызывать множество значений для определенного поля, если его значение еще до сих пор не установлено.

Примечание

При использовании этого способа, это значение по умолчанию не отображается на экране (например, Новая заявка по телефону) и будет присвоено после создания заявки.

Для активации этой возможности необходимо включить параметр: "Ticket::EventModulePost###TicketDynamicFieldDefault".

Пример 4.27. Активировать Field1 в событии TicketCreate.

- *Группировать:* Заявку
- *Подгруппа:* Core::TicketDynamicFieldDefault
- *Настройка:* Ticket::TicketDynamicFieldDefault###Element1

Примечание

Эта настройка может быть установлена в любую из 16 Ticket::TicketDynamicFieldDefault###Element.

Если необходимо задать более 16 полей, пользовательский XML файл должен быть помещен в папку \$OTRS_HOME/Kernel/Config/files для расширения этой возможности.

- *Значение:*

Ключ	Content
Событие	TicketCreate
Название	Поле1
Значение	новое значение

10.2.12. Установить значение по умолчанию для Пользовательских Настроек

Значение по умолчанию для динамического поля может быть перезаписано значением, определенным пользователем, которое хранится в пользовательских настройках.

При использовании этого способа, значение поля по умолчанию будет показываться на всех экранах, для которых оно активировано (если оно уже не получило другого значения).

Параметр "sysconfig" - "PreferencesGroups###DynamicField" расположен в подгруппе "Frontend::Agent::Preferences". В этом параметре задан пример того, как создать запись для экрана личных настроек агента специального значения по умолчанию для динамического поля, для избранного пользователя. Этот параметр имеет ограничение - он разрешает задать значение только для одного динамического поля. Если это требуется сделать для двух и более полей, необходимо создать собственный XML файл конфигурации для добавления дополнительных настроек, аналогичных этой.

Примечание

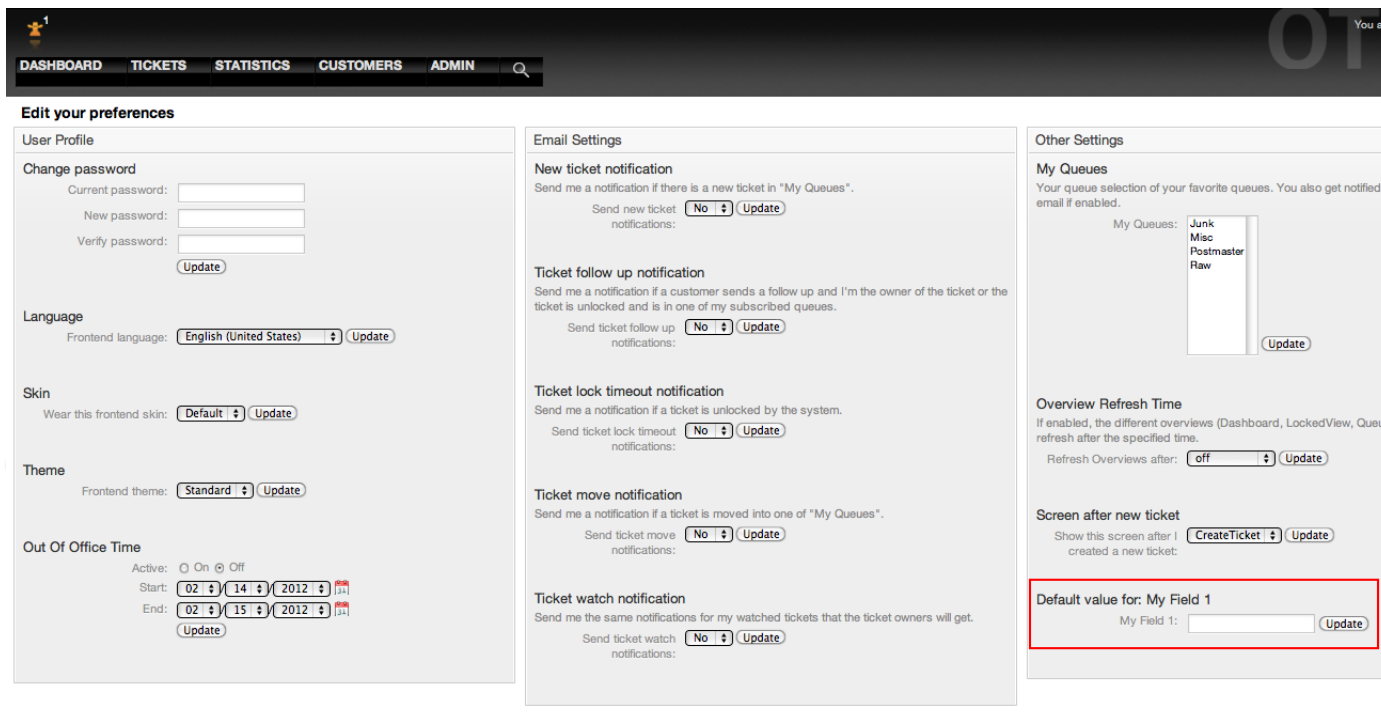
Необходимо помнить, что если добавляются эти настройки в XML файл, каждое имя параметра должно быть уникальным в системе и отличаться от "PreferencesGroups###DynamicField". Например, PreferencesGroups###101-DynamicField-Поле1, PreferencesGroups###102-DynamicField-Поле2, PreferencesGroups###Мое-Поле1, PreferencesGroups###Мое-Поле2, и т.д..

Пример 4.28. Активировать поле Field1 на странице Пользовательские Настройки.

- *Группировать:* Заявку
- *Подгруппа:* Frontend::Agent::Preferences
- *Настройка:* PreferencesGroups###101-DynamicField-Field1
- *Значение:*

Ключ	Content
Событие	TicketCreate
Активный	1
Block	Input
Поле	Другие настройки

Ключ	Content
Data:	<code>\$Env{"UserDynamicField_Field1"}</code>
Ключ:	Мое Поле 1
Метка:	Значение по умолчанию для: My Field 1
Module:	Kernel::Output::HTML::PreferencesGeneric
PrefKey:	UserDynamicField_Field1
Prio:	7000



The screenshot shows the OTRS user preferences page. The navigation bar includes Dashboard, Tickets, Statistics, Customers, and Admin. The main content area is titled 'Edit your preferences' and is divided into several sections:

- User Profile:** Change password (Current, New, Verify), Language (Frontend: English (United States)), Skin (Wear this frontend skin: Default), Theme (Frontend theme: Standard), and Out Of Office Time (Active: On/Off, Start: 02:14:2012, End: 02:15:2012).
- Email Settings:** New ticket notification, Ticket follow up notification, Ticket lock timeout notification, Ticket move notification, and Ticket watch notification. Each has a 'Send ... notifications:' field with a 'No' dropdown and an 'Update' button.
- Other Settings:** My Queues (Junk, Misc, Postmaster, Raw), Overview Refresh Time (Refresh Overviews after: off), Screen after new ticket (Show this screen after: CreateTicket), and Default value for: My Field 1 (My Field 1: [input field]).

The 'Default value for: My Field 1' section is highlighted with a red border in the original image.

Powered by OTRS 3.1.x CVS

Рисунок: Field1 в окне с пользовательских настроек агентов.

11. Generic Interface

OTRS Generic Interface состоит из многоуровневого интерфейса, позволяющего OTRS взаимодействовать с другими системами с помощью веб-сервисов. Это взаимодействие может быть двунаправленной:

- *OTRS as Provider:* OTRS acts as a server listening to requests from the External System, processing the information, performing the requested action, and answering the request.
- *OTRS as Requester:* OTRS acts as a client collecting information, sending the request to the Remote System, and waiting for the response.

11.1. Слои интерфейса Generic Interface

Generic Interface is build based on a layer model, to be flexible and easy to customize.

A layer is a set of files, which control how the Generic Interface performs different parts of a web service. Using the right configuration, one can build different web services for different External Systems without creating new modules.

Примечание

If the Remote System does not support the current bundled modules of the Generic Interface, special modules need to be developed for that specific web service.

Перечень модулей предоставляемых Generic Interface поставляется вместе с OTRS и будет обновлен или увеличен с течением времени.

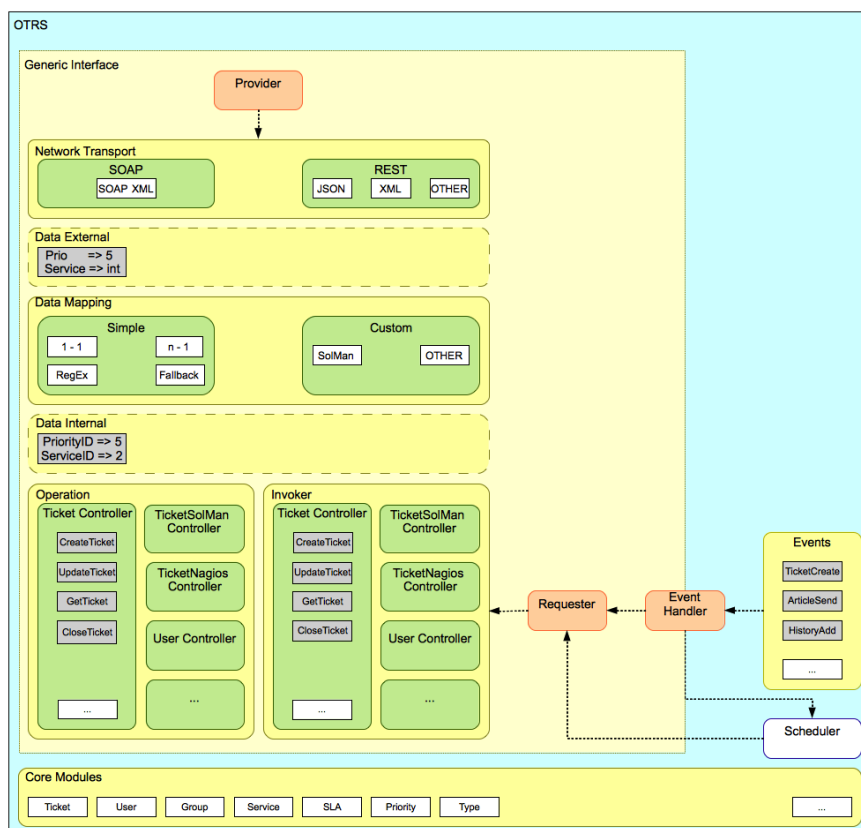


Рисунок: Слои графического интерфейса

11.1.1. Транспортная Сеть

This layer is responsible for the correct communication with the Remote System. It receives requests and generates responses when acting as provider, and generates requests and receives responses when acting as requester.

Provider communication is handled by a new web server handle called "nph-genericinterface.pl".

Requester communication could be initiated during an event triggered by a Generic Interface module or any other OTRS module. This event is caught by the event handler and depending on the configuration the event will be processed directly by the requester object or delegated to the Scheduler (a separated daemon designed to process tasks asynchronously).

11.1.2. Отображение Данных

This layer is responsible for translating data structures between OTRS and the Remote System (data internal and data external layers). Usually Remote Systems have different data structures than OTRS (including different values and names for those values),

and here resides the importance of the layer to change the received information into something that OTRS can understand and on the opposite way send the information to each Remote System using their data dictionaries.

Example: "Priority" (OTRS) might be called "Prio" in a remote system and it could be that value "1 Low" (OTRS) should be mapped to "Information" on the remote system.

11.1.3. Контроллер

Controllers are collections of similar Operations or Invokers. For example, a Ticket controller might contain several standard ticket operations. Custom controllers can be implemented, for example a "TicketExternalCompany" controller which may contain similar functions as the standard Ticket controller, but with a different data interface, or function names (to adapt to the Remote System function names) or complete different code.

One application for Generic Interface could be to synchronize information with one Remote System that only can talk with another Remote System of the same kind. In this case new controllers needs to be developed and the Operations and Invokers has to emulate the Remote System behavior in such way that the interface that OTRS exposes is similar to the Remote System's interface.

11.1.4. Операция (OTRS как поставщик услуг)

An Operation is a single action that can be performed within OTRS. All operations have the same programming interface, they receive the data into one specific parameter, and return a data structure with a success status, potential error message and returning data.

Normally operations uses the already mapped data (internal) to call core modules and perform actions in OTRS like: Create a Ticket, Update a User, Invalidate a Queue, Send a Notification, etc. An operation has full access to the OTRS API to perform the action.

11.1.5. Invoker - запрашивающая сторона (OTRS как запрашивающая сторона, клиент)

An Invoker is an action that OTRS performs against a Remote System. Invokers use the OTRS Core modules to process and collect the needed information to create the request. When the information is ready it has to be mapped to the Remote System format in order to be sent to the Remote System, that will process the information execute the action and send the response back, to either process the success or handle errors.

11.2. Поток связи Generic Interface

The Generic Interface has a defined flow to perform actions as a provider and as a requester.

Эти потоки описаны ниже:

11.2.1. OTRS как Поставщик

11.2.1.1. Удаленный Запрос:

1. HTTP-запрос

- OTRS receives HTTP request and passes it through the layers.
- The provider module is in charge to execute and control these actions.

2. Транспортная Сеть

- The network transport module decodes the data payload and separates the operation name from the rest of the data.
- Название операции и данные операции будут возвращены поставщику.

3. Внешние данные

- Данные отправленные из удаленной системы (Это не слой базирующийся на модуле).

4. Mapping

- Данные преобразуются из формата Внешней Системы во внутренний формат данных OTRS так, как это указано в конфигурации отображения для этой операции (Mapping для входящих данных запроса).
- Преобразованные данные возвращаются обратно поставщику.

5. Внутренние Данные

- Data as transformed and prepared to be passed to the operation (This is not a module based layer).

6. Операция

- Принимает и проверяет данные.
- Осуществляет контроль доступа пользователей.
- Выполняет действие.

11.2.1.2. OTRS Ответ:

1. Операция

- Возвращает результирующие данные поставщику.

2. Внутренние Данные

- Данные возвращаемые из операции.

3. Mapping

- The data is transformed back to the Remote system format as specified in the mapping configuration (Mapping for outgoing response data).
- Преобразованные данные возвращаются обратно поставщику.

4. Внешние данные

- Data as transformed and prepared to be passed to Network Transport as response.

5. Транспортная Сеть

- Получает данные уже в формате Удаленной Системы.
- Создает правильный ответ для этого типа транспортной сети.

6. HTTP ответ

- Ответ отправляется обратно клиенту веб-сервиса.

- В случае ошибки, сообщение об ошибке отправляется удаденной системе (например SOAP-ошибка, HTTP-ошибка и т.д.).

11.2.2. OTRS как запрашиваемая сторона

11.2.2.1. OTRS Запрос:

1. Обработчки Event Trigger

- В зависимости от настройки веб-сервис определяет будет ли запрос синхронным или асинхронным.
 - Синхронный
 - A direct call to the Requester is made in order to create a new request and to pass it through the layers.
 - Асинхронный
 - Create a new Generic Interface (Requester) task for the OTRS Scheduler (by delegating the request execution to the Scheduler, the user experience could be highly improved, otherwise all the time needed to prepare the request and the remote execution will be added to the OTRS Events that trigger those requests).
 - In its next cycle the Scheduler process reads the new task and creates a call to the Requester that will create a new request and then passes it through the layers.

2. Запрашивающая сторона

- Получает данные от событий.
- Проверяет полученные данные (при необходимости).
- Вызвать модули ядра для дополнения данных (при необходимости).
- Возвращает структуру данных запроса или отправляет сигнал "Остановка связи" для Запрашивающей Стороны (requester), чтобы корректно отменить запрос.

3. *Внутренние Данные*

- Данные передаются от запрашивающей стороны (Это не слой, основанный на модуле).

4. Mapping

- The data is transformed to the Remote system format as specified in the mapping configuration (Mapping for outgoing response data).
- Уже преобразованные данные возвращаются Запрашивающей Стороне.

5. *Внешние данные*

- Данные преобразованы и подготовлены для отправки удаленной системе.

6. Транспортная Сеть

- Receives the remote operation name and the data already transformed to the Remote System format from the requester.
- Создает действительный запрос для транспортной сети.

- Отправляет запрос удаленной системе и ожидает ответ от нее

11.2.2.2. Удаленный Ответ

1. Транспортная сеть

- Receives the response and decodes the data payload.
- Возвращает данные запрашивающей стороне.

2. Внешние данные

- Данные полученные от Удаленной Системы

3. Mapping

- Данные преобразуются из формата внешней системы во внутренний формат данных OTRS так, как это указано в конфигурации отображения для этой операции (Mapping для входящих данных отклика).
- Уже преобразованные данные возвращаются Запрашивающей Стороне.

4. Внутренние Данные

- Data as transformed and ready to be passed back to the requester.

5. Запрашивающая сторона

- Получает возвращаемые данные.
- Handles the data as needed specifically by each Invoker (included error handling if any).
- Return the Invoker result and data to the Requester.

6. Обработчик или Планировщик Действий

- Receives the data from the Requester, in the case of the Scheduler this data might contain information to Re-Schedule the task immediately or in the future.

11.3. Веб-сервисы

Веб-сервисы это метод взаимодействия между двумя системами, в нашем случае, OTRS и удаленной системой/Remote System.

The heart of the Web Service is its configuration, where it is defined what actions the web service can perform internally (Operation), what actions the OTRS request can perform Remote System (Invokers), how data is converted from one system to the other (Mapping), and over which protocol the communication will take place (Transport)

The Generic Interface is the framework that makes it possible to create Web Services for OTRS in a pre-defined way, using already made building blocks that are independent from each other and interchangeable.

11.4. Графический интерфейс веб-сервиса

Графический пользовательский интерфейс (GUI) является тем инструментом, который позволяет создавать сложные настройки веб-сервиса в дружелюбной и приятной среде интерфейса. Он позволяет:

- Создание и Удаление веб-сервисов

- Импортировать и Экспортировать настройки (в формате YAML) для существующих веб-сервисов.
- Просматривать, Возвращаться и Экспортировать устаревшие конфигурационные параметры для существующих веб-сервисов на странице истории конфигурации.
- Отслеживать все сообщения журнала для каждого из веб-сервисов на странице Отладчика.

11.4.1. Обзор Веб-интерфейса

Ссылка "Веб-Сервисы/Web Services" на главном экране интерфейса администратора (Панели Администратора) осуществляет переход к экрану обзора веб-сервисов, с помощью которого, вы можете управлять их настройками. Вы можете добавлять новые или изменять настройки существующих на этом экране.

Every web service configuration screen has in the upper part of the screen a "bread crumbs" style navigation path. This navigation path is useful to know exactly in which part of the web service configuration we are, and also enables the user to jump back to any part of the configuration process at any time (this action will not save any changes).

Примечание

Чтобы создать новый веб-сервис, нажмите кнопку "Добавить веб-сервис", введите все необходимые данные.

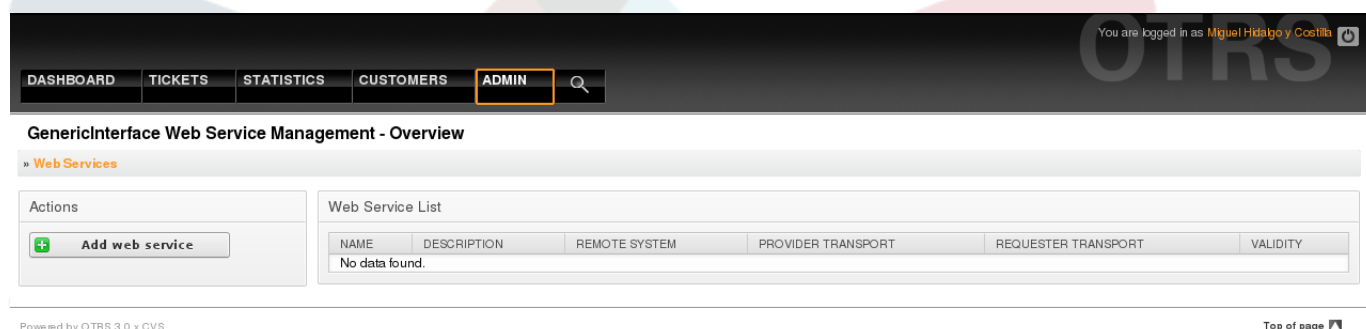


Рисунок: Обзор Веб-сервисов

11.4.2. Добавить Веб Сервис

Единственное поле в этой части - это "Имя" веб-сервиса, которое должно быть уникальным в системе и не может быть пустым. Остальные поля также необходимы

для настройки, такие как "Уровень отладки/Debug Threshold" и "Действительность", но эти поля уже заполнены значениями по умолчанию, для каждого списка.

Значением по умолчанию для поля "Уровень отладки/Debug Threshold" является "debug". Каждое последующее значение "Уровень отладки/Debug Threshold" является более строгим и отбрасывает журналы связи более низкого порядка, чем то, что установлено в системе ранее.

Отладка уровней (от нижних к верхним)

- Отладка
- Информация
- Уведомление
- Ошибка

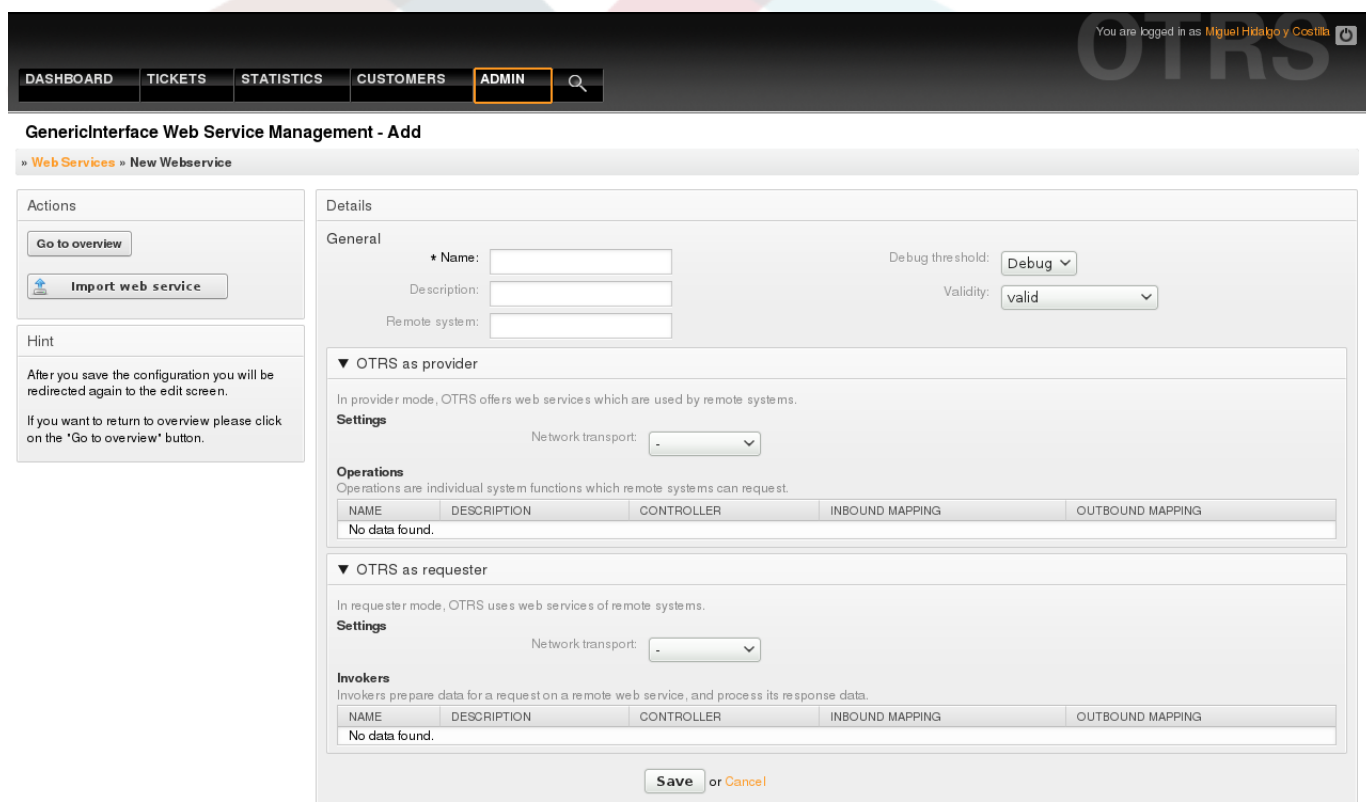
Также есть возможность определить протокол транспортной сети для "OTRS как Поставщика" и "OTRS как Запрашиваемую Сторону".

Нажмите кнопку "Сохранить", чтобы зарегистрировать новый веб-сервис в базе данных или кнопку "Отменить" чтобы отменить эту операцию.

Если у вас уже имеется файл конфигурации веб-сервисов в формате YAML, вы можете импортировать его, нажав на кнопку "Импортировать веб-сервис" слева на экране. Для дальнейшей информации об импорте веб-сервисов обратитесь к следующей секции "Изменение Веб-сервисов/Web Service Change".

Примечание

Нажмите на имя веб-сервиса на странице просмотра чтобы изменить или добавить больше элементов веб-сервису.



The screenshot shows the OTRS Admin interface for adding a new web service. The top navigation bar includes Dashboard, Tickets, Statistics, Customers, and Admin. The main content area is titled "GenericInterface Web Service Management - Add" and "Web Services » New Webservice".

Actions: Go to overview, Import web service.

Hint: After you save the configuration you will be redirected again to the edit screen. If you want to return to overview please click on the 'Go to overview' button.

Details:

General:

- Name:
- Description:
- Remote system:
- Debug threshold: Debug
- Validity: valid

OTRS as provider:

In provider mode, OTRS offers web services which are used by remote systems.

Settings: Network transport: -

Operations: Operations are individual system functions which remote systems can request.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
No data found.				

OTRS as requester:

In requester mode, OTRS uses web services of remote systems.

Settings: Network transport: -

Invokers: Invokers prepare data for a request on a remote web service, and process its response data.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
No data found.				

Save or Cancel

Рисунок: Добавление нового веб-сервиса

11.4.3. Изменения веб-сервиса:

На этом экране имеется полный набор по управлению каждой частью веб-сервиса. Слева, в колонке "Действия" вы найдете несколько кнопок, позволяющих выполнить все доступные действия над веб-сервисом:

- Клонирование веб-сервиса.
- Экспортировать веб-сервис.
- Импортировать веб-сервис.
- История конфигурации.
- Удалить веб-сервис.
- Отладчик.

Примечание

Переход к "История конфигурации" и "Отладчик" приведет к открытию различных страниц.

11.4.3.1. Клонирование веб-сервиса:

Для клонирования веб-сервиса необходимо нажать кнопку "Клонировать веб-сервис", появится диалоговое окно, в котором можно использовать имя по умолчанию или же установить новое имя для (клонированного) веб-сервиса.

Примечание

Запомните название веб-сервиса должно быть уникальным для всей системы.

Нажмите кнопку "Клонировать" чтобы создать копию веб-сервиса или кнопку "Отменить", чтобы закрыть диалоговое окно.

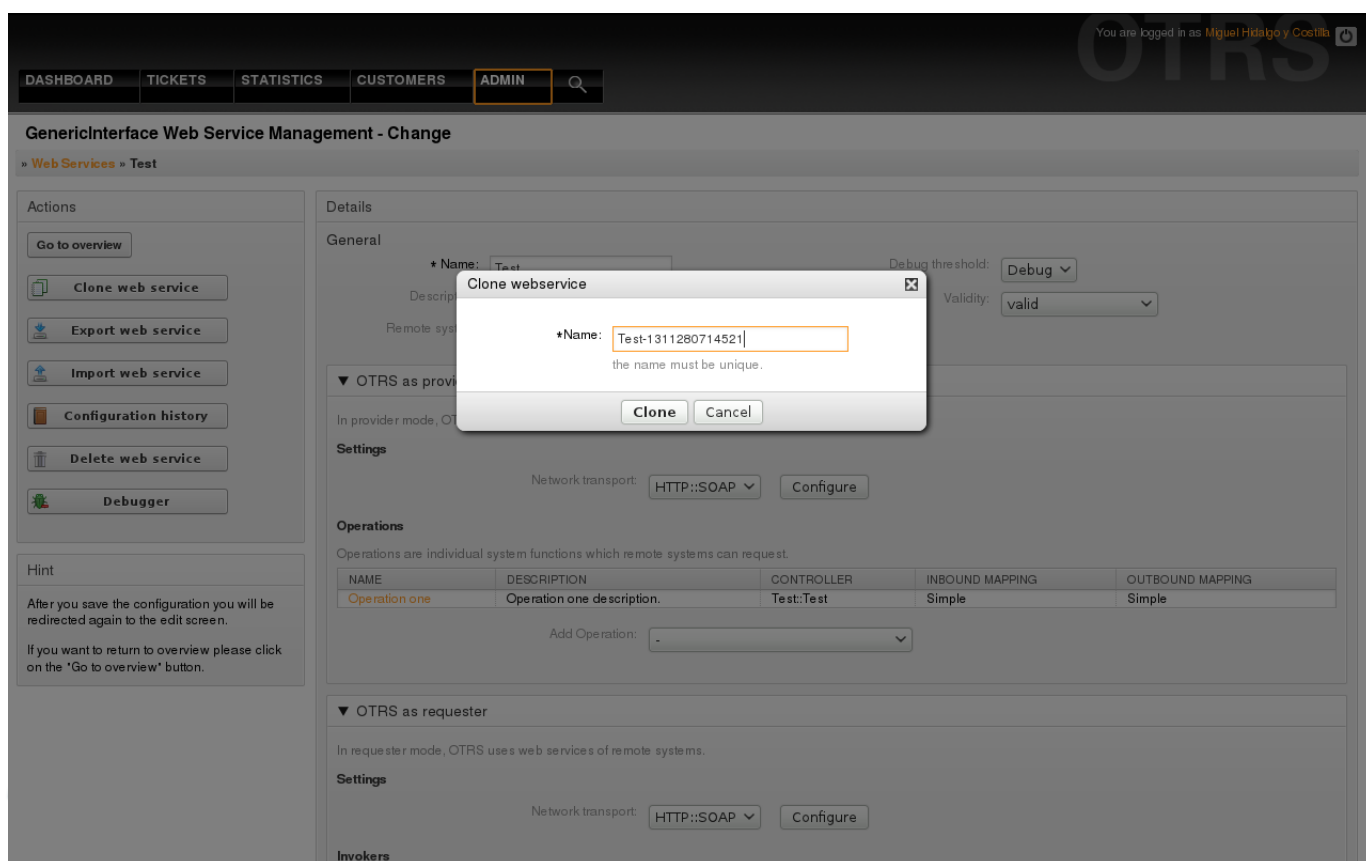


Рисунок: Клонирование веб-сервиса.

11.4.3.2. Экспорт веб-сервиса

Кнопка "Экспорт веб-сервиса/Export web service" дает возможность выгрузить конфигурацию текущего веб-сервиса в файл формата YAML, для загрузки его и сохранения в файловой системе. Это может быть особенно полезно, если вы желаете перенести веб сервис с одного сервера на другой, к примеру, из тестовой среды в рабочую.

Предупреждение

Все сохраненные пароли в настройках веб-сервиса будут экспортированы в виде обычного текста.

Сразу же после нажатия кнопки "Экспорт веб-сервиса" появится диалоговое окно сохранения, на подобие того, которое появляется когда вы нажимаете в браузере ссылку сохранить.

Примечание

Каждый браузер в каждой операционной системе имеет свои собственные экраны диалога при сохранении и их вид. В зависимости от браузера и его настроек, возможен вариант, когда диалог при сохранении не отображается и файл сохраняется в каталоге по умолчанию для вашей файловой системы. Обратитесь к документации на браузер для уточнения этих моментов.

GenericInterface Web Service Management - Change

» **Web Services** » Test

Actions

Go to overview

Clone web service

Export web service

Import web service

Configuration history

Delete web service

Debugger

Hint

After you save the configuration you will be redirected again to the edit screen.

If you want to return to overview please click on the 'Go to overview' button.

Details

General

+ Name: Test

Description:

Remote system:

OTRS as provider:

In provider mode, OTRS should: Debug

Validity: valid

Settings

What should Firefox do with this file?

Open with: gedit (default)

Save File

Do this automatically for files like this from now on.

Cancel OK

Operations

Operations are individual

NAME	INBOUND MAPPING	OUTBOUND MAPPING
Operation one	Simple	Simple

Add Operation: -

OTRS as requester

Save Save and finish or Cancel

Рисунок: Экспорт веб-сервиса.

11.4.3.3. Импорт веб-сервиса:

Для загрузки имеющегося YAML файла конфигурации веб-сервиса требуется использование функции импорта. Нажмите кнопку "Импорт веб-сервиса", найдите нужный файл или укажите полный путь к нему в окне ввода

Нажмите кнопку "Импорт" чтобы создать новый веб-сервис с файла или кнопку "Отменить" чтобы закрыть диалоговое окно.

Примечание

Имя веб-сервиса будет взято из имени конфигурационного файла (например, если имя файла MyWebservice.yml, то имя веб-сервиса будет - MyWebservice). Если в системе уже зарегистрирован веб-сервис с таким именем, система открывает экран изменения веб-сервиса, чтобы вы могли поменять имя импортируемого веб-сервиса.

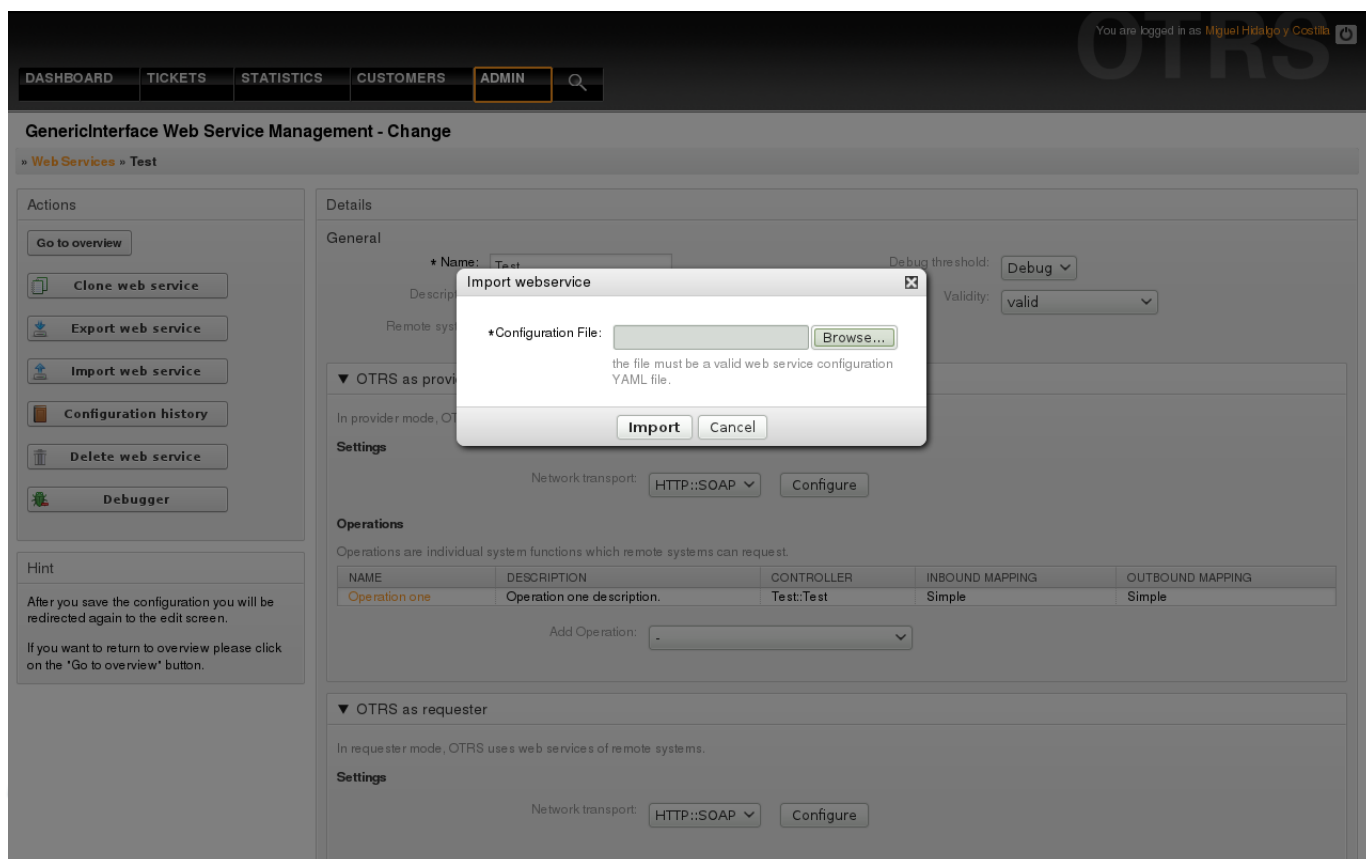


Рисунок: Импорт веб-сервиса.

11.4.3.4. История веб-сервиса

Каждое изменение настроек веб-сервиса порождает новую запись в истории веб-сервиса (типа журнала). Экран истории веб-сервиса отображает список всех версий конфигурации для веб-сервиса. Каждая строка (версия) в "Истории конфигурации" представляет одно изменение в истории веб-сервиса.

Нажатие на конкретную строку приводит к показу всей конфигурации на конкретную дату/время. Конфигурация отображается в секции "Детали истории" на этом экране. На этом экране вы также имеете возможность экспортировать выбранную версию конфигурации веб-сервиса или восстановить эту версию, сделав ее текущей.

"Экспортировать настройки веб-сервиса" ведет себя точно так же как "Экспортировать веб-сервис". Для более подробной информации обратитесь к этому разделу.

Если сделанные изменения в конфигурации веб-сервиса приводят к неправильным результатам и исправить вручную сделанные изменения затруднительно, вы можете нажать на кнопку "Вернуть конфигурацию веб-сервиса". Нажатие приведет к открытию диалогового окна с вопросом, действительно ли вы желаете вернуться к предыдущей версии. Нажмите на кнопку "Вернуть конфигурацию веб-сервиса" в этом окне для замены текущей конфигурации выбранной версией или нажмите "Отменить" для закрытия окна.

Предупреждение

Помните, что все пароли сохраненные в конфигурации веб-сервиса будут экспортироваться в виде обычного текста.

Пожалуйста, будьте внимательны при восстановлении параметров конфигурации, эта операция не может быть отменена.

You are logged in as **Miguel Hidalgo y Costilla**
DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

GenericInterface Configuration History for Web Service Test

» Web Services » Test » History

Actions

[Go back to Web Service](#)

Hint

Here you can view older versions of the current web service's configuration, export or even restore them.

Configuration History List

VERSION	CREATE TIME
5	2011-07-21 15:23:02
4	2011-07-21 13:57:38
3	2011-07-21 13:48:04
2	2011-07-21 13:36:14
1	2011-07-21 13:33:11

Select a single configuration version to see its details.

History Details: Version 3, 2011-07-21 13:48:04

Export web service configuration | Restore web service configuration

```

---
Debugger:
  DebugThreshold: debug
  TestMode: 0
Description: A test web service config
Provider:
  Operation:
    Operation one:
      Description: Operation one description.
      MappingInbound:
        Type: Simple
      MappingOutbound:
        Type: Simple
      Type: Test::Test
  Transport:
    Config:
      Authentication: {}
    Type: HTTP::SOAP
  RemoteSystem: remote
  Requester:
    Transport:
      Config:
        Authentication: {}
      Type: HTTP::SOAP
          
```

Powered by OTRS 3.0.x CVS
Top of page

Рисунок: История веб-сервиса

11.4.3.5. Удаление веб-сервиса

Иногда возникает необходимость удалить веб-сервис полностью. Для этого можно нажать кнопку "Удалить веб-сервис", появится диалоговое окно с вопросом о подтверждении операции.

Кликните на кнопку "Удалить" чтобы подтвердить удаление веб-сервиса или кнопку "Отменить" чтобы закрыть диалоговое окно.

Предупреждение

Операция удаления веб-сервиса не может быть отменена, пожалуйста, будьте внимательны при удалении веб-сервисов.

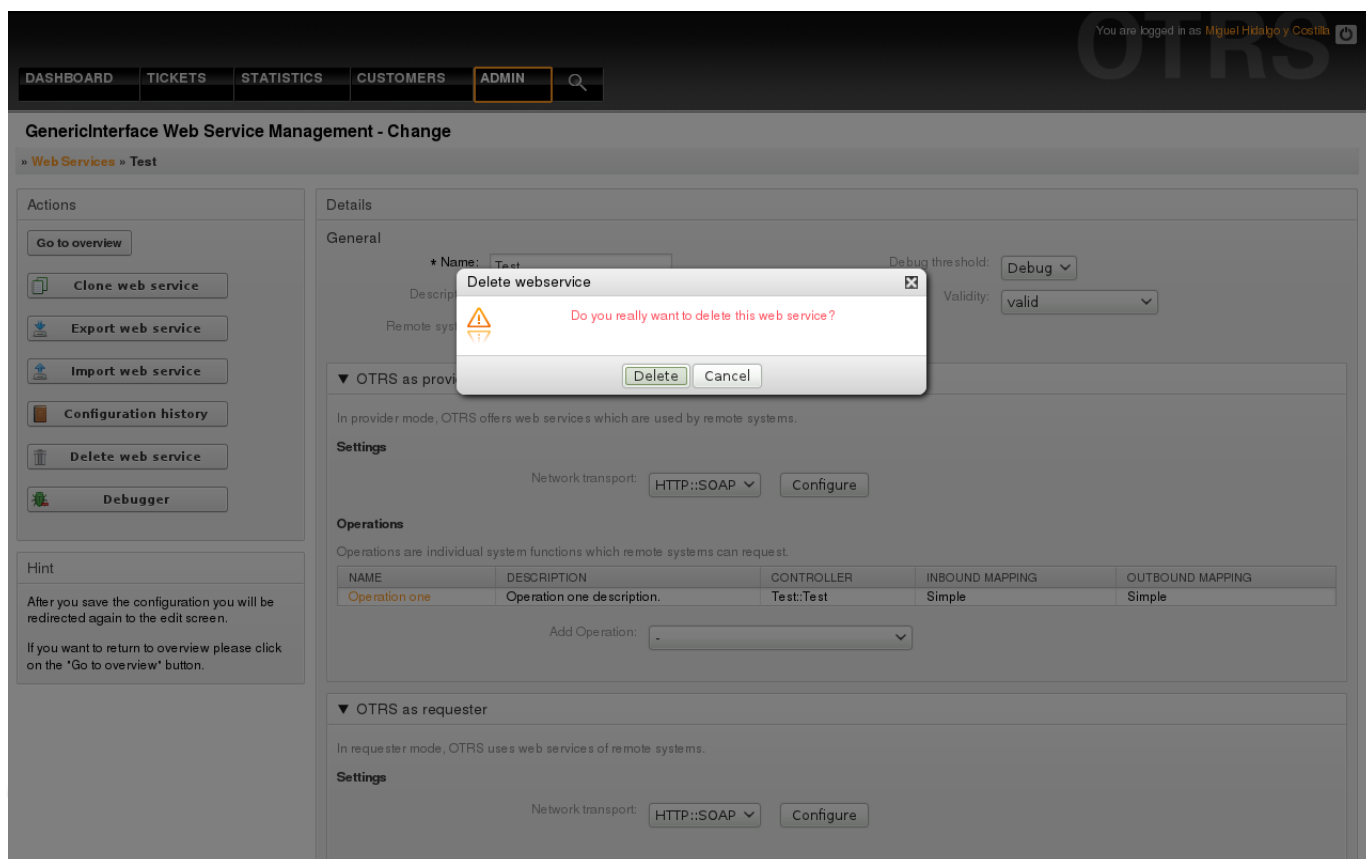


Рисунок: Удаление веб-сервиса

11.4.3.6. Отладчик веб-сервиса:

Отладчик хранит журнал веб-сервиса. На странице отладчика можно следить за всеми связями веб-сервера, когда он работает в роли поставщика и получателя.

Когда отображается эта страница начинает загружаться список запросов. После того как список будет полностью загружен, можно выбрать одну из строк (что означает последовательность связей) и просмотреть подробную информацию. Эта информация будет отображаться во всплывающем окне ниже.

Можно сузить список используя фильтр в правой части страницы. Фильтрацию можно производить по:

- Тип связи (поставщик или запрашиваемая сторона)
- Дата: до и / или после определенной даты
- Удаленный IP-адрес
- Сочетание всех.

После того как настройки фильтра завершены, нажмите кнопку "Обновить" и новый список, отвечающий вашим критериям будет отображен.

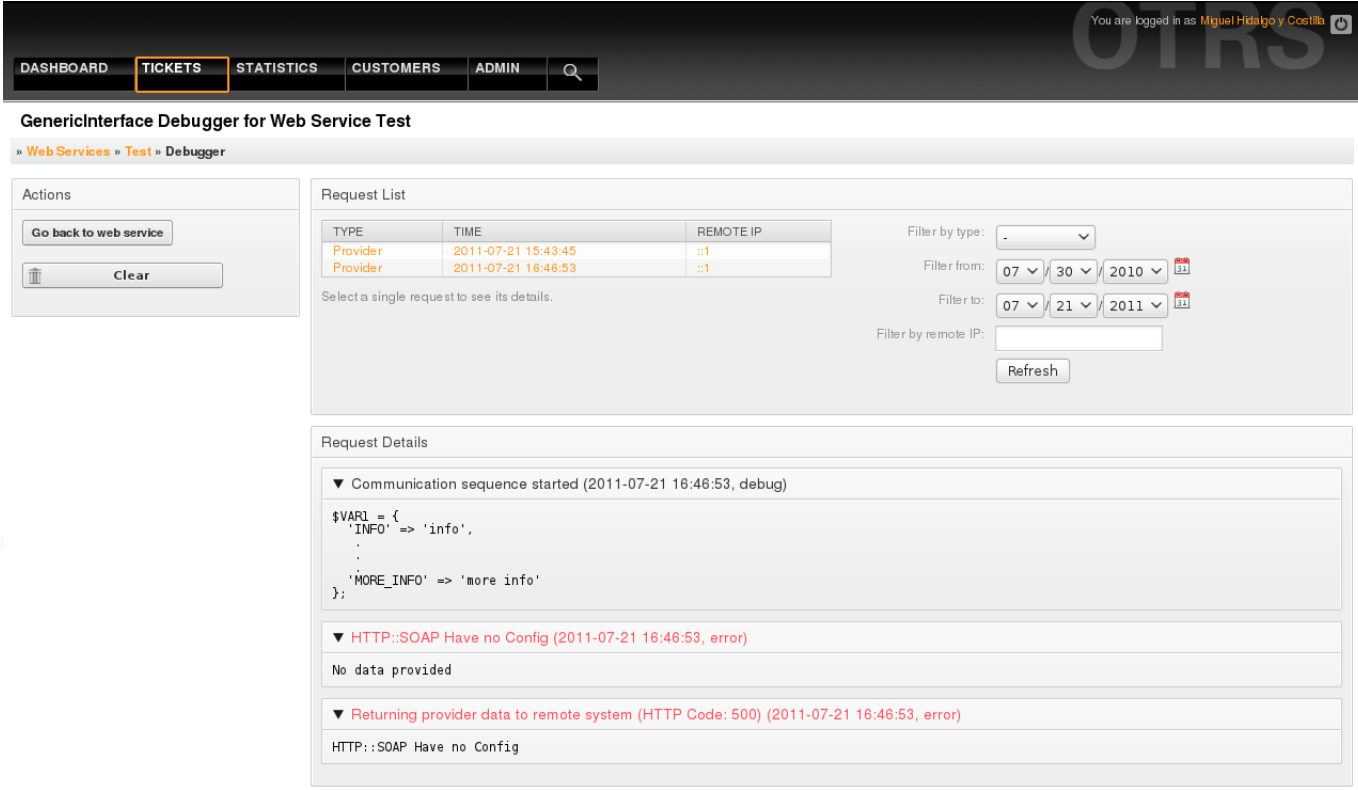
Примечание

В зависимости от критериев поиска установленных в фильтре, новый поиск может вернуть пустой список.

В левой части экрана, под фреймом действий, вы можете выбрать "Вернуться к веб-сервису" или очистить журнал отладки, нажатием на кнопку "Очистить". При

этом откроется диалоговое окно, где вас попросят подтвердить очистку журнала. Нажмите "Очистить" в диалоговом окне для выполнения этого действия или "Отменить" для закрытия диалога.

В разделе "Подробности Запроса" можно просматривать подробную информацию о коммуникации. Здесь можно следить за полным потоком и проверять всевозможные ошибки или подтверждать успешные ответы.



GenericInterface Debugger for Web Service Test

» Web Services » Test » Debugger

Actions

Go back to web service

Clear

TYPE	TIME	REMOTE IP
Provider	2011-07-21 15:43:45	:::1
Provider	2011-07-21 16:46:53	:::1

Filter by type: -

Filter from: 07 / 30 / 2010

Filter to: 07 / 21 / 2011

Filter by remote IP:

Refresh

Request Details

▼ Communication sequence started (2011-07-21 16:46:53, debug)

```
$VAR1 = {
  'INFO' => 'info',
  .
  .
  'MORE_INFO' => 'more info'
};
```

▼ HTTP::SOAP Have no Config (2011-07-21 16:46:53, error)

No data provided

▼ Returning provider data to remote system (HTTP Code: 500) (2011-07-21 16:46:53, error)

HTTP::SOAP Have no Config

Powered by OTRS 3.0.x CVS Top of page

Рисунок: Отладчик веб-сервиса.

11.4.3.7. Изменение настроек веб-службы

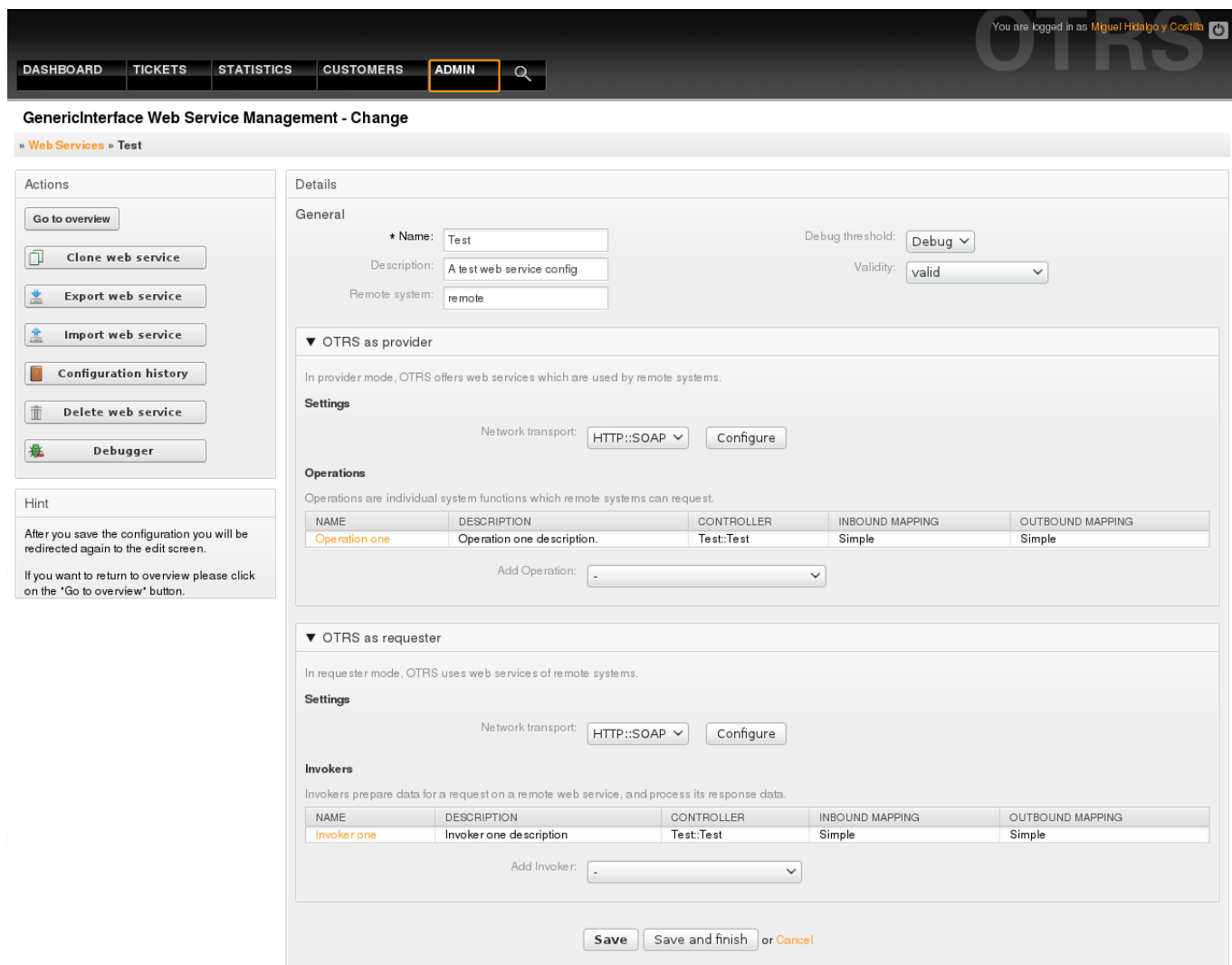
Возвратившись на экран изменения, перейдем к обзору правой его части. Здесь вы имеете возможность изменить все основные данные веб-сервиса, такие как имя, уровень отладки и т.д. Также, здесь, ниже, имеются еще две секции позволяющие изменить специальные параметры для типа связи "OTRS как провайдер" и "OTRS как получатель".

The web service configuration needs to be saved on each level. This means that if a setting is changed, links to other, deeper parts of the configuration will be disabled forcing you to save the current configuration level. After saving the disabled links will be re-enabled again allowing you to continue with the configuration.

On the "OTRS as provider" section it is possible to set or configure the network transport protocol. Only network transport back-ends that are registered are shown on the list. To configure the network transport click on the "Configure" button. It is also possible to add new operations in this box. To do this select one of the available operations from the "Add Operation" list. This will lead you to the operation configuration screen. After saving the new operation it will be listed in the table above.

"OTRS как запрашивающая сторона" очень похож на предыдущий, но вместо "операций" здесь можно добавить invocers

Нажмите кнопку "Сохранить" для сохранения и продолжения настройки веб-сервиса. "Сохранить и продолжить" - для сохранения и возврата в окно обзора веб-сервисов, или "Отменить" для отмены, сделанных на этом уровне, изменений и возврата к окну обзора веб-сервисов.



GenericInterface Web Service Management - Change

» Web Services » Test

Actions

- Go to overview
- Clone web service
- Export web service
- Import web service
- Configuration history
- Delete web service
- Debugger

Hint

After you save the configuration you will be redirected again to the edit screen.
If you want to return to overview please click on the "Go to overview" button.

Details

General

* Name: Debug threshold:

Description: Validity:

Remote system:

OTRS as provider

In provider mode, OTRS offers web services which are used by remote systems.

Settings

Network transport:

Operations

Operations are individual system functions which remote systems can request.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
Operation one	Operation one description.	Test:Test	Simple	Simple

Add Operation:

OTRS as requester

In requester mode, OTRS uses web services of remote systems.

Settings

Network transport:

Invokers

Invokers prepare data for a request on a remote web service, and process its response data.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
Invoker one	Invoker one description	Test:Test	Simple	Simple

Add Invoker:

or

Рисунок: Изменения веб-службы.

Примечание

Like the other Generic Interface configuration screens such as Network Transport, Operation, Invoker and Mapping, the initial configuration (add) screen will only present two options: "Save" and "Cancel". If the configuration is re-visited, a new option "Save and Finish" will appear. The behavior of this feature is defined below.

"Сохранить и Закончить" сохранит настройки текущего уровня в базе данных и вернет к предыдущей странице в иерархии конфигурации.

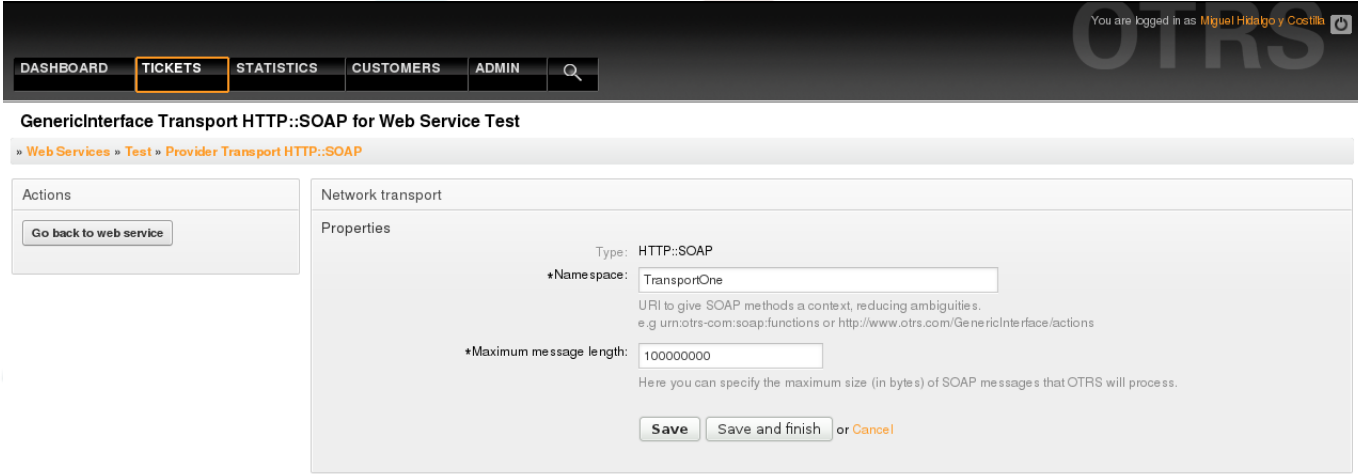
"Save and Finish" сохранит текущий уровень настроек в базе данных и вернется к предыдущей странице в иерархии конфигурации.

"Отменить" будет отменять любые изменения конфигурации и возвращать их к текущему уровню конфигурации, после чего возвращаться на предыдущему экрану в иерархии настроек.

11.4.3.7.1. Транспортная сеть Веб-сервиса

In future the list of available network transports will be increased. Currently only the "HTTP::SOAP" transport is available. Each transport has different configuration options to setup and they might use different frontend modules to configure it, but mostly they should look similar to the "HTTP::SOAP" transport configuration module.

It is quite simple to configure the "HTTP::SOAP" protocol as provider. There are only two settings: "Namespace" and "Maximum message length". These fields are required. The first one is a URI to give SOAP methods a context, reducing ambiguities, and the second one is a field where you can specify the maximum size (in bytes) for SOAP messages that OTRS will process.



The screenshot shows the OTRS web interface. At the top right, it says "You are logged in as Miguel Hidalgo y Costilla". The navigation menu includes "DASHBOARD", "TICKETS" (highlighted), "STATISTICS", "CUSTOMERS", and "ADMIN". The main content area is titled "GenericInterface Transport HTTP::SOAP for Web Service Test" and has a breadcrumb trail: "» Web Services » Test » Provider Transport HTTP::SOAP".

On the left, there is an "Actions" section with a button "Go back to web service". The main configuration area is titled "Network transport" and "Properties". It shows:

- Type: HTTP::SOAP
- *Namespace: TransportOne
- *Maximum message length: 100000000

Below the "Maximum message length" field, there is a note: "Here you can specify the maximum size (in bytes) of SOAP messages that OTRS will process." At the bottom of the form, there are buttons for "Save", "Save and finish", and "or Cancel".

At the bottom left of the page, it says "Powered by OTRS 3.0.x CVS". At the bottom right, there is a "Top of page" link.

Рисунок: Транспортная сеть Веб-сервиса.

11.4.3.7.2. Операции веб-сервиса

The actions that can be performed when you are using OTRS as a provider are called "Operations". Each operation belongs to a controller. Controllers are collections of operations or invokers, normally operations from the same controller need similar settings and share the same configuration dialog. But each operation can have independent configuration dialogues if needed.

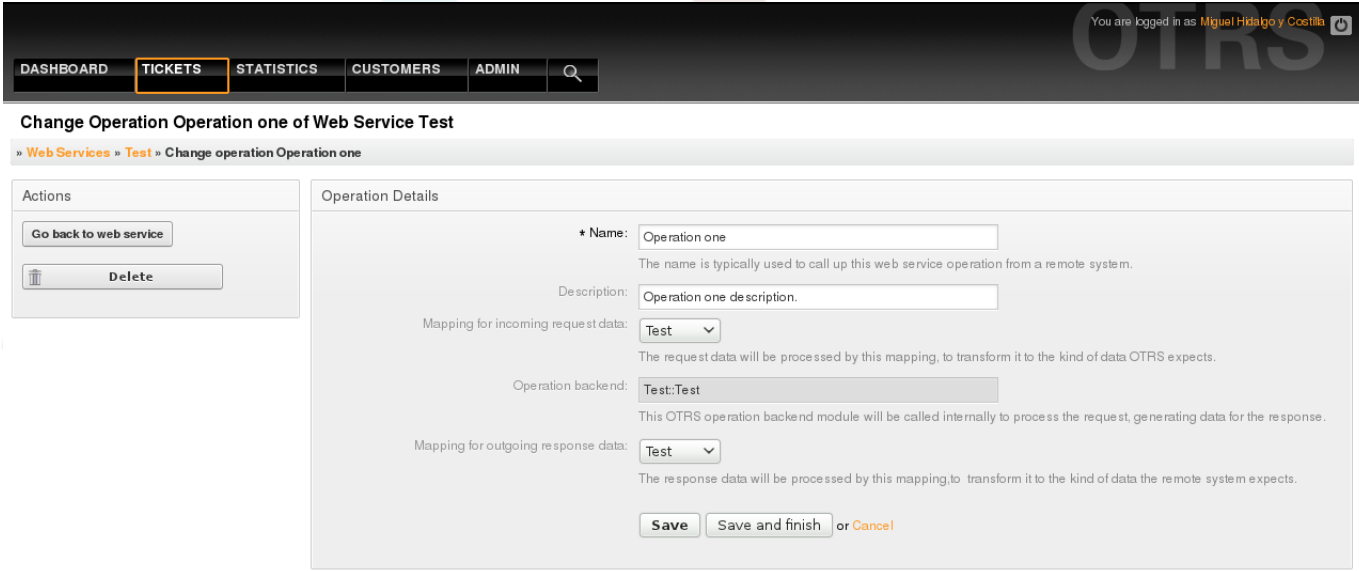
Name, Description, Backend, and Mappings are fields that normally appear on every operation, other special fields can appear in non default configuration dialogues to fulfill specific needs of the operation.

Normally there are two mapping configuration sections on each operation, one for the incoming data and another one for the outgoing data. You can choose different mapping types (backends) for each mapping direction, since their configuration is independent from each other and also independent from the operation backend. The normal and most

common practice is that the operation uses the same mapping type in both cases (with inverted configuration). The complete mapping configuration is done in a separate screen which depends on the mapping type.

The operation backend is pre-populated and is not editable. You will see this parameter when you choose the operation on the web service edit screen. The field is only informative.

In the left part of the screen on the action column you have the options: "Go back to web service" (discarding all changes since the last save) and "Delete". If you click on the last one, a dialog will open and ask you if you like to remove the operation. Click on "Delete" button to confirm the removal of the operation and its configuration or "Cancel" to close the delete dialog.



The screenshot shows the OTRS web interface. At the top, there is a navigation menu with 'DASHBOARD', 'TICKETS', 'STATISTICS', 'CUSTOMERS', and 'ADMIN'. The 'TICKETS' menu item is highlighted. Below the navigation, the page title is 'Change Operation Operation one of Web Service Test'. The breadcrumb trail is '» Web Services » Test » Change operation Operation one'. On the left, there is an 'Actions' panel with two buttons: 'Go back to web service' and 'Delete'. The main area is titled 'Operation Details' and contains the following fields:

- * Name: Operation one (Text input field)
- Description: Operation one description. (Text input field)
- Mapping for incoming request data: Test (Dropdown menu)
- Operation backend: Test:Test (Text input field)
- Mapping for outgoing response data: Test (Dropdown menu)

Below the fields, there are three buttons: 'Save', 'Save and finish', and 'Cancel'. The footer of the page indicates 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Рисунок: Операции веб-сервиса

11.4.3.7.3. Web Service Requester Network Transport

The network transport configuration for the requester is similar to the configuration for the provider. For the Requester "HTTP::SOAP" network transport there are more fields to be set.

Apart from the "Endpoint" (URI of the Remote System web service interface to accept requests) and "Namespace" which are required fields, you can also specify:

- Кодировка (такая как utf-8, latin1, iso-8859-1, cp1250, и т.д.) для SOAP-сообщений.
- SOAPAction Header: you can use this to send an empty or filled SOAPAction header. Set to "No" and the SOAPAction header on the SOAP message will be an empty string, or set to "Yes" to send the soap action in Namespace#Action format and define the separator (typically "/" for .Net web services and "#" for the rest).

- Аутентификация: установить механизм аутентификации, установлено в "-" чтобы не использовать любую проверку подлинности или выбрать ее из списка, после чего появится поле с подробной информацией.

Примечание

В настоящее время реализован только базовый механизм аутентификации "BasicAuth" (HTTP). Использовать его или нет, зависит от настроек удаленной системы. Если решите использовать, необходимо указать Имя пользователя и пароль для доступа к удаленному серверу.

Предупреждение

Если вы используете пароль для аутентификации, то при экспорте настроек веб-сервиса в YAML-файл этот пароль будет раскрыт и записан в виде простой текстовой строки внутри файла YAML. Знайте об этом и примите меры предосторожности в случае необходимости.

' data-bbox="131 336 986 708"/>

Figure: Web service requester network transport.

11.4.3.7.4. Вызов веб-сервиса:

The actions that can be performed when you are using OTRS as a requester are called "Invokers". Each invoker belongs to a controller (controllers are collections of operations or invokers). Usually invokers from the same controller need similar settings and share the same configuration dialogues. Each invoker can have independent configuration dialogues if needed.

Name, Description, Backend, and Mappings are fields that normally appear on every invoker. Additionally the list of event triggers and other special fields can appear on non default configuration dialogues to fulfill special needs of the invoker.

Normally there are two mapping configuration sections for each invoker, one for the incoming data and another one for the outgoing data. You can choose different mapping types (backends) for each mapping direction, since their configuration is independent from each other and also independent from the invoker backend. The normal and most common practice is that the invoker uses the same mapping type in both cases, with inverted configuration. The complete mapping configuration is done in a separate screen, which depends on the mapping type.

The invoker backend is pre-populated and can not be edited. You will see this parameter when you choose the invoker on the web service edit screen. The field is only informative.

Event triggers are events within OTRS such as "TicketCreate", "ArticleSend", etc. These can act as triggers to execute the invoker. Each invoker needs to have at least one event trigger registered, or the invoker will be useless, because it will never be called. The asynchronous property of the event triggers define if the OTRS process will handle the invoker or if it will be delegated to the Scheduler.

Примечание

The OTRS Scheduler is a separate process that executes tasks in the background. Using this the OTRS process itself will not be affected if the Remote System takes a long time to respond, if it is not available or if there are network problems. If you don't use the scheduler using web services can make OTRS slow or non-responsive. Therefore it is highly recommend to use asynchronous event triggers as often as possible.

To add an Event trigger, first select the event family from the first list, then the event name from the second list, then set the asynchronous property (if unchecked means that the event trigger will not be asynchronous) and finally click on the plus button. A new event trigger will be created and it will be listed on the invoker "Event Triggers" list.

To delete an Event trigger, simply locate the event trigger to be deleted in the "Event Triggers" list and click on the trash icon at the end of the row. This will open a dialog that asks you if you are sure to delete the event trigger. Click "Delete" to remove the event trigger from the list, or "Cancel" to close the dialog.

In the left part of the screen on the action column you have the options: "Go back to web service" (discarding all changes since the last save) and "Delete". If you click on the last one, a dialog will emerge and ask you if you like to remove the invoker. Click on the "Delete" button to confirm the removal of the invoker and its configuration or "Cancel" to close the delete dialog.

Change Invoker Invoker one of Web Service Test

» Web Services » Test » Change invoker Invoker one

Actions

[Go back to web service](#)

[Delete](#)

Invoker Details

* Name:

The name is typically used to call up an operation of a remote web service.

Description:

Invoker backend:

This OTRS invoker backend module will be called to prepare the data to be sent to the remote system, and to process its response data.

Mapping for outgoing request data:

The data from the invoker of OTRS will be processed by this mapping, to transform it to the kind of data the remote system expects.

Mapping for incoming response data:

The response data will be processed by this mapping, to transform it to the kind of data the invoker of OTRS expects.

Event Triggers:

EVENT	ASYNCHRONOUS	DELETE
HistoryAdd	Yes	Delete

This invoker will be triggered by the configured events.

Add Event Trigger: Asynchronous

To add a new event select the event object and event name and click on the '+' button.
Asynchronous event triggers are handled by the OTRS Scheduler in background (recommended).
Synchronous event triggers would be processed directly during the web request.

or

Figure: Web service invoker.

11.4.3.7.5. Веб Сервис

There are cases where you need to transform the data from one format to another (map or change data structure), because normally a web service is used to interact with a Remote System, that is highly probable that is not another OTRS system and / or could not understand the OTRS data structures and values. In these cases some or all values have to be changed, and sometimes even the names of the values (keys) or even the complete structure, in order to match with the expected data on the other end. To accomplish this task the Generic Interface Mapping Layer exists.

Each Remote System has its own data structures and it is possible to create new mapping modules for each case (e.g. there is a customized mapping module for SAP Solution Manager shipped with OTRS), but it is not always necessary. The module Mapping::Simple should cover most of the mapping needs.

Примечание

When Mapping::Simple does not cover all mapping needs for a web service, a new mapping module should be created. To learn more about how to create new mapping modules please consult the OTRS Development Manual.

Этот модуль дает возможность установить значение для отображения по умолчанию для каждого ключа или значения для всех передаваемых данных.

At the beginning of the screen you will see a general section where you can set the default rules that will apply for all the unmapped keys and values. There are three options available, these options are listed below:

- Не изменять: не касается ключей и значений в любом случае.

- Ignore (drop key/value pair): when this is applied to the key it deletes the key and value, because when a key is deleted then in consequence its associated value is deleted too. When this is applied to the value, only the value is deleted, keeping the key, that now will be associated to an empty value.
- MapTo (use provided key or value as default): all keys and / or values without a defined map rule, will use this as default, when you select this option a new text field will appear to set this default.

Clicking on the "+" button for new key map, will display a new box for a single mapping configuration. You can add as many key mappings as needed. Just click on the "+" button again and a new mapping box will appear below the existing one. From this mapping boxes you can define a map for a single key, with the next options:

- Точное значение(я): строка со старым значением будет заменена новой в случае полного совпадения "старого" ключа.
- Регулярное выражение: строки будут изменены с помощью правила преобразования.

Pressing the new value map "+" button will display a new row for a value map. Here it is also possible to define rules for each value to be mapped with the same options as for the key map (Exact value and Regular expression). You can add as many values to map as needed, and if you want to delete one of them, just click on the "-" button for each mapping value row.

Удаление целого раздела ключей отображения возможно, просто нажмите на кнопку "-", расположеную в верхнем правом углу каждого окна, для которого вы хотите удалить.

If you need to delete a complete mapping configuration: go back to the corresponding operation or invoker screen, look for the mapping direction that you select before and set its value to "-", and save the configuration to apply changes.

GenericInterface Mapping Simple for Web Service Test

» Web Services » Test » Operation Operation one » Simple Mapping for Incoming Data

Actions

[Go back to operation](#)

Mapping Simple

Default rule for unmapped keys:
This rule will apply for all keys with no mapping rule.

Default rule for unmapped values:
This rule will apply for all values with no mapping rule.

New key map:

▼ Mapping for Key KeyNew ⊖

Key mapping: Map key: matching the: *to new key:

Value mapping:

Map value: matching the: *to new value:

Map value: matching the: *to new value:

New value map:

or

Figure: Web service mapping.

11.5. Интерфейс командной строки Веб-Сервиса

Интерфейс командной строки (CLI) это быстрый путь для работы с веб-сервисами. Он состоит из набора инструментов, которые можно использовать для выполнения основных операций, таких как:

- Создать, Обновить, Читать, Просматривать и Удалить веб-сервисы основанные на YAML-файлах.
- Чтение журнала Отладчика с опциями фильтрации.

Примечание

Нет нужды в использовании интерфейса командной строки для работы с веб-сервисами. Интерфейс администратора содержит полный набор экранов/разделов для взаимодействия с каждой из составляющих веб-сервисов. Читайте раздел описания по работе этого интерфейса/GUI, включенный в это руководство.

11.5.1. Конфигурация Веб-сервиса

Файл "WebserviceConfig.pl" был разработан для того чтобы создать простой, но в то же время быстрый и мощный инструмент для работы с настройками веб-сервиса. Он предоставляет возможность выполнять следующие действия:

- Add: для создания веб-сервисов с использованием YAML-файлов в качестве источника конфигурации.

- **Update:** изменить существующий веб-сервис, конфигурация может быть изменена с использованием других или измененных YAML-файлов.
- **Read:** для вывода на экран текущих настроек веб-сервиса.
- **List:** чтобы получить полный список всех веб-сервисов, зарегистрированных в системе.
- **Delete:** для удаления веб-сервиса из системы. Будьте осторожны при выполнении этой операции, ее невозможно отменить.

Предупреждение

Операция веб-сервиса READ будет отображать на экране все настройки в виде простого текста, включая сохраненные пароли. Помните об этом и соблюдайте необходимые меры предосторожности!

Пример: Создание новой конфигурации веб-сервиса:

```
shell> OTRS_HOME/bin/otrs.WebserviceConfig.pl -a write -n <webservice_name>
-f /path/to/yaml/file
```

Также можно использовать файл 'otrs.WebserviceConfig.pl' со следующими опциями:

- **-a read -i <webservice_id>** - Для чтения сохраненной конфигурации.
- **-a write -n <webservice_name> -f /path/to/yaml/file** - Для создания нового веб-сервиса.
- **-a write -i <webservice_id> -f /path/to/yaml/file** - Для обновления веб-сервиса.
- **-a list** - Для просмотра списка доступных веб-сервисов.
- **-a delete -i <webservice_id>** - Для удаления веб-сервиса.

11.5.2. Отладчик веб-сервиса:

Еще одним доступным инструментом командной строки является скрипт "otrs.GenericInterfaceDebugRead.pl", который предоставляет интерфейс для поиска записей в журнале отладчика веб-сервиса.

Пример: Поиск записей в журнале отладчика:

```
shell> bin/otrs.GenericInterfaceDebugRead.pl
```

Дополнительные параметры могут быть использованы для сценария "otrs.GenericInterfaceDebugRead.pl":

- **-c** - для фильтрации по Communication ID (md5sum формат).
- **-t** - для фильтрации по CommunicationType ('Provider' или 'Requester').
- **-a** - для фильтрации по дате (На текущую дату или После нее).
- **-b** - для фильтрации по дате (на текущую дату или До нее).
- **-i** - для фильтрации по IP-адресу (должен быть правильный IPv4 или IPv6 адрес).
- **-w** - для фильтрации по ID веб-сервиса.
- **-d** - включить подробные передаваемые данные.

Пример: Поиск записей журнала отладчика со всеми параметрами:

```
shell> ./otrs.GenericInterfaceDebugRead.pl -c
a7cc4d9f5c70387a9bfbbe1351bc88966 -t Provider -a '2011-07-22 00:00:00' -b '2011-07-26
00:00:00' -i 127.0.0.1 -w 123 -d 1
```

Примечание

Настоятельно рекомендуем включать, как минимум один из параметров фильтрации, перечисленных выше, или даже больше, если выбран ключ "-d", т.к. *большое* количество информации может быть получено и отображаться на экране, что приведет в большому времени ожидания и получению избыточной информации.

11.6. Конфигурация Веб-сервиса

From its design the web services were conceived to be portable from one OTRS system to another, e.g. from a test or development environment to a production system. Therefore it was needed to have an easy way to extract the web service configuration from the database, and import it to another. To accomplish this task the Generic Interface uses YAML files as the web services configuration basis.

Why YAML? YAML is a markup language designed to be human friendly to read and write (it is easier to understand than JSON), it does not have some of the limitations of XML like numeric tags, it is open, standardized, and is complete enough to store the whole web service configuration.

Примечание

Чтобы узнать больше о YAML пожалуйста посетите ссылку <http://www.yaml.org/>.

Ниже приводится конфигурация веб-сервера на основе файла конфигурации в форматеYAML:

```
---
Debugger:
  DebugThreshold: debug
Description: This an example of a web service configuration
Provider:
  Operation:
    CloseIncident:
      Description: This is a test operation
      MappingInbound: {}
      MappingOutbound: {}
      RemoteSystemGuid: ''
      Type: Test::Test
    Test:
      Description: This is a test operation
      MappingInbound:
        Config:
          KeyMapDefault:
            MapTo: ''
            MapType: Keep
          KeyMapExact:
            Prio: Priority
          ValueMap:
            Priority:
              ValueMapExact:
                Critical: 5 Very High
                Information: 1 Very Low
                Warning: 3 Normal
```

```
ValueMapDefault:
  MapTo: 3 Normal
  MapType: MapTo
Type: Simple
MappingOutbound:
  Config:
    KeyMapDefault:
      MapTo: ''
      MapType: Ignore
    KeyMapExact:
      Priority: Prio
    ValueMap:
      Prio:
        ValueMapExact:
          1 Very Low: Information
          3 Normal: Warning
          5 Very High: Critical
    ValueMapDefault:
      MapTo: ''
      MapType: Ignore
    Type: Simple
  Type: Test::Test
Transport:
  Config:
    MaxLength: 10000000
    NameSpace: http://www.example.com/actions
    Type: HTTP::SOAP
RemoteSystem: remote.system.description.example.com
Requester:
  Invoker:
    Test:
      Description: This is a test invoker
      Events:
        - Asynchronous: 1
          Event: TicketCreate
        - Asynchronous: 0
          Event: ArticleUpdate
    MappingInbound:
      Type: Simple
    MappingOutbound:
      Type: Simple
    Type: Test::Test
Transport:
  Config:
    Authentication:
      Password: '*****'
      Type: BasicAuth
      User: otrs
    Encoding: utf-8
    Endpoint: http://www.example.com:8080/endpoint
    NameSpace: http://www.example.com/actions
    SOAPAction: Yes
    SOAPActionSeparator: '#'
  Type: HTTP::SOAP
```

11.6.1. Описание Конфигурации

11.6.1.1. Общий

- Описание: краткий текст, описывающий веб-сервис.
- RemoteSystem: краткое описание Удаленной Системы.
- Debugger: хранилище для хранения настроек отладчика.
- Provider: хранилище для хранения настроек поставщика.
- Requester: хранилище для хранения настроек о запрашивающей стороне.

11.6.1.2. Отладчик

- DebugThreshold: уровень отладчика

Возможные значения

- debug: все журналы хранятся в базе данных.
- info: информация, уведомления и журнал ошибок хранятся в базе данных.
- notice: ошибки уровня уведомление (notice) и ошибка (error) хранятся в базе данных.
- error: только ошибки уровня error хранятся в базе данных.

11.6.1.3. Поставщик

- Operation: хранилище для настроек каждой операции.
- Transport: хранилище для настроек поставщика транспортной сети.

11.6.1.3.1. Операция

- <OperationName>: Уникальное название операции, хранилище для своих собственных настроек (от 0..n, но без повторений).

11.6.1.3.1.1. <OperationName>

Этот раздел основан на операция с типом "Test::Test" другие операции могут содержать больше или другие настройки.

- Описание: краткий текст, описывающий операцию.
- MappingInbound: хранилище для настроек отображаемых данных для данных из входящего запроса.
- MappingOutbound: хранилище для отображающихся настроек для исходящих данных.
- Тип бэкэнд операция в формате Controller::Operation.

11.6.1.3.1.1.1. MappingInbound

Этот раздел основан на отображаемых значениях с типом "Simple". Другие отображаемые значения могут содержать больше или другие настройки.

- Config: хранилище настроек для этих отображаемых значений.
- Тип: mapping backend.

11.6.1.3.1.1.1.1. Config

- KeyMapDefault: хранилище для всех не отображаемых значений ключей.
- ValueMapDefault: хранилище для всех не отображаемых значений настроек.
- KeyMapExact: хранилище для всех ключей точных отображаемых значений (от 0 .. 1).
- KeyMapRegEx: хранилище для отображаемых значений всех регулярных выражений.
- ValueMap: хранилище для всех значений отображений (mapping-a).

11.6.1.3.1.1.1.1. KeyMapDefault

- MapTo: новое значение, которое будет использоваться (применима только если параметр MapType установлен в значение MapTo).
- MapType: правило для отображения.

Возможные значения

- Keep: оставить без изменений.
- Ignore: понизить.
- MapTo: заменить на значение MapTo.

11.6.1.3.1.1.1.2. ValueMapDefault

Тоже что и KeyMapDefault.

11.6.1.3.1.1.1.3. KeyMapExact

- <oldkey>: <newkey> (от 0 .. n но без повторений).

11.6.1.3.1.1.1.4. KeyMapRegEx

- <oldkey(RegEx)>: <newkey> (от 0 .. n но без повторений).

11.6.1.3.1.1.1.5. ValueMap

- <newkey>: контейнер для отображаемых значений для этого нового ключа (зависит от новых ключей от KeyMapExact и KeyMapRegEx).

11.6.1.3.1.1.1.5.1. <newkey>

- ValueMapExact: хранилище для всех точных отображаемых значений (от 0 .. 1).
- ValueMapRegEx: хранилище для отображаемых значений всех регулярных выражений (от 0 .. 1).

11.6.1.3.1.1.1.5.1.1. valueMapExact

- <oldvalue>: <newvalue> (cardinality 0 .. n без повторений).

11.6.1.3.1.1.1.5.1.2. ValueMapRegEx

- <oldvalue(RegEx)>: <newvalue> (от 0 .. n без повторений).

11.6.1.3.1.1.2. MappingOutbound

Тоже что и MappingInbound.

11.6.1.3.1.1.3. Передача

Этот раздел основана на сетевом поставщике HTTP::SOAP, другие поставщики могут содержать больше или другие настройки.

- Config: хранилище для конкретных параметров конфигурации транспортной сети.
- Type: the provider network transport backend.

11.6.1.3.1.1.3.1. Config

- MaxLength: максимальная длина в байтах для чтения SOAP-сообщений системой OTRS.

- Namespace: URI , который дает контекст для всех операций, принадлежащих этому веб-сервису.

11.6.1.4. Запрашиваемая сторона

- Invoker: хранилище для настроек каждой метки.
- Transport: хранилище для настроек транспортной сети запрашивающей стороны.

11.6.1.4.1. Запрашивающая сторона

- <InvokerName>: Уникальное имя для вызывающей стороны, хранилище для своих собственных настроек (от 0..n, без повторов).

11.6.1.4.1.1. <InvokerName>

Этот раздел основан на типе "Test::Test", другие invoker-ры могут иметь больше или другие настройки.

- Описание: короткий текст для описания вызывающей стороны
- События: хранилище настроек не именованного списка действий.
- MappingInbound: хранилище для настроек отображаемых значений для данных поступающих от ответов.
- MappingOutbound: хранилище для настроек отображаемых значений для исходящих данных запроса.
- Тип: invoker backend, в формате Controller::Invoker.

11.6.1.4.1.1.1. События

- *List Element*: (от 0 .. n)
- Asynchronous: установить если вызов вызывающей стороны будет возложен на Планировщика

Возможные значения

- 0: не обрабатывается Планировщиком (Scheduler).
- 1: Обрабатывается Планировщиком (Scheduler).
- Event: имя события триггера.

Возможные Значения (для событий заявки)

- TicketCreate
- TicketDelete
- TicketTitleUpdate
- TicketUnlockTimeoutUpdate
- TicketQueueUpdate
- TicketTypeUpdate
- TicketServiceUpdate
- TicketSLAUpdate

- TicketCustomerUpdate
- TicketFreeTextUpdate
- TicketFreeTimeUpdate
- TicketPendingTimeUpdate
- TicketLockUpdate
- TicketArchiveFlagUpdate
- TicketStateUpdate
- TicketOwnerUpdate
- TicketResponsibleUpdate
- TicketPriorityUpdate
- HistoryAdd
- HistoryDelete
- TicketAccountTime
- TicketMerge
- TicketSubscribe
- TicketUnsubscribe
- TicketFlagSet
- TicketFlagDelete
- TicketSlaveLinkAdd
- TicketSlaveLinkDelete
- TicketMasterLinkDelete

Возможные Значения (для действий над статьями)

- Действия над статьями
- ArticleCreate
- ArticleFreeTextUpdate
- ArticleUpdate
- ArticleSend
- ArticleBounce
- ArticleAgentNotification
- ArticleCustomerNotification
- ArticleAutoResponse

- ArticleFlagSet
- ArticleFlagDelete
- ArticleAgentNotification
- ArticleCustomerNotification

11.6.1.4.1.1.2. MappingInbound

То же что и Действие MappingInbound

11.6.1.4.1.1.3. MappingOutbound

Тоже что и Действие MappingInbound.

11.6.1.4.1.1.4. Передача

Этот раздел основан на транспортной сети запрашивающей стороны HTTP::SOAP, транспортные сети могут содержать больше или другие настройки.

- Config: хранилище для конкретных параметров конфигурации транспортной сети.
- Тип: бэкэнд транспортной сети запрашивающей стороны.

11.6.1.4.1.1.4.1. Config

- Аутентификация: хранилище для настроек аутентификации.
- Encoding: кодирование сообщения SOAP-запроса
- Endpoint: URI Уделнного Сервера веб-службы для приема запросов OTRS.
- NameSpace: URI который предоставят связь для всех invoker-ов, которые принадлежат этому веб-серверу.
- SOAPAction: для отправки пустого или заполненного SOAPAction-заголовка в формате SOAP сообщения (in "<NameSpace> <Separator> <Action>" format).

Возможные значения

- YES: для отправки заполненного заголовка SOAPAction.
- No: для отправки пустого обработчика SOAP-действия.
- SOAPActionSeparator: для установки <Separator> из заполненных SOAPAction заголовков.

Возможные значения

- '/': используется для .net веб-сервисов.
- '#': используется для всех веб-сервисов на базе протокола REST.

11.6.1.4.1.1.4.1.1. Authentication

- User: имя привилегированного пользователя, который имеет доступ к удаленному веб-сервису.
- Пароль: пароль для привелигированного пользователя в формате обычного текста.
- Тип: тип аутентификации.

11.7. Коннекторы (Connectors)

A Connector is in essence a set of actions that are either called Operations if OTRS acts as a web service provider or Invokers if OTRS acts as a web service requester. But it can also include special Mappings or Transports.

One Connector can either have only Operations, Only Invokers or both. A connector can even use parts of other connectors like the Mappings or Transports if they are not to specific for the Connector that is trying to implement them.

In other words a Connector is not limited to just the Controller layer but it can be extended to Data Mapping or Network Transport layers if needed.

Due to the modular design of the Generic Interface a Connector can be seen as a plug-in; this means that by adding Connectors the capabilities of the generic interface can be extended using: OTRS Feature add ons, OTRS Custom modules, 3rd Party modules, and so on.

11.7.1. Комплект Коннекторов (Connectors)

В комплекте с этой версией OTRS следующие коннекторы готовы к использованию.

- Session
- Заявк

11.7.1.1. Session Connector

Этот коннектор может создать действительный SessionID, который в дальнейшем может быть использован в любой другой операции.

Поставщики:

- Операции:
 - SessionCreate

11.7.1.1.1. Операции

11.7.1.1.1.1. SessionCreate

Creates a new new valid SessionID to be used in other operations from other connectors like TicketCreate.

Примечание

To use the SessionID in other operations from other connectors it is necessary that the operation implements authentication by SessionID. All the rest of the bundled operations are capable of accepting a valid SessionID as an authentication method.

Возможные Атрибуты:

```
<SessionCreate>
  <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <UserLogin?></UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?></CustomerUserLogin>
  <!--Optional:-->
  <Password?></Password>
</SessionCreate>
```

11.7.1.2. Коннектор Заявки

Этот коннектор использует базовую функциональность для взаимодействия с заявками

Поставщики:

- Операции:
 - TicketCreate
 - TicketUpdate
 - TicketGet
 - TicketSearch

11.7.1.2.1. Операции

11.7.1.2.1.1. TicketCreate

Предоставляет интерфейс для создания заявки в OTRS. Заявка должна содержать сообщение и может содержать несколько вложений, все заданные динамические поля, также могут получить значения при операции "Создать заявку/TicketCreate"

Возможные Атрибуты:

```

<TicketCreate>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?</UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?</CustomerUserLogin>
  <!--Optional:-->
  <SessionID?</SessionID>
  <!--Optional:-->
  <Password?</Password>
  <Ticket>
    <Title?</Title>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <QueueID?</QueueID>
    <!--Optional:-->
    <Queue?</Queue>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <TypeID?</TypeID>
    <!--Optional:-->
    <Type?</Type>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ServiceID?</ServiceID>
    <!--Optional:-->
    <Service?</Service>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <SLAID?</SLAID>
    <!--Optional:-->
    <SLA?</SLA>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <StateID?</StateID>
    <!--Optional:-->
    <State?</State>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
  
```

```

<!--Optional:-->
<PriorityID?></PriorityID>
<!--Optional:-->
<Priority?></Priority>
<!--You have a CHOICE of the next 2 items at this level-->
<!--Optional:-->
<OwnerID?></OwnerID>
<!--Optional:-->
<Owner?></Owner>
<!--You have a CHOICE of the next 2 items at this level-->
<!--Optional:-->
<ResponsibleID?></ResponsibleID>
<!--Optional:-->
<Responsible?></Responsible>
<CustomerUser?></CustomerUser>
<!--Optional:-->
<CustomerID?></CustomerID>
<!--Optional:-->
<PendingTime>
  <Year?></Year>
  <Month?></Month>
  <Day?></Day>
  <Hour?></Hour>
  <Minute?></Minute>
</PendingTime>
</Ticket>
<Article>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ArticleTypeID?></ArticleTypeID>
  <!--Optional:-->
  <ArticleType?></ArticleType>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <SenderTypeID?></SenderTypeID>
  <!--Optional:-->
  <SenderType?></SenderType>
  <!--Optional:-->
  <From?></From>
  <Subject?></Subject>
  <Body?></Body>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ContentType?></ContentType>
  <Charset?></Charset>
  <MimeType?></MimeType>
  <!--Optional:-->
  <HistoryType?></HistoryType>
  <!--Optional:-->
  <HistoryComment?></HistoryComment>
  <!--Optional:-->
  <AutoResponseType?></AutoResponseType>
  <!--Optional:-->
  <TimeUnit?></TimeUnit>
  <!--Optional:-->
  <NoAgentNotify?></NoAgentNotify>
  <!--Zero or more repetitions:-->
  <ForceNotificationToUserID?></ForceNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeNotificationToUserID?></ExcludeNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeMuteNotificationToUserID?></ExcludeMuteNotificationToUserID>
</Article>
<!--Zero or more repetitions:-->
<DynamicField>
  <Name?></Name>
  <!--1 or more repetitions:-->
  <Value?></Value>
</DynamicField>
<!--Zero or more repetitions:-->
<Attachment>
  <Content>cid:61886944659</Content>

```

```

    <ContentType?></ContentType>
    <Filename?></Filename>
  </Attachment>
</TicketCreate>

```

11.7.1.2.1.2. TicketUpdate

Операция "Изменение заявки/TicketUpdate" добавляет возможность изменения атрибутов существующей заявки или добавить новое сообщение/заметку, включая вложения и все определенные динамические поля для заявки и нового сообщения.

Примечание

Нет необходимости создавать новое сообщение/заметку для изменения атрибутов заявки.

Возможные Атрибуты:

```

<TicketUpdate>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?></UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?></CustomerUserLogin>
  <!--Optional:-->
  <SessionID?></SessionID>
  <!--Optional:-->
  <Password?></Password>
  <!--You have a CHOICE of the next 2 items at this level-->
  <TicketID?></TicketID>
  <TicketNumber?></TicketNumber>
  <!--Optional:-->
  <Ticket>
    <!--Optional:-->
    <Title?></Title>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <QueueID?></QueueID>
    <!--Optional:-->
    <Queue?></Queue>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <TypeID?></TypeID>
    <!--Optional:-->
    <Type?></Type>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ServiceID?></ServiceID>
    <!--Optional:-->
    <Service?></Service>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <SLAID?></SLAID>
    <!--Optional:-->
    <SLA?></SLA>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <StateID?></StateID>
    <!--Optional:-->
    <State?></State>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <PriorityID?></PriorityID>
    <!--Optional:-->
    <Priority?></Priority>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->

```



```

<OwnerID?></OwnerID>
<!--Optional:-->
<Owner?></Owner>
<!--You have a CHOICE of the next 2 items at this level-->
<!--Optional:-->
<ResponsibleID?></ResponsibleID>
<!--Optional:-->
<Responsible?></Responsible>
<!--Optional:-->
<CustomerUser?></CustomerUser>
<!--Optional:-->
<CustomerID?></CustomerID>
<!--Optional:-->
<PendingTime>
  <Year?></Year>
  <Month?></Month>
  <Day?></Day>
  <Hour?></Hour>
  <Minute?></Minute>
</PendingTime>
</Ticket>
<!--Optional:-->
<Article>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ArticleTypeID?></ArticleTypeID>
  <!--Optional:-->
  <ArticleType?></ArticleType>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <SenderTypeID?></SenderTypeID>
  <!--Optional:-->
  <SenderType?></SenderType>
  <!--Optional:-->
  <From?></From>
  <Subject?></Subject>
  <Body?></Body>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ContentType?></ContentType>
  <Charset?></Charset>
  <MimeType?></MimeType>
  <!--Optional:-->
  <HistoryType?></HistoryType>
  <!--Optional:-->
  <HistoryComment?></HistoryComment>
  <!--Optional:-->
  <AutoResponseType?></AutoResponseType>
  <!--Optional:-->
  <TimeUnit?></TimeUnit>
  <!--Optional:-->
  <NoAgentNotify?></NoAgentNotify>
  <!--Zero or more repetitions:-->
  <ForceNotificationToUserID?></ForceNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeNotificationToUserID?></ExcludeNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeMuteNotificationToUserID?></ExcludeMuteNotificationToUserID>
</Article>
<!--Zero or more repetitions:-->
<DynamicField>
  <Name?></Name>
  <!--1 or more repetitions:-->
  <Value?></Value>
</DynamicField>
<!--Zero or more repetitions:-->
<Attachment>
  <Content>cid:166861569966</Content>
  <ContentType?></ContentType>
  <Filename?></Filename>
</Attachment>
</TicketUpdate>

```

11.7.1.2.1.3. TicketGet

Эта операция используется для получения всех атрибутов заявки, включая динамические поля, все сообщения и все вложения принадлежащие заявке.

Возможные Атрибуты:

```
<TicketGet>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?></UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?></CustomerUserLogin>
  <!--Optional:-->
  <SessionID?></SessionID>
  <!--Optional:-->
  <Password?></Password>
  <!--1 or more repetitions:-->
  <TicketID?></TicketID>
  <!--Optional:-->
  <DynamicFields?></DynamicFields>
  <!--Optional:-->
  <Extended?></Extended>
  <!--Optional:-->
  <AllArticles?></AllArticles>
  <!--Optional:-->
  <ArticleSenderType?></ArticleSenderType>
  <!--Optional:-->
  <ArticleOrder?></ArticleOrder>
  <!--Optional:-->
  <ArticleLimit?></ArticleLimit>
  <!--Optional:-->
  <Attachments?></Attachments>
</TicketGet>
```

11.7.1.2.1.4. TicketSearch

Операция "ПоискЗаявки/TicketSearch" возвращает список всех Ticket IDs/Идентификаторов заявок, удовлетворяющих заданному критерию.

Возможные Атрибуты:

```
<TicketSearch>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?></UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?></CustomerUserLogin>
  <!--Optional:-->
  <SessionID?></SessionID>
  <!--Optional:-->
  <Password?></Password>
  <!--Optional:-->
  <Limit?></Limit>
  <!--Zero or more repetitions:-->
  <TicketNumber?></TicketNumber>
  <!--Zero or more repetitions:-->
  <Title?></Title>
  <!--Zero or more repetitions:-->
  <Queues?></Queues>
  <!--Zero or more repetitions:-->
  <QueueIDs?></QueueIDs>
```

```

<!--Optional:-->
<UseSubQueues>?</UseSubQueues>
<!--Zero or more repetitions:-->
<Types>?</Types>
<!--Zero or more repetitions:-->
<TypeIDs>?</TypeIDs>
<!--Zero or more repetitions:-->
<States>?</States>
<!--Zero or more repetitions:-->
<StateIDs>?</StateIDs>
<!--Zero or more repetitions:-->
<StateType>?</StateType>
<!--Zero or more repetitions:-->
<StateTypeIDs>?</StateTypeIDs>
<!--Zero or more repetitions:-->
<Priorities>?</Priorities>
<!--Zero or more repetitions:-->
<PriorityIDs>?</PriorityIDs>
<!--Zero or more repetitions:-->
<Services>?</Services>
<!--Zero or more repetitions:-->
<ServiceIDs>?</ServiceIDs>
<!--Zero or more repetitions:-->
<SLAs>?</SLAs>
<!--Zero or more repetitions:-->
<SLAIDs>?</SLAIDs>
<!--Zero or more repetitions:-->
<Locks>?</Locks>
<!--Zero or more repetitions:-->
<LockIDs>?</LockIDs>
<!--Zero or more repetitions:-->
<OwnerIDs>?</OwnerIDs>
<!--Zero or more repetitions:-->
<ResponsibleIDs>?</ResponsibleIDs>
<!--Zero or more repetitions:-->
<WatchUserIDs>?</WatchUserIDs>
<!--Zero or more repetitions:-->
<CustomerID>?</CustomerID>
<!--Zero or more repetitions:-->
<CustomerUserLogin>?</CustomerUserLogin>
<!--Zero or more repetitions:-->
<CreatedUserIDs>?</CreatedUserIDs>
<!--Zero or more repetitions:-->
<CreatedTypes>?</CreatedTypes>
<!--Zero or more repetitions:-->
<CreatedTypeIDs>?</CreatedTypeIDs>
<!--Zero or more repetitions:-->
<CreatedPriorities>?</CreatedPriorities>
<!--Zero or more repetitions:-->
<CreatedPriorityIDs>?</CreatedPriorityIDs>
<!--Zero or more repetitions:-->
<CreatedStates>?</CreatedStates>
<!--Zero or more repetitions:-->
<CreatedStateIDs>?</CreatedStateIDs>
<!--Zero or more repetitions:-->
<CreatedQueues>?</CreatedQueues>
<!--Zero or more repetitions:-->
<CreatedQueueIDs>?</CreatedQueueIDs>
<!--Zero or more repetitions:-->
<DynamicFields>
  <!--You have a MANDATORY CHOICE of the next 6 items at this level-->
  <!--Optional:-->
  <Equals>?</Equals>
  <!--Optional:-->
  <Like>?</Like>
  <!--Optional:-->
  <GreaterThan>?</GreaterThan>
  <!--Optional:-->
  <GreaterThanEquals>?</GreaterThanEquals>
  <!--Optional:-->
  <SmallerThan>?</SmallerThan>
  <!--Optional:-->

```

```

    <SmallerThanEquals>?</SmallerThanEquals>
  </DynamicFields>
  <!--Optional:-->
  <Ticketflag>
    <!--Optional:-->
    <Seen>?</Seen>
  </Ticketflag>
  <!--Optional:-->
  <From>?</From>
  <!--Optional:-->
  <To>?</To>
  <!--Optional:-->
  <Cc>?</Cc>
  <!--Optional:-->
  <Subject>?</Subject>
  <!--Optional:-->
  <Body>?</Body>
  <!--Optional:-->
  <FullTextIndex>?</FullTextIndex>
  <!--Optional:-->
  <ContentSearch>?</ContentSearch>
  <!--Optional:-->
  <ConditionInline>?</ConditionInline>
  <!--Optional:-->
  <ArticleCreateTimeOlderMinutes>?</ArticleCreateTimeOlderMinutes>
  <!--Optional:-->
  <ArticleCreateTimeNewerMinutes>?</ArticleCreateTimeNewerMinutes>
  <!--Optional:-->
  <ArticleCreateTimeNewerDate>?</ArticleCreateTimeNewerDate>
  <!--Optional:-->
  <ArticleCreateTimeOlderDate>?</ArticleCreateTimeOlderDate>
  <!--Optional:-->
  <TicketCreateTimeOlderMinutes>?</TicketCreateTimeOlderMinutes>
  <!--Optional:-->
  <ATicketCreateTimeNewerMinutes>?</ATicketCreateTimeNewerMinutes>
  <!--Optional:-->
  <TicketCreateTimeNewerDate>?</TicketCreateTimeNewerDate>
  <!--Optional:-->
  <TicketCreateTimeOlderDate>?</TicketCreateTimeOlderDate>
  <!--Optional:-->
  <TicketChangeTimeOlderMinutes>?</TicketChangeTimeOlderMinutes>
  <!--Optional:-->
  <TicketChangeTimeNewerMinutes>?</TicketChangeTimeNewerMinutes>
  <!--Optional:-->
  <TicketChangeTimeNewerDate>?</TicketChangeTimeNewerDate>
  <!--Optional:-->
  <TicketChangeTimeOlderDate>?</TicketChangeTimeOlderDate>
  <!--Optional:-->
  <TicketCloseTimeOlderMinutes>?</TicketCloseTimeOlderMinutes>
  <!--Optional:-->
  <TicketCloseTimeNewerMinutes>?</TicketCloseTimeNewerMinutes>
  <!--Optional:-->
  <TicketCloseTimeNewerDate>?</TicketCloseTimeNewerDate>
  <!--Optional:-->
  <TicketCloseTimeOlderDate>?</TicketCloseTimeOlderDate>
  <!--Optional:-->
  <TicketPendingTimeOlderMinutes>?</TicketPendingTimeOlderMinutes>
  <!--Optional:-->
  <TicketPendingTimeNewerMinutes>?</TicketPendingTimeNewerMinutes>
  <!--Optional:-->
  <TicketPendingTimeNewerDate>?</TicketPendingTimeNewerDate>
  <!--Optional:-->
  <TicketPendingTimeOlderDate>?</TicketPendingTimeOlderDate>
  <!--Optional:-->
  <TicketEscalationTimeOlderMinutes>?</TicketEscalationTimeOlderMinutes>
  <!--Optional:-->
  <TTicketEscalationTimeNewerMinutes>?</TTicketEscalationTimeNewerMinutes>
  <!--Optional:-->
  <TicketEscalationTimeNewerDate>?</TicketEscalationTimeNewerDate>
  <!--Optional:-->
  <TicketEscalationTimeOlderDate>?</TicketEscalationTimeOlderDate>
  <!--Optional:-->

```

```

<ArchiveFlags>?</ArchiveFlags>
<!--Zero or more repetitions:-->
<OrderBy>?</OrderBy>
<!--Zero or more repetitions:-->
<SortBy>?</SortBy>
<!--Zero or more repetitions:-->
<CustomerUserID>?</CustomerUserID>
</TicketSearch>

```

11.7.2. Примеры:

11.7.2.1. Конфигурация Веб-сервиса

The following is a basic but complete web service configuration file in YAML format to use all the Ticket Connector operations. In order to use it in OTRS you need to copy the content, save it into a file called GenericTicketConnector.yml, and import it into OTRS in the Web Services screen in the Admin panel by clicking in the "Add web service" action from the overview screen and then clicking in the "Import web service" action in the add screen.

```

---
Debugger:
  DebugThreshold: debug
  TestMode: 0
Description: Ticket Connector Sample
FrameworkVersion: 3.2.x CVS
Provider:
  Operation:
    SessionCreate:
      Description: Creates a Session
      MappingInbound: {}
      MappingOutbound: {}
      Type: Session::SessionCreate
    TicketCreate:
      Description: Creates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketCreate
    TicketUpdate:
      Description: Updates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketUpdate
    TicketGet:
      Description: Retrieve Ticket data
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketGet
    TicketSearch:
      Description: Search for Tickets
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketSearch
  Transport:
    Config:
      MaxLength: 100000000
      NameSpace: http://www.otrs.org/TicketConnector/
      Type: HTTP::SOAP
RemoteSystem: ''
Requester:
  Transport:
    Type: ''

```

11.7.2.2. Запрашивающая сторона Perl SOAP

The following code is a Perl script that can connect to OTRS via the generic interface. In order to perform the operations provided by the Ticket Connector, it uses two Perl CPAN modules SOAP::Lite and Data::Dumper. Please make sure that your environment is capable to use these modules before you try to run the script.

```
#!/usr/bin/perl -w
# --
# otrs.SOAPRequest.pl - sample to send a SOAP request to OTRS Generic Interface Ticket
Connector
# Copyright (C) 2001-2015 OTRS AG, http://otrs.com/
# --
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU AFFERO General Public License as published by
# the Free Software Foundation; either version 3 of the License, or
# any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU Affero General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
# or see http://www.gnu.org/licenses/agpl.txt.
# --

use strict;
use warnings;

# use ../ as lib location
use File::Basename;
use FindBin qw($RealBin);
use lib dirname($RealBin);

use SOAP::Lite;
use Data::Dumper;

# ---
# Variables to be defined.

# this is the URL for the web service
# the format is
# <HTTP_TYPE>:://<OTRS_FQDN>/nph-genericinterface.pl/Webservice/<WEB_SERVICE_NAME>
# or
# <HTTP_TYPE>:://<OTRS_FQDN>/nph-genericinterface.pl/WebserviceID/<WEB_SERVICE_ID>
my $URL = 'http://localhost/otrs/nph-genericinterface.pl/Webservice/GenericTicketConnector';

# this name space should match the specified name space in the SOAP transport for the web
service.
my $NameSpace = 'http://www.otrs.org/TicketConnector/';

# this is operation to execute, it could be TicketCreate, TicketUpdate, TicketGet,
TicketSearch
# or SessionCreate. and they must to be defined in the web service.
my $Operation = 'TicketCreate';

# this variable is used to store all the parameters to be included on a request in XML
format. Each
# operation has a determined set of mandatory and non mandatory parameters to work
correctly. Please
# check the OTRS Admin Manual in order to get a complete list of parameters.
my $XMLData = '
<UserLogin>some user login</UserLogin>
<Password>some password</Password>
<Ticket>
  <Title>some title</Title>
```

```

    <CustomerUser>some customer user login</CustomerUser>
    <Queue>some queue</Queue>
    <State>some state</State>
    <Priority>some priority</Priority>
  </Ticket>
</Article>
  <Subject>some subject</Subject>
  <Body>some body</Body>
  <ContentType>text/plain; charset=utf8</ContentType>
</Article>
';

# ---

# create a SOAP::Lite data structure from the provided XML data structure.
my $SOAPData = SOAP::Data
  ->type( 'xml' => $XMLData );

my $SOAPObject = SOAP::Lite
  ->uri($NameSpace)
  ->proxy($URL)
  ->$Operation($SOAPData);

# check for a fault in the soap code.
if ( $SOAPObject->fault ) {
  print $SOAPObject->faultcode, " ", $SOAPObject->faultstring, "\n";
}

# otherwise print the results.
else {

  # get the XML response part from the SOAP message.
  my $XMLResponse = $SOAPObject->context()->transport()->proxy()->http_response()-
  >content();

  # deserialize response (convert it into a perl structure).
  my $Deserialized = eval {
    SOAP::Deserializer->deserialize($XMLResponse);
  };

  # remove all the headers and other not needed parts of the SOAP message.
  my $Body = $Deserialized->body();

  # just output relevant data and no the operation name key (like TicketCreateResponse).
  for my $ResponseKey ( keys %{$Body} ) {
    print Dumper( $Body->{$ResponseKey} );
  }
}

```

12. Планировщик OTRS

Планировщик OTRS/Scheduler это независимый системный процесс, который выполняет задания в фоновом режиме. Этот тип процессов известен как *daemons* в Unix / Linux системах или как *services* в Windows средах. Он независим, но это не означает, что Планировщик может существовать отдельно, он полностью интегрирован в OTRS и может использовать модули OTRS необходимые для его работы.

По соображениям здравого смысла, Планировщик должен запускаться время от времени. Это выполняется автоматически раз в день, но может быть настроено при необходимости в SysConfig редактированием параметра "Scheduler::RestartAfterSeconds".

Планировщик OTRS является полностью автоматическим процессом, и взаимодействие с человеком состоит только в том, чтобы периодически проверять его состояние, запускать или останавливать по мере необходимости.

Примечание

Если Планировщик остановлен по какой либо причине, все отложенные задания зарегистрированные во время его простоя будут выполнены как только он будет запущен вновь (если только их запуск не предусмотрен позднее в будущем).

12.1. Графический интерфейс планировщика

Планировщик не виден в Графическом Пользовательском Интерфейсе OTRS до того времени пока он остается работать.

12.1.1. Планировщик Не Запускает Уведомления

Есть два типа уведомлений, если система обнаруживает, что Планировщик не запущен. Они базируются на частоте обновления состояния процесса Планировщика. Если разница между текущим временем и временем последнего обновления состояния в 2 раза больше частоты обновления процесса, предупреждающее сообщение будет отображаться в OTRS области уведомлений. Если она больше в 4 раза, то, вместо предупреждения, будет отображаться тревожное сообщение.

Параметр Планировщика обновление времени может быть настроен с помощью параметра "Scheduler::PIDUpdateTime" в файле SysConfig.

Если вы видите предупреждающее сообщение, необязательно сразу предпринимать какие-то действия, но настоятельно рекомендуется проверить, что процесс запущен. Если отображентся тревожное сообщение, велика вероятность, что Планировщик действительно не работает и его следует запустить.

По умолчанию Планировщик не запускает уведомлений, если есть действительный веб-сервис, зарегистрированный в базе данных и он отображается только для пользователей из группы "admin".

Чтобы отключить уведомления (не рекомендуется) или же изменить или добавить группы уведомлений, пожалуйста, отредактируйте настройку "Frontend::NotifyModule###800-Scheduler-Check" в файле SysConfig.



Figure: Уведомления планировщика.

12.1.2. Запуск Планировщика

При нажатии на уведомление о том, что Планировщик не запущен (предупреждающее или тревожное) откроется диалоговое окно, позволяющее стартовать Планировщик заново. Планировщик может быть запущен в нормальном или форсированном режиме, активированием соответствующего поля выбора в этом окне.

Примечание

Форсированный старт Планировщика необходим только в случае ненормального завершения предыдущего процесса Планировщика и его Process ID уже занесен в базу данных.

Используйте инструменты командной строки, описанные ниже, чтобы иметь полный контроль над процессами Планировщика и иметь возможность проверить его реальное состояние.

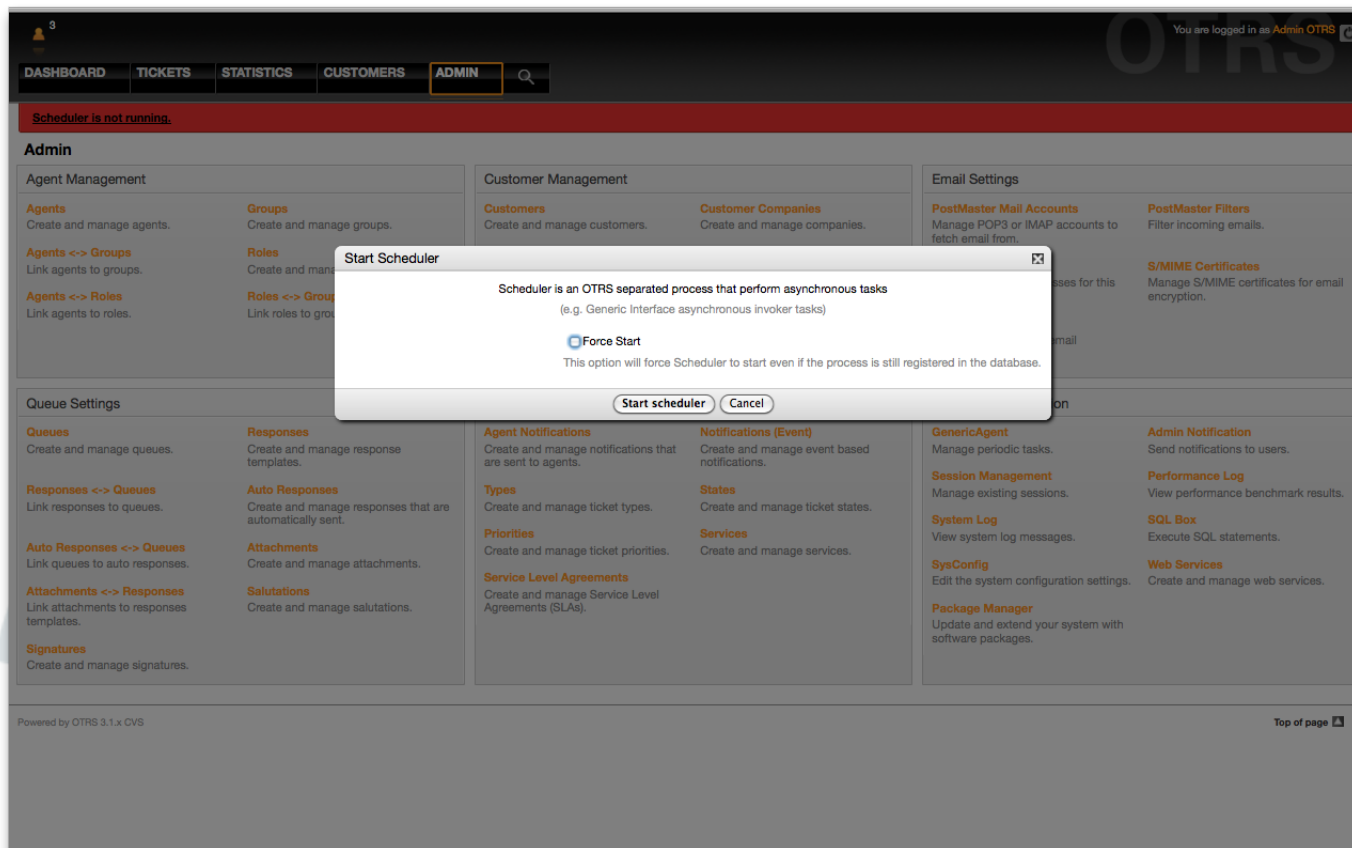


Рисунок: Запуск Планировщика.

12.2. Интерфейс командной строки Планировщика

Командная строка Планировщика позволяет управлять процессами (Запуск / Остановка) или запрашивать их состояние. Также есть инструменты для регистрации процесса, который будет управлять операционной системой.

В комплекте с OTRS есть два набора инструментов командной строки, один для операционной системы Unix / Linux, а другой для MS Windows.

12.2.1. Unix / Linux

12.2.1.1. Файлы Init.d Планировщика

Init.d файлы это специальные скрипты, которые вызываются операционной системой при старте и завершении работы.

OTRS предлагает init.d скрипты для автоматического процесса Планировщика старта/завершения OTRS операционной системой. Эти скрипты расположены в OTRS_HOME/scripts.

Скрипты init.d должны быть скопированы в соответствующее место вашей операционной системы. Им должны быть даны соответствующие права и установлены

значения некоторых внутренних переменных для правильной работы. Если вы использовали OTRS RPMs для установки системы, это будет сделано автоматически.

Init.d Скрипт Внутренних Переменных

- **OTRS_HOME** - путь установки OTRS.
- **User** - имя пользователя процесса apache.
- **Group** - apache обрабатывает группу пользователей с таким именем.

Примечание

В настоящее время OTRS предоставляет сценарии init.d только для Linux-платформ.

Таблица 4.7. Список init сценариев и поддерживаемых Операционных систем

Сценарий инициализации	Поддерживаемые операционные системы
otrs-scheduler-linux	Red Hat, Fedora, CentOS, SUSE, openSUSE, Debian, Ubuntu
otrs-scheduler-gentoo-init.d, scheduler-gentoo-conf.d	otrs- Gentoo

Пример 4.29. Пример для запуска OTRS-Планировщика в linux

```
shell> /etc/init.d/otrs-scheduler-linux start
```

Возможные действия

- **start** запуск процесса OTRS-Планировщика.
- **stop** остановка процесса OTRS-Планировщика.
- **restart** перезапуск процесса OTRS-Планировщика.
- **status** запросить статус процесса OTRS-Планировщика.

Для корректной работы Планировщика необходима база данных, поскольку он регистрирует свой ID процесса, именно по этой причине необходимо:

- Выполните скрипт init.d для *запуска* процесса Планировщика, после того как запущен процесс базы данных.
- Выполните скрипт init.d для *остановки* процесса Планировщика, до того как процесс базы данных остановлен.

Примечание

Если вы желаете, чтобы Планировщик стартовал при загрузке системы, обратитесь к документации на операционную систему. Там вы найдете куда поместить скрипты init.d и как их настроить для автоматического старта, а также указать порядок их выполнения.

12.2.1.2. Файл "Демона" Планировщика

Это часть Планировщика, который остается работать в фоновом режиме, проверяя задания, которые нужно выполнить. Она также обеспечивает основные функции для управления процессом.

Все Unix / Linux системы используют файл **OTRS_HOME/bin/otrs.Scheduler.pl**.

Пример 4.30. Пример для Запуска OTRS-Планировщика

```
shell> OTRS_HOME/bin/otrs.Scheduler.pl -a start
```

Доступные Опции

- **-a** действие.

Возможные значения

- **start**- старт процесса Планировщика.
- **stop**- остановка процесса Планировщика.
- **status**- для запроса статуса процесса Планировщика.
- **-f** принудительный старт или остановка процесса Планировщика.

Пример 4.31. Пример принудительной остановки Планировщика OTRS

```
shell> OTRS_HOME/bin/otrs.Scheduler.pl -a stop -f 1
```

Примечание

Принудительная остановка Планировщика используется для удаления ID процесса из базы данных когда планировщик не работает и процесс остается зарегистрированным.

Принудительный запуск Планировщика используется для старта его процесса, когда планировщик не работает и процесс зарегистрирован.

Force start or stop are only necessary if the start of the process is needed to be done before the process update time expires. Otherwise an expired entry in the database is discarded by normal start.

12.2.2. Windows

12.2.2.1. Установщик Планировщика

В Windows системах, Планировщик OTRS регистрируется как Сервис через Windows Service Control Manager (SCM). Чтобы этот процесс управлялся с помощью SCM необходимо зарегистрировать этот сервис. Если вы используете Windows Installer, он это сделает сам и никаких дополнительных ручных операций не требуется.

OTRS предоставляет сценарий **OTRS_HOME/bin/otrs.Scheduler4WinInstaller.pl** для регистрации и отмены регистрации OTRS-планировщика в SCM.

Примечание

Регистрация и разрегистрация может быть выполнена только членами группы Администратор в Windows. Чтобы правильно зарегистрировать сервис, сначала кликните правой кнопкой мыши на cmd.exe и выберите "Запуск от имени Администратора"

Пример 4.32. Пример регистрации Планировщика OTRS как сервиса Windows

```
install shell> perl OTRS_HOME/bin/otrs.Scheduler4WinInstaller.pl -a
```

Доступные Опции

- **-a** действие.

Возможные значения

- **install**- для установки процесса Планировщика в Windows SCM.
- **remove**- для удаления процесса Планировщика из Windows SCM.

После регистрации в Windows SCM процесс Планировщика OTRS может использоваться как любой другой сервис Windows. Его можно стартовать, останавливать, перезапускать, а также настроить его запуск вручную или автоматически.

Примечание

Обратитесь к документации Windows для более подробного изучения Сервисов Windows и Windows SCM.

12.2.2.2. Сервисный Файл Планировщика

Это часть Планировщика, который остается работать в фоновом режиме, проверяя задания, которые нужно выполнить. Она также обеспечивает основные функции для управления процессом.

Операционная система Windows использует файл **OTRS_HOME/bin/otrs.Scheduler4Win.pl**.

Пример 4.33. Пример запуска Планировщика OTRS в Windows

```
shell> OTRS_HOME/bin/otrs.Scheduler4Win.pl -a start
```

Доступные Опции

- **-a** действие.

Возможные значения

- **start**- старт процесса Планировщика.
- **stop**- остановка процесса Планировщика.
- **status**- для запроса статуса процесса Планировщика.

- **-f** принудительный старт или остановка процесса Планировщика.

Пример 4.34. Пример принудительной остановки Планировщика OTRS в Windows

```
shell> OTRS_HOME/bin/otrs.Scheduler4Win.pl -a stop -f 1
```

Примечание

Принудительная остановка Планировщика используется для удаления ID процесса из базы данных когда планировщик не работает и процесс остается зарегистрированным.

Принудительный старт Планировщика используется для запуска процесса Планировщика, если Планировщик не работает и его Process ID уже занесен в базу данных.

Force start or stop are only necessary if starting the process is needed to be done before the process update time expires. Otherwise an expired entry in the database would be discarded by a normal start.

Глава 5. Настройка

1. Списки Контроля Доступа (ACLs)

1.1. Введение

Начиная с OTRS 2.0 и выше, Списки Контроля Доступа (Access Control Lists - ACLs) могут быть использованы для управления доступом к заявкам, модулям, очередям и т.д или для влияния действия по заявке (закрытие, перемещение и т.д) в определенных ситуациях. Списки прав доступа (ACLs) могут быть использованы в дополнение к существующей системе прав доступа на основе ролей и групп. Можно использовать ACLs, для настройки элементарных потоков работ/workflows на основе атрибутов заявки.

Списки прав доступа ACL не могут быть созданы используя Интерфейс SysConfig, а должны быть напрямую введены в файл Kernel/Config.pm. В этой главе приведены примеры в которых можно пройти процесс создания ACL и описание всех доступных параметров ACL. ПРИМЕЧАНИЕ: В настоящее время в текущей версии OTRS процесс создания ACL изменен. Доступен редактор ACL автоматизирующий процесс их создания и синхронизации SysConfig interface.

1.2. Примеры

Пример 5.1. Списки прав доступа (ACL) позволяют перемещать в очереди только заявки с приоритетом 5.

Этот пример показывает основную структуру ACL. Для начала, она должна иметь название. В нашем случае это "ACL-Name-2". Обратите внимание, что перед запуском ACLs будут отсортированы по номерам, поэтому имена нужно использовать тщательно.

Во-вторых, существует секция "Properties", которая фильтрует заявки. Все условия/критерии, заданные здесь, будут проверяться для определения, должен ли ACL быть применен к ним, или нет. В нашем примере, проверяется, находится ли заявка в очереди "Raw" и имеет ли приоритет "5 very high". Это также применимо к форме изменения заявки (например, если заявка в очереди "raw" и имеет приоритет "3 normal", но, затем, приоритет изменяется на "5 very high" выбором из выпадающего списка, то условие также будет верно).

Наконец, раздел "Возможные" определяет изменения экрана. В нашем случае из имеющихся очередей только очередь "Alert" может быть выбрана на странице Заявки.

```
# ticket acl
$self->{TicketAcl}->{'100-Example-ACL'} = {
  # match properties
  Properties => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
      Priority => ['5 very high'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
```

```

    Queue => ['Alert'],
  },
};

```

Пример 5.2. ACL позволяют перемещать в другие очереди только заявки с приоритетом 5, уже имеющиеся в базе данных.

Этот пример очень похож на предыдущий, но в этом случае только заявки в очереди "Raw" и с приоритетом "5 very high", уже имеющиеся в базе данных будут соответствовать критерию. Такой тип ACL не учитывает изменения в формах изменения заявки, до того как заявка будет действительно обновлена в базе данных.

```

# ticket acl
$Self->{TicketAcl}->{'100-Example-ACL'} = {
  # match properties
  PropertiesDatabase => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
      Priority => ['5 very high'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
      Queue => ['Alert'],
    },
  },
};

```

Пример 5.3. Списки прав доступа (ACL) делают невозможным закрытие заявок в очереди raw и скрывают кнопку "закрыть".

Здесь вы можете увидеть как поле заявки (состояние), может проверяться на более чем возможное для ввода значение. Можно также ограничить возможные действия, выполняемые над конкретной заявкой. В этом случае, заявка не может быть закрыта.

```

$Self->{TicketAcl}->{'101-Second-Example-ACL'} = {
  # match properties
  Properties => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
      State => ['new', 'open', 'pending reminder'],
    },
    # possible action options
    Action => {
      AgentTicketBounce => 1,
      AgentTicketClose => 0,
      AgentTicketCompose => 1,
      AgentTicketCustomer => 1,
      AgentTicketForward => 1,
      AgentTicketFreeText => 1,
      AgentTicketHistory => 1,
      AgentTicketLink => 1,
      AgentTicketLock => 1,
      AgentTicketMerge => 1,
    }
  }
};

```

```

    AgentTicketMove           => 1,
    AgentTicketNote           => 1,
    AgentTicketOwner          => 1,
    AgentTicketPending        => 1,
    AgentTicketPhone          => 1, # only used to hide the Split action
    AgentTicketPhoneInbound   => 1,
    AgentTicketPhoneOutbound  => 1,
    AgentTicketPrint          => 1,
    AgentTicketPriority        => 1,
    AgentTicketResponsible    => 1,
    AgentTicketWatcher        => 1,
    AgentTicketZoom           => 1,
    AgentLinkObject           => 1, # only used to hide the Link action
  },
},
};

```

Пример 5.4. ACL удаляет навсегда возможность установить состояние "closed successful".

Этот пример показывает как можно использовать "негативные" фильтры (состояние "closed successful" будет удалено из списков выбора). Вы также можете видеть, что не заданные условия для "properties" для заявки будет означать, что оно верно для всех заявок, т.е. ACL применяется всегда. Это может быть полезно, если вы желаете скрыть определенные значения по умолчанию и допустить ввод других значений только в заданных обстоятельствах (например, если агент принадлежит определенной группе).

```

$self->{TicketAcl}->{'102-Third-ACL-Example'} = {
  # match properties
  Properties => {
    # current ticket match properties (match always)
  },
  # return possible options
  PossibleNot => {
    # possible ticket options
    Ticket => {
      State => ['closed successful'],
    },
  },
};

```

Пример 5.5. ACL только отображает Hardware-сервисы для заявок, которые были созданы в очередях, начинающихся с "HW".

Этот пример также показывает как можно использовать регулярные выражения для сопоставления заявок и фильтрации доступных опций.

```

$self->{TicketAcl}->{'Only-Hardware-Services-for-HW-Queues'} = {
  # match properties
  # note we don't have "Ticket => {" because there's no ticket yet
  Properties => {
    Queue => {
      Name => ['[RegExp]HW'],
    }
  },
  # return possible options
  Possible => {
    # possible ticket options
    Ticket => {
      Service => ['[RegExp]^(Hardware)'],
    },
  },
};

```


1.3. Ссылка

В нижеприведенном Сценарии представлен список всех параметров, которые могут быть использованы для списка прав доступа (ACLs).

Читайте раздел про ACL в документации ProcessManagement с детальным описанием использования ACL для процессных заявок.

Пример 5.6. Ссылка отображает все возможные важные настройки ACL

```
# ticket acl
$self->{TicketAcl}->{'200-ACL-Reference'} = {
  # match properties (current values from the form)
  Properties => {

    # the used frontend module
    Frontend => {
      Action => ['AgentTicketPhone', 'AgentTicketEmail'],
    },

    # the logged in agent
    User => {
      UserLogin => ['some login'],
      Group_rw => [
        'hotline',
      ],
      Role => [
        'admin',
      ],
      # ...
    },

    # the logged in customer
    CustomerUser => {
      UserLogin => ['some login'],
      Group_rw => [
        'hotline',
      ],
      Role => [
        'admin',
      ],
      # ...
    },

    # process properties
    Process => {
      ProcessEntityID      => ['P1'],    # the Process that the current ticket is
      ActivityEntityID     => ['A1'],    # the current Activity of the ticket
      ActivityDialogEntityID => ['AD1'],  # the current ActivityDialog that the Agent/
      Customer is using
    },

    # ticket properties
    Queue => {
      Name      => ['Raw'],
      QueueID   => ['some id'],
      GroupID   => ['some id'],
      Email     => ['some email'],
      RealName  => ['OTRS System'],
      # ...
    },

    Service => {
      ServiceID => ['some id'],
      Name      => ['some name'],
      ParentID  => ['some id'],
      # ...
    },
  },
}
```

```

Type => {
  ID => ['some id'],
  Name => ['some name'],
  # ...
},
Priority = {
  ID => ['some id'],
  Name => ['some name'],
  # ...
},
SLA = {
  SLAID => ['some id'],
  Name => ['some name'],
  Calendar => ['some calendar'],
  # ...
},
State = {
  ID => ['some id'],
  Name => ['some name'],
  TypeName => ['some state type name'],,
  TypeID => ['some state type id'],
  # ...
},
Owner => {
  UserLogin => ['some login'],
  Group_rw => [
    'some group',
  ],
  Role => [
    'admin',
  ],
  # ...
},
Responsible => {
  UserLogin => ['some login'],
  Group_rw => [
    'some group',
  ],
  Role => [
    'admin',
  ],
  # ...
},
DynamicField => {
  # Names must be in DynamicField_<field_name> format.
  # Values in [ ... ] must always be the untranslated internal data keys
  # specified in the dynamic field definition and
  # not the data values shown to the user.
  DynamicField_Field1 => ['some value'],
  DynamicField_OtherField => ['some value'],
  DynamicField_TicketFreeText2 => ['some value'],
  # ...
},
# alternatively, ticket properties can be specified in the ticket hash
Ticket => {
  Queue => ['Raw'],
  State => ['new', 'open'],
  Priority => ['some priority'],
  Lock => ['lock'],
  CustomerID => ['some id'],
  CustomerUserID => ['some id'],
  Owner => ['some owner'],
  DynamicField_Field1 => ['some value'],
  DynamicField_MyField => ['some value'],
  # ...
},
},
# match properties (existing values from the database)
PropertiesDatabase => {
  # See section "Properties", the same config can be used here.
  # ...
}

```

```

}

# return possible options (white list)
Possible => {
  # possible ticket options (white list)
  Ticket => {
    Queue => ['Hotline', 'Coordination'],
    State => ['some state'],
    Priority => ['5 very high'],
    DynamicField_Field1 => ['some value'],
    DynamicField_MyField => ['some value'],
    # ...
    NewOwner => ['some owner'],
    OldOwner => ['some owner'],
    # ...
  },

  # Limit the number of possible ActivityDialogs the Agent/Customer
  # can use in a process ticket.
  ActivityDialog => ['AD1', 'AD3'],

  # Limit the number of possible Processes that can be started
  Process => ['P1', 'P2'],

  # possible action options (white list)
  Action => {
    AgentTicketBounce => 1,
    AgentTicketClose => 1,
    AgentTicketCompose => 0,
    AgentTicketCustomer => 0,
    AgentTicketForward => 0,
    AgentTicketFreeText => 1,
    AgentTicketHistory => 1,
    AgentTicketLink => 0,
    AgentTicketLock => 1,
    AgentTicketMerge => 0,
    AgentTicketMove => 1,
    AgentTicketNote => 1,
    AgentTicketOwner => 1,
    AgentTicketPending => 1,
    AgentTicketPhone => 1, # only used to hide the Split action
    AgentTicketPhoneInbound => 0,
    AgentTicketPhoneOutbound => 1,
    AgentTicketPrint => 1,
    AgentTicketPriority => 0,
    AgentTicketResponsible => 1,
    AgentTicketWatcher => 1,
    AgentTicketZoom => 1,
    AgentLinkObject => 1, # only used to hide the Link action
  },
},

# remove options (black list)
PossibleNot => {
  # See section "Possible"
  # ...
},
};

```

Примечание

While matching ACLs if CustomerUserID parameter sent, the ACL mechanism will compare the defined ACLs using the supplied CustomerUserID to gather the CustomerUser details to fill the CustomerUser hash and it also overrides the Customer information in the Ticket hash for the Properties match. On the other hand this calculations are also made for the PropertiesDatabase part, but using the Ticket Customer as the basis to gather the data.

Обратите внимание, что в Пользовательском Интерфейсе, CustomerUserID всегда отправляется с залогиненным Клиентом.

Be aware that in ticket search screens (AgentTicketSearch and CustomerTicketSearch) the only affected attributes by ACLs are the Dynamic Fields. This means that this screens you can not restrict any other attribute like ticket type, state, queue, etc.

2. Управление Процессами

2.1. Введение

Эта функция в OTRS Позволяет поделировать процессы (потоки работ/задач) в системе управления заявками. Основная идея заключается в том, чтобы задать повторяющиеся процессы и делегировать части работы различным сотрудникам, а также управлять продолжением работы по процессу, в зависимости от определенных условий.

2.2. Пример процесса

Давайте рассмотрим некий пример чтобы все было более наглядно и понятно. Мы будем рассматривать процесс покупки книги:

2.2.1. Запись спроса

Прежде чем заказ будет размещен, запрос на литературу работником будет записан. Следующая книга нужна в нашем примере:

Title: Prozessmanagement für Dummies
Autor: Thilo Knuppertz
ISBN: 3527703713

2.2.2. Утверждение менеджером

Начальник отдела где работает работник должен решить что делать с заказом. В случае отказа, причина отказа должна быть записана старшим по должности. В случае одобрения заказ передается в отдел закупок.

2.2.3. Обрабатывается отделом закупок

Закупка, в данном случае, состоит в том, чтобы найти, где книга может быть заказана на лучших условиях. Если ее нет на рынке, это должно быть записано в заказе. В случае успешного размещения заказа, закупщик записывает поставщика, цену и дату поставки.

2.2.4. Обрабатывается почтовым отделом

Посылка будет доставлена в компанию. Отдел приемки, проверяет посылку и записывает дату приема. Теперь работник - заявитель будет извещен, что его заказ прибыл и готов к выдаче.

2.3. Выполнение примера

Если предположить что заявка действует в этом рабочем потоке как сопроводительный документ, который может принимать сообщения об изменениях, мы получаем полное представление о процессных заявках.

Проанализировав предыдущий пример мы можем выделить следующие пункты:

- Possibilities to record data, let's call them *Activity Dialogs*,

- Проверки, которые могут автоматически реагировать на изменение данных, назовем *Переходы*,
- изменения, которые могут быть применены для обработки заявки после успешного перехода обработки заявки, назовем их *Transition Actions*.

Также нам необходим дополнительный элемент, который не так очевиден:

- A possibility to offer more than just one Activity Dialog to be available. In our example this is needed when the manager must have the choice between "Approve" and "Deny". Let's call this *Activity*.

Now, with Activities, Activity Dialogs, Transitions and Transition Actions we have the necessary tools to model the individual steps of our example. What's still missing is an area where for each work-flow the order of the steps can be specified. Let's call this *Process*. To be able to refer to all these entities later, we will assign to them an abbreviation in parentheses. This abbreviation is based on an internal identification mechanism called EntityIDs.

EntityIDs согласовываются с одной или двумя буквами (в зависимости от части процесса), а затем порядковый номер, например:

- Процесс: 'P1', 'P2' ... 'Pn'.
- Деятельность: 'A1', 'A2' ... 'An'.
- Диалог Активности: 'AD1', 'AD2' ... 'ADn'.
- Переход: 'T1', 'T2' ... 'Tn'.
- Переход действий: 'TA1', 'TA2' ... 'TAn'.

Перед созданием процесса и его частей необходимо подготовить систему. Нужно определить некоторые Очереди, Клиентов и Динамические Поля, а также установить некоторые опции в файле SysConfig.

Создать следующие Очереди:

- Управление
- Сотрудники
- Покупка
- Почтовое отделение

Создаст следующих Пользователей:

- Менеджер
- Сотрудник

Создать следующие Динамически Поля:

- Название

Метка	Название
Тип	Text
Объект	Заявк

- Автор

Метка	Автор
-------	-------

Тип	Text
Объект	Заявк

- ISBN

Метка	ISBN
Тип	Text
Объект	Заявк

- Состояние

Метка	Состояние
Тип	Dropdown
Объект	Заявк
Возможные значения	<ul style="list-style-type: none"> • Утверждение • Отказано в утверждении • Утвержденный • Заказ отменен • Заказ размещен • Посылка получена

Примечание: Используйте именно эти возможные значения для "Ключ" и "Значение" при создании динамических полей.

- Поставщик

Метка	Поставщик
Тип	Text
Объект	Заявк

- Цена

Метка	Цена
Тип	Text
Объект	Заявк

- DeliveryDate

Метка	Дата доставки
Тип	Date
Объект	Заявк

- DateOfReceipt

Метка	Дата получения
Тип	Date
Объект	Заявк

Устанавливает следующие параметры SysConfig:

- 'Ticket::Responsible': Yes
- 'Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicFieldGroups':

Ключ:	Содержание:
Книга	Название, Автор, ISBN
Общий	Состояние
Заказ	Цена, Поставщик, Дата доставки
Отгрузка	DateOfReceipt

- 'Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicField':

Ключ:	Содержание:
Автор	1
DateOfReceipt	1
DeliveryDate	1
ISBN	1
Цена	1
Состояние	1
Поставщик	1
Название	1

Теперь давайте перейдем к описанию реального процесса. На следующем шаге мы определим конкретные объекты, которые нам необходимы.

2.3.1. Процесс (в качестве контейнера)

Для создания нового процесса необходимо перейти к пункту Управление процессами на панели Администратора, после нажатия на который мы попадем на экран Обзора процессов. После создания процесса мы получим возможность создания всех его объектов (всех составляющих процесса).

Примечание

Деятельности, Диалоги Активности, Переходы и Действия переходов определенные в одном процессе будут доступны для всех процессов системы.

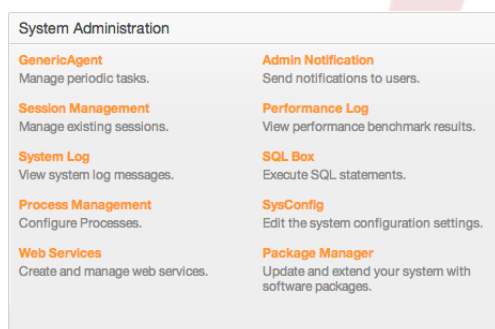


Рисунок: Страница Администратора OTRS - Администрирование системы.

Нажмите на действие "Создать Новый Процесс" в окне Действий.

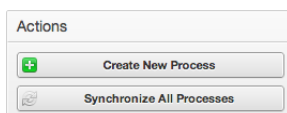


Figure: Создать кнопку Новый Процесс.

Заполняем информацию о процессе, указываем Название Процесса и его Описание, а статус оставим со значением "inactive", пока не закончим все задачи. Сохраняем процесс.

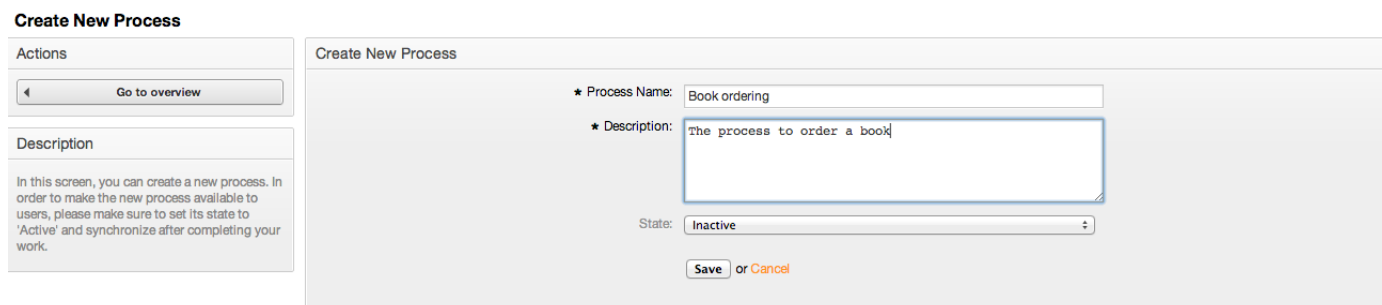


Рисунок Добавить новый процесс.

2.3.2. Диалоги Действий

Кликните по имени процесса в окне Обзора процессов, затем в списке "Доступные элементы процесса" нажмите на "Диалоги Активности" (это действие распахнет список опций для диалогов активности и сожмет другие списки, к нему не относящиеся, напоминает эффект гармошки), затем нажмите на "Создать новый Диалог Активности".

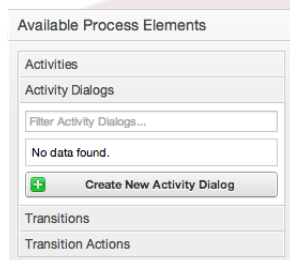
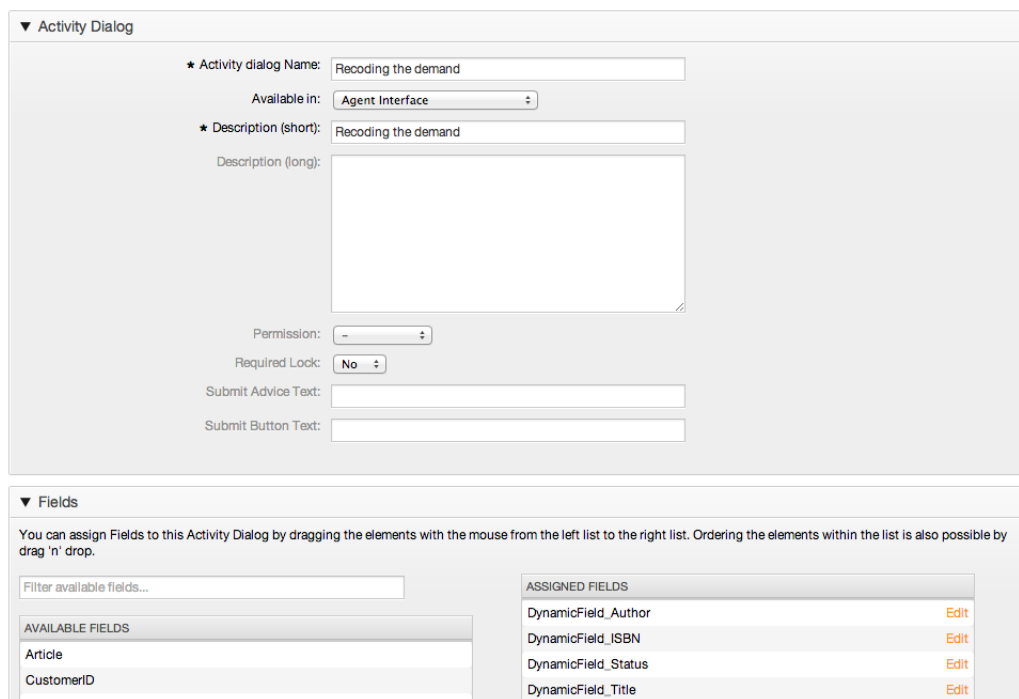


Рисунок: Создать кнопку Новый Диалог Деятельности.

Во вновь открывшемся окне, заполните поле "Имя Диалога Активности", а также "Описание (краткое)", для этого примера мы оставим остальные поля со значениями по умолчанию. Для назначения полей Диалогу Активности просто захватите мышкой нужное поле из списка "Доступные поля" и перенесите в список "Назначенные поля". Их порядок в списка "Назначенные поля" будет соответствовать их порядку на экране. Для изменения порядка просто захватите мышкой нужное поле и переместите его в нужное место.



▼ Activity Dialog

★ Activity dialog Name:

Available in:

★ Description (short):

Description (long):

Permission:

Required Lock:

Submit Advice Text:

Submit Button Text:

▼ Fields

You can assign Fields to this Activity Dialog by dragging the elements with the mouse from the left list to the right list. Ordering the elements within the list is also possible by drag 'n' drop.

Filter available fields...

AVAILABLE FIELDS	ASSIGNED FIELDS
Article	DynamicField_Author Edit
CustomerID	DynamicField_ISBN Edit
DynamicField_AText	DynamicField_Status Edit
	DynamicField_Title Edit

Figure: Добавить новый Диалог Деятельности.

После того как помещены в список "Назначенные поля", другое всплывающее окно покажет некоторые данные о поле. Мы оставим им всем умалчиваемые значения и только для полей типа Сообщение/Заметка мы должны проверить, что они имеют тип "note-internal".



Edit Field Details

Edit Details for Field "Article"

Description (short):

Description (long):

Default value:

ArticleType:

Display:

Figure: Edit field details (Article).

Если все поля привязаны, нажмите на кнопку отправить в главном всплывающем окне чтобы изменения вступили в силу.

В этом примере мы будем использовать поля типа Article для комментариев, другой способ состоит в том, чтобы создать динамическое поле типа TextArea. Оставшиеся из упомянутых полей в строках ниже это динамические поля которые мы задали ранее.

Обратите вниманиче что на этой странице все Динамические Поля имеют префикс "DynamicField_" как в "DynamicField_Title", не путайте с полем "Название", которое является Названием Заявки.

Создать следующие Диалоги Активности:

- "Запись спроса" (AD1)

Диалог Активности, который содержит все требуемые поля для хранения данных о заказе (Название, Автор и ISBN) и поле Состояние/Status с возможностью выбора "Утверждение".

- "Отказать" (AD2)

Диалог Активности с полем комментария и полем статуса с опцией "Отказать".

- "Утвержден" (AD3)

Здесь нам просто нужно поле статуса с опцией "Approved/Утверждено".

- "Заказ отменен" (AD4)

Диалог Активности который позволяет отказаться от покупки (книги нет на складе). Здесь нам также потребуется поле с комментарием и поле статус со значением "Заказ отменен".

- "Заказ размещен" (AD5)

Диалог Активности с полями Поставщик, Цена и Дата доставки для покупки и поле статуса со значением "Заказ размещен".

- "Поставка получена" (AD6)

Диалог Активности для почтового отделения с полем для Даты получения и полем статуса со значением "Поставка получена".

Чтобы ограничить поле состояние для каждого из диалогов активности нам необходимо добавить некоторые ACL-права в файл Kernel/Config.pm или в новый perl-файл в Kernel/Config/Files.

```

$self->{TicketAcl}->{'P1-AD1-1'} = {
    Properties => {
        Process => {
            ActivityDialogEntityID => ['AD1'],
        },
    },
    Possible => {
        Ticket => {
            DynamicField_Status => ['Approval'],
        },
    },
};

$self->{TicketAcl}->{'P1-AD2-1'} = {
    Properties => {
        Process => {
            ActivityDialogEntityID => ['AD2'],
        },
    },
    Possible => {
        Ticket => {
            DynamicField_Status => ['Approval denied'],
        },
    },
};

$self->{TicketAcl}->{'P1-AD3-1'} = {
    Properties => {
        Process => {
            ActivityDialogEntityID => ['AD3'],
        },
    },
};

```

```

Possible => {
  Ticket => {
    DynamicField_Status => ['Approved'],
  },
},
};

$Self->{TicketAcl}->{'P1-AD4-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD4'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Order denied'],
    },
  },
},
};

$Self->{TicketAcl}->{'P1-AD5-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD5'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Order placed'],
    },
  },
},
};

$Self->{TicketAcl}->{'P1-AD6-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD6'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Shipment received'],
    },
  },
},
};

```

2.3.3. Переходы

В "Доступные Элементы Процесса" щелкните "Переходы", а затем кликните на "Создать новый переход".

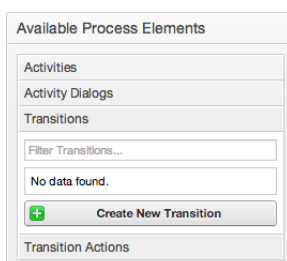


Figure: Создать кнопку Новый Переход.

In the opened popup screen fill the "Transition Name", then in the conditions, for this examples we will use just one condition and just one field, for both we can leave the Type of Linking as "and" and we will use the filed match type value as "String".

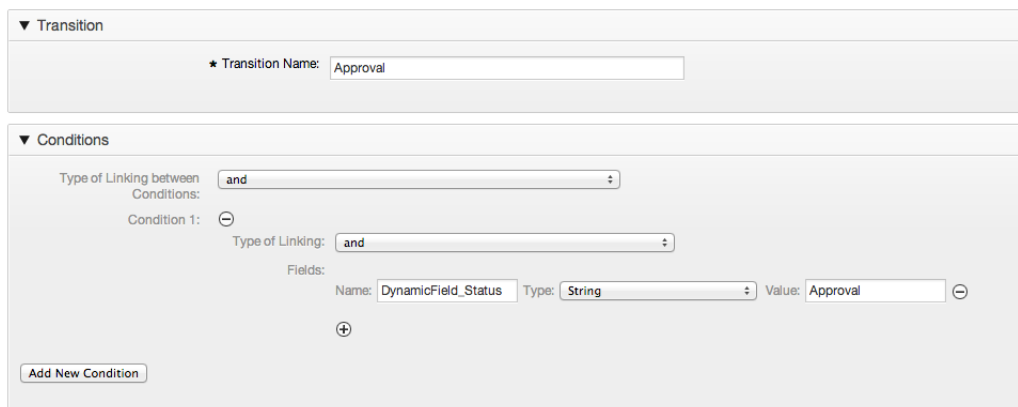


Рисунок: Добавить новый Переход Add new Transition.

Когда все условия установлены, нажмите кнопку Отправить, чтобы сохранить изменения.

Создать следующие Переходы:

- "Утверждение" (T1)

Переход, который проверяет, установлен ли статус поля в "Approval".

- "Отказано в утверждении" (T2)

Переход проверяющий установку статуса поля в значение "Approval denied".

- "Утвержден" (T3)

Переход, проверяющий установку статуса поля в значение "Approved".

- "Заказ отменен" (T4)

Переход проверяющий установку статуса поля в значение "Order denied".

- "Заказ размещен" (T5)

Переход проверяющий установку статуса поля в значение "Order denied".

- "Поставка получена" (T6)

Переход, который проверяет установлено ли поле статуса в значение "Shipment received".

2.3.4. Действия Перехода

Кликните на "Действия Переходов" в меню "Доступные элементы процесса", а затем кликните "Создать Новое Действие Перехода".

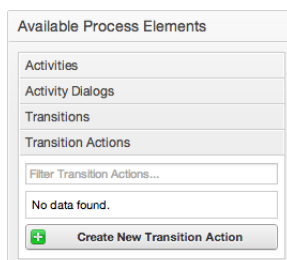


Рисунок: Кнопка создания нового перехода.

В открывшемся всплывающем окне заполните поле "Название Перехода" и "Модуль Деятельности Перехода", а затем добавьте требуемые и опциональные параметры, названия и значения.

Все Модули Переходов Действий находятся в Kernel/System/ProcessManagement/TransitionAction и ниже приводится список Переходов Действий включенных в этот релиз

- DynamicFieldSet
- TicketArticleCreate
- TicketCustomerSet
- TicketLockSet
- TicketOwnerSet
- TicketQueueSet
- TicketResponsibleSet
- TicketServiceSet
- TicketSLASet
- TicketStateSet
- TicketTitleSet
- TicketTypeSet

Каждый модуль имеет свои собственные и другие параметры, пожалуйста просмотрите документацию к этому модулю [module documentation](#), чтобы узнать все обязательные и опциональные параметры.

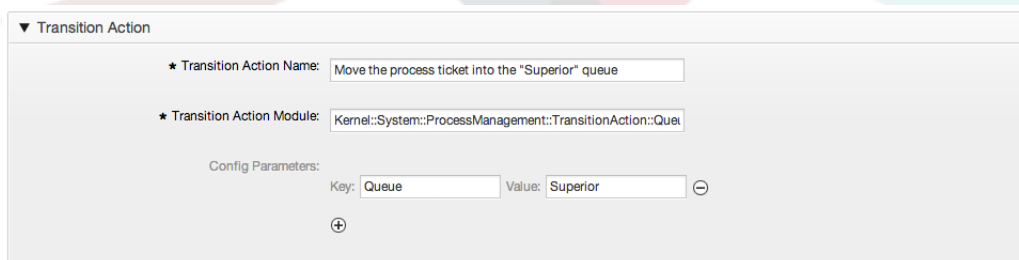


Рисунок: Добавить новый переход деятельности.

Как только все параметры и значения установлены, нажмите кнопку отправить чтобы изменения вступили в силу.

Создать следующие Переходы:

- "Переместить обработку Заявки в очередь 'Менеджмент' " (TA1)

Предполагается, что это действие будет выполнено когда применен Переход "Утверждение" (T1).

- "Изменяет ответственного за заявку на 'manager'" (TA2)

Будет выполнено когд применен Переход "Approval" (T1).

- "Перемещает процесс обработки заявки в очередь 'Employees' "" (TA3)

Будет выполнено когда:

- Применение Перехода "Отказано в утверждении" (T2)
- Применение Перехода "Order denied" (T4)
- Применение Перехода "Поставка получена" (T6)
- "Изменить ответственного за заявку на 'Employee'" (TA4)

Будет выполнено когда:

- Применен переход "Отказано в утверждении" (T2)
- Применен переход "Заказ отменен" (T4)
- Применен переход "Поставка получена" (T6)
- "Перемещает обработку Заявки в очередь 'Purchasing' " (TA5)

Будет выполнено когда применен Переход "Утверждено" (T3).

- "Перемесить процессную заявку в очередь "Post office" " (TA6)

Будет выполнено когда применен Переход "Заказ размещен" (T5).

- "Закрыть Заявку со статусом успешно" (TA7)

Будет выполнено когда:

- Применен переход "Поставка получена" (T6)
- "Закрыть заявку со статусом не успешно" (TA8)

Будет выполнено когда:

- Применен переход "Отказано в утверждении" (T2)
- Применен переход "Заказ отменен" (T4)

As you can see, there are places where the same Transition Actions should be executed. Therefore it is reasonable to make it possible to link Transition Actions freely with Transitions to be able to reuse them.

2.3.5. Деятельность

Мы выбрали такой подход чтобы рассматривать Деятельности в качестве корзины, которая может хранить один или больше Диалогов Деятельности. Нам нужны эти действия в нашем примере:

Кликните на "Деятельности" в меню "Доступные Элементы Процесса", а затем кликните на "Создать новую Деятельность".

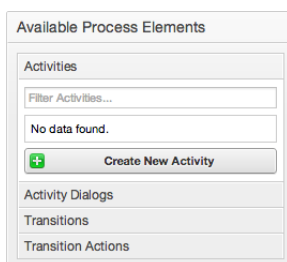


Figure: Create New Activity button.

In the opened popup screen fill the "Activity Name", Then drag the required Activity Dialogs from the "Available Activity Dialogs" pool, and drop them into the "Assigned Activity Dialogs" pool. These dialogs will be presented (in the ticket zoom screen) in the same order as it is defined on this screen translating from top to bottom, to left to right.

Такой порядок особенно важен в первой Деятельности, так как первый Диалог Деятельности для этой Деятельности является единственным, который находится при запуске процесса

Создать следующие Деятельности:

- "Recording the demand" (A1)
Contains the Activity Dialog "Recording the demand" (AD1)
- "Approval/Утверждение" (A2)
Содержит Диалоги Активности "Отказано в одобрении" (AD2) а также "Утвержден" (AD3)
- "Заказ" (A3)
Содержит Диалоги Активности "Заказ отклонен" (AD4) а также "Заказ размещен" (AD5)
- "Входящие" (A4)
Contains the Activity Dialog "Shipment received" (AD6)
- "Process complete" (A5): This is an Activity without possible Activity Dialogs. It will be set after "Approval denied", "Order denied" or "Shipment received" and represents the end of the process.

Теперь мы можем видеть, что Деятельность точно определяет состояние процесса Заявки. После успешного Перехода Заявка перемещается из одного вида деятельности в другой.

2.3.6. Схема процесса заказа книги

Давайте закончим наш пример с последней недостающей частью головоломки - Процессом, как описанием потока работ. В нашем случае это весь рабочий процесс заказа. Другие процессы могут быть похожими, типа заказа офисных принадлежностей, или совершенно иными.

The process has a starting point which consists of the start Activity and the start Activity Dialog. For any new book order, the start Activity Dialog (first Activity Dialog for the first Activity) is the first screen that is displayed. If this is completed and saved, the Process ticket will be created and can follow the configured work-flow.

The process also contains the directions for how the process ticket can move through the Process. Let's call this the "Path". It consists of the start Activity, one or more Transitions (possibly with Transition Actions), and other Activities.

Assuming that the Activities has already assigned their Activity Dialogs drag an Activity from the accordion (in the left part of the screen) and drop it into the canvas area (below process information). Notice that an arrow from the process start (green circle) to the Activity is placed automatically. (This is the first Activity and its first Activity Dialog is the first screen that will be shown when the process starts).

Available Process Elements

Activities

Filter Activities...

- Recording the demand ✎ 🗑
- Approval ✎ 🗑
- Order ✎ 🗑
- Incoming ✎ 🗑
- Process complete ✎ 🗑

+ Create New Activity

Activity Dialogs

Transitions

Transition Actions

Description

The Elements listed above in this sidebar can be moved to the canvas area on the right by using drag'n'drop.

Activities: You can place Activities on the canvas area to assign this Activity to the Process.


Activity Dialogs: To assign an Activity Dialog to an Activity drop the Activity Dialog element from this sidebar over the Activity placed in the canvas area.

Transitions: You can start a connection between to Activities by dropping the Transition element over the Start Activity of the connection. After that you can move the loose end of the arrow to the End Activity.

Transition Actions: Actions can be assigned to a Transition by dropping the Action Element onto the label of a Transition.

State: Inactive

▼ Add and Edit Activities, Activity Dialogs and Transitions



Save or Save and finish or Cancel

Рисунок: Помещение первой Активности на полотно схемы.

Next, drag another Activity into the canvas too. now we will have two Activities in the canvas the first one is connected to the start point and the second has no connections, you can hover the mouse over each activity to reveal their own Activity Dialogs.

Available Process Elements

Activities

Filter Activities...

- Recording the demand ✎ 🗑
- Approval ✎ 🗑
- Order ✎ 🗑
- Incoming ✎ 🗑
- Process complete ✎ 🗑

+ Create New Activity

Activity Dialogs

Transitions

Transition Actions

Description

The Elements listed above in this sidebar can be moved to the canvas area on the right by using drag'n'drop.

Activities: You can place Activities on the canvas area to assign this Activity to the Process.

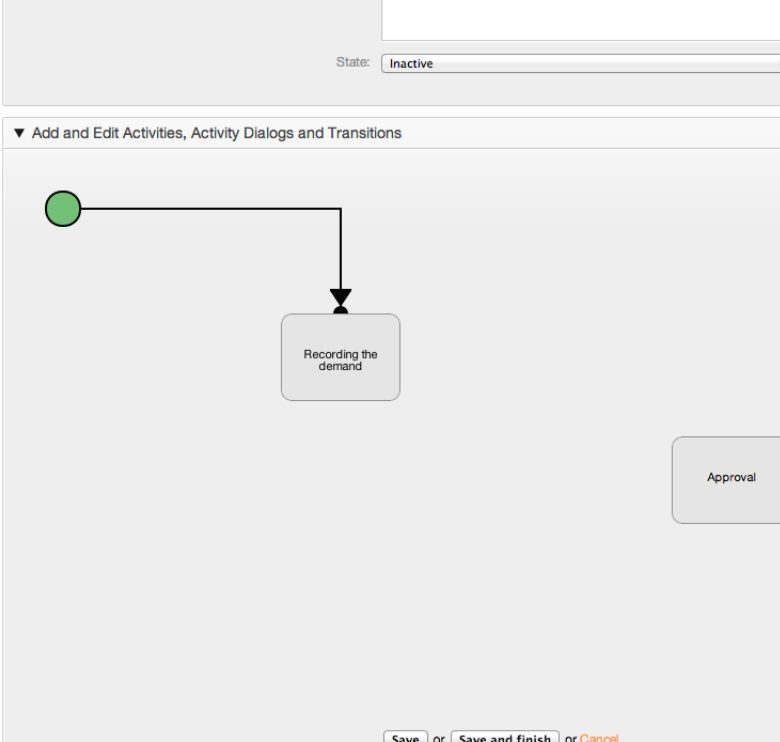
Activity Dialogs: To assign an Activity Dialog to an Activity drop the Activity Dialog element from this sidebar over the Activity placed in the canvas area.

Transitions: You can start a connection between to Activities by dropping the Transition element over the Start Activity of the connection. After that you can move the loose end of the arrow to the End Activity.

Transition Actions: Actions can be assigned to a Transition by dropping the Action Element onto the label of a Transition.

State: Inactive

▼ Add and Edit Activities, Activity Dialogs and Transitions



Save or Save and finish or Cancel

Figure: Drag second Activity into the canvas.

Then let's create the "Path" (connection) between this two Activities, for this we will use the Transitions, Click on Transitions in the accordion drag a Transition and drop it inside the first Activity, notice that the Activity change its color indicating that the Transition is attached, as soon as the Transition is dropped the end point of the Transition arrow will be placed next to the process start point. Drag the Transition arrow end point and drop it inside the other Activity to create the connection between the Activities.

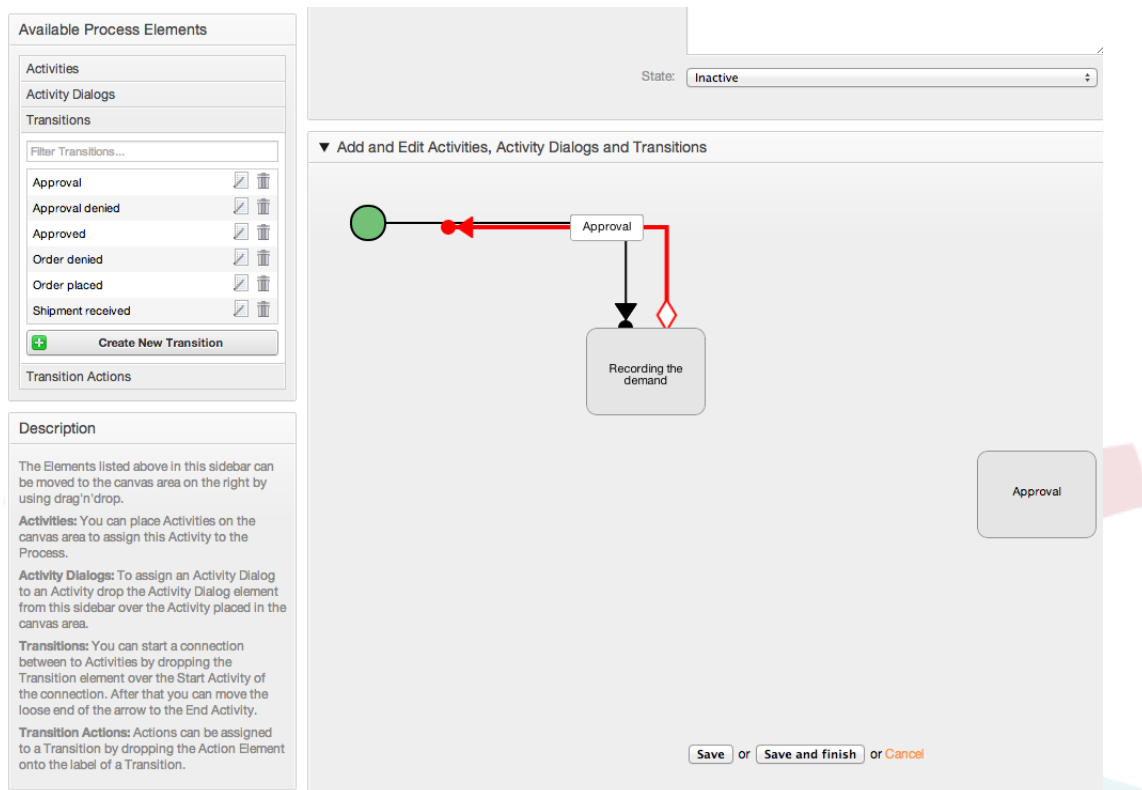


Figure: Drag a Transition into the canvas.

Now that the "Path" between the Actions is defined, then we need to assign the Transition Actions to the Transition, double click the Transition label (in the canvas), this will open a new popup window.

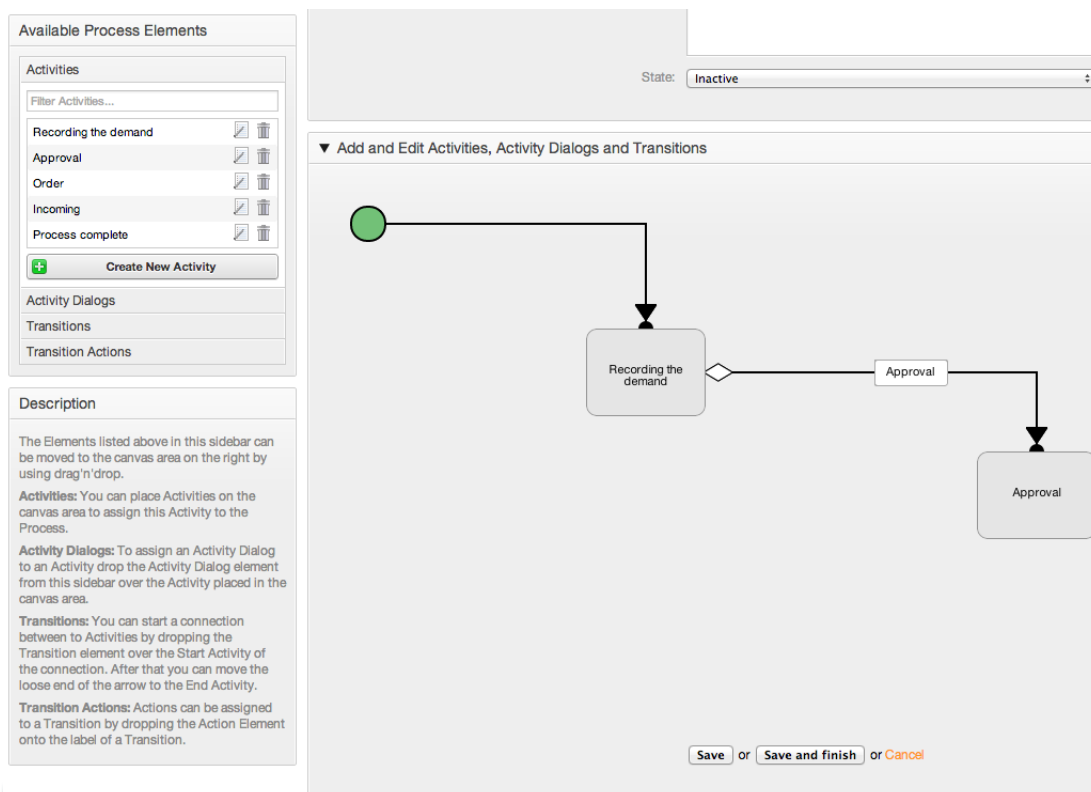


Рисунок: Связать Активности используя Переходы.

Захватите мышью требуемые Активности Перехода из списка Доступных Активностей Перехода и перетащите в список Назначенных Активностей Перехода и затем нажмите кнопку Отправить.

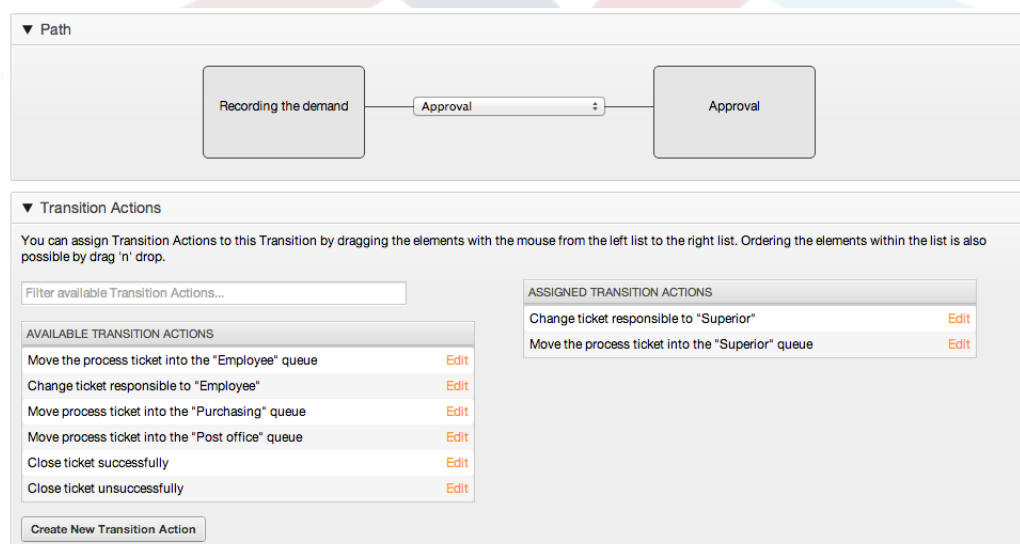


Рисунок: Назначение Действий Перехода.

Затем, вернитесь к главному окну редактирования Процесса и нажмите кнопку Сохранить под схемой и сохраните все сделанные изменения.

Завершите "путь" добавляя следующие Активности, Переходы и Действия Переходов:

Запись спроса до значения "Approval"

- Начальная точка: Activity: Запись запроса (A1)
- Возможный Переход: Утверждение (T1)
 - If the condition of this activity is fulfilled, the ticket will move to Activity: Approval (A2)
 - Кроме того выполнится следующее действие TransitionActions:
 - "Переместить заявку в очередь 'Менеджмент' " (TA1)
 - "Set ticket responsible to 'manager'" (TA2)

The Activity: "Recording the demand" (A1) is a defined step of the process ticket, where there is the possibility for the Transition: "Approval" (T1). If this applies, the ticket will move to the next Activity: "Approval" (A2), and the Transition Actions: "Moving the ticket into the 'Management' Queue" (TA1) and "Set ticket responsible to 'manager'" (TA2) are executed. In the Activity: "Approval" (A2), the Activity Dialogs: "Approval denied" (AD2) and "Approved" (AD3) are available.

Утверждение

- Starting Point: Activity "Approval" (A2)
- Возможные Транзакции:
 - "Отказано в утверждении" (T2)
 - If this matches, the process ticket will move to Activity:"Process complete" (A5).
 - Additionally, the following Transition Actions are executed:
 - "Move process ticket to the 'Employees' Queue" (TA3)
 - "Set ticket responsible to 'Employee'" (TA4)
 - "Закреть заявку со статусом не успешно" (TA8)
 - "Утвержден" (T3)
 - If this matches, the process ticket will move to Activity: "Order" (A3).
 - Additionally, the following Transition Action is executed:
 - "Move process ticket to 'Purchasing' Queue" (TA5)

We can see that from the current Activity, which defines a step of the process ticket, there are one or more possibilities for Transition which have exactly one target Activity (and possibly one or more Transition Actions).

Заказ

- Starting Point: Activity "Order" (A3)
- Возможные Транзакции:
 - "Заказ отменен" (T4)
 - If this matches, the process ticket will move to Activity:"Process complete" (A5).
 - Additionally, the following Transition Actions are executed:
 - "Move process ticket to the 'Employees' Queue" (TA3)

- "Set ticket responsible to 'Employee'" (TA4)
- "Закрывать заявку со статусом не успешно" (TA8)
- "Заказ размещен" (T5)
 - If this matches, the process ticket will move to Activity: "Incoming" (A5).
 - Additionally, the following Transition Action is executed:
 - "Move process ticket to 'Post office' Queue" (TA6)

Входящие

- Starting Point: Activity "Incoming" (A4)
- Возможные Транзакции:
 - "Поставка получена" (T6)
 - If this matches, the process ticket will move to Activity: "Process complete" (A5).
 - Additionally, the following Transition Actions are executed:
 - "Move process ticket to the 'Employees' Queue" (TA3)
 - "Set ticket responsible to 'Employee'" (TA4)
 - "Закрывать Заявку со статусом успешно" (TA7)

Полный путь к Процессу покупки книги будет иметь следующий вид:

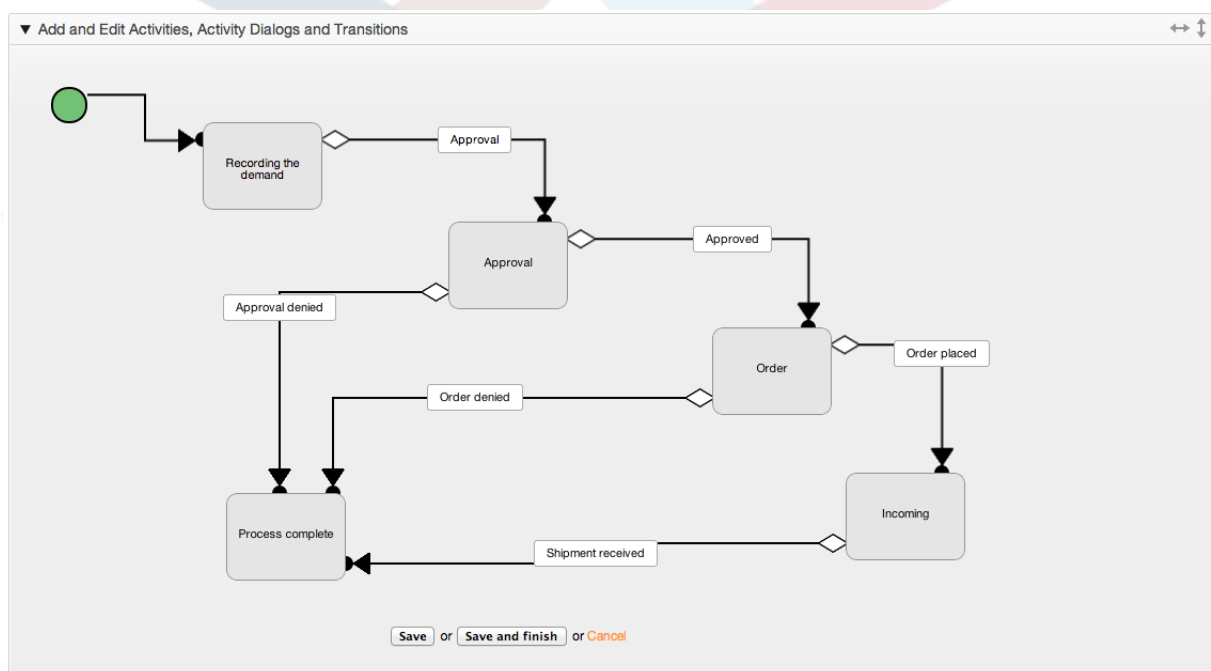


Figure: Book ordering complete process path.

После окончания прорисовки Процесса нажмите кнопку "Сохранить", в нижней части схемы и затем, нажмите "Синхронизировать все процессы". При этом вся информация по процессам будет собрана из базы данных и будет создан кэш-файл (на языке Perl). Этот файл содержит конфигурацию процессов, которая будет использоваться для создания и обработки процессных заявок.

Все изменения сделанные в Процессе (в интерфейсе админа) потребуют пересинхронизации файла кэша, чтобы изменения были отражены в системе.

Также, возможно импортировать процесс целиком из YAML файла, но обязательно до этого создать все динамические поля, пользователей, очереди и пр. необходимые для каждого процесса.

Обратите внимание, что если процесс требует использования ACL, они также должны быть созданы вручную.

Ниже приведен полный текст YAML файла для процесса заказа книги из примера:

```
---
Activities:
  A1:
    ActivityDialogs:
      - AD1
    ChangeTime: 2012-11-23 14:49:22
    Config:
      ActivityDialog:
        1: AD1
    CreateTime: 2012-11-23 11:49:38
    EntityID: A1
    ID: 151
    Name: Recording the demand
  A2:
    ActivityDialogs:
      - AD2
      - AD3
    ChangeTime: 2012-12-13 00:55:12
    Config:
      ActivityDialog:
        1: AD2
        2: AD3
    CreateTime: 2012-11-23 11:50:11
    EntityID: A2
    ID: 152
    Name: Approval
  A3:
    ActivityDialogs:
      - AD4
      - AD5
    ChangeTime: 2012-11-23 18:12:14
    Config:
      ActivityDialog:
        1: AD4
        2: AD5
    CreateTime: 2012-11-23 11:50:35
    EntityID: A3
    ID: 153
    Name: Order
  A4:
    ActivityDialogs:
      - AD6
    ChangeTime: 2012-11-23 18:12:35
    Config:
      ActivityDialog:
        1: AD6
    CreateTime: 2012-11-23 11:51:00
    EntityID: A4
    ID: 154
    Name: Incoming
  A5:
    ActivityDialogs: []
    ChangeTime: 2012-11-23 11:51:33
    Config: {}
    CreateTime: 2012-11-23 11:51:33
    EntityID: A5
    ID: 155
    Name: Process complete
```

```
ActivityDialogs:
AD1:
  ChangeTime: 2012-12-06 02:16:21
  Config:
    DescriptionLong: ''
    DescriptionShort: Recording the demand
    FieldOrder:
      - DynamicField_Author
      - DynamicField_ISBN
      - DynamicField_Title
      - DynamicField_Status
    Fields:
      DynamicField_Author:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_ISBN:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Status:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Title:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
    Interface:
      - AgentInterface
    Permission: ''
    RequiredLock: 0
    SubmitAdviceText: ''
    SubmitButtonText: ''
  CreateTime: 2012-11-23 14:34:43
  EntityID: AD1
  ID: 154
  Name: Recording the demand
AD2:
  ChangeTime: 2012-11-23 14:57:41
  Config:
    DescriptionLong: ''
    DescriptionShort: Approval denied
    FieldOrder:
      - Article
      - DynamicField_Status
    Fields:
      Article:
        Config:
          ArticleType: note-internal
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Status:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
    Interface:
      - AgentInterface
    Permission: ''
    RequiredLock: 0
    SubmitAdviceText: ''
    SubmitButtonText: Deny Request
  CreateTime: 2012-11-23 14:36:39
  EntityID: AD2
  ID: 155
```

```
Name: Approval denied
AD3:
ChangeTime: 2012-12-14 03:14:23
Config:
  DescriptionLong: ''
  DescriptionShort: Approved
  FieldOrder:
    - DynamicField_Status
  Fields:
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
  Interface:
    - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: Approve Request
CreateTime: 2012-11-23 14:37:35
EntityID: AD3
ID: 156
Name: Approved
AD4:
ChangeTime: 2012-11-23 14:58:52
Config:
  DescriptionLong: ''
  DescriptionShort: Order rejected
  FieldOrder:
    - Article
    - DynamicField_Status
  Fields:
    Article:
      Config:
        ArticleType: note-internal
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Status:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
  Interface:
    - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: Reject Order
CreateTime: 2012-11-23 14:38:48
EntityID: AD4
ID: 157
Name: Order rejected
AD5:
ChangeTime: 2012-12-06 02:20:12
Config:
  DescriptionLong: ''
  DescriptionShort: Order placed
  FieldOrder:
    - DynamicField_DeliveryDate
    - DynamicField_Price
    - DynamicField_Supplier
    - DynamicField_Status
  Fields:
    DynamicField_DeliveryDate:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Price:
```

```

    DefaultValue: ''
    DescriptionLong: ''
    DescriptionShort: ''
    Display: 1
  DynamicField_Status:
    DefaultValue: ''
    DescriptionLong: ''
    DescriptionShort: ''
    Display: 1
  DynamicField_Supplier:
    DefaultValue: ''
    DescriptionLong: ''
    DescriptionShort: ''
    Display: 1
  Interface:
  - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: Place Order
  CreateTime: 2012-11-23 14:41:28
  EntityID: AD5
  ID: 158
  Name: Order placed
AD6:
  ChangeTime: 2012-11-23 14:42:43
  Config:
    DescriptionLong: ''
    DescriptionShort: Shipment received
    FieldOrder:
    - DynamicField_DateOfReceipt
    - DynamicField_Status
  Fields:
    DynamicField_DateOfReceipt:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
  Interface:
  - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: ''
  CreateTime: 2012-11-23 14:42:43
  EntityID: AD6
  ID: 159
  Name: Shipment received
Process:
  Activities:
  - A1
  - A2
  - A3
  - A4
  - A5
  ChangeTime: 2012-12-06 02:31:59
  Config:
    Description: The process to order a book
  Path:
    A1:
      T1:
        ActivityEntityID: A2
        TransitionAction:
        - TA2
        - TA1
    A2:
      T2:

```



```
    ActivityEntityID: A5
    TransitionAction:
      - TA3
      - TA4
      - TA8
  T3:
    ActivityEntityID: A3
    TransitionAction:
      - TA5
  A3:
    T4:
      ActivityEntityID: A5
      TransitionAction:
        - TA3
        - TA4
        - TA8
    T5:
      ActivityEntityID: A4
      TransitionAction:
        - TA6
  A4:
    T6:
      ActivityEntityID: A5
      TransitionAction:
        - TA3
        - TA4
        - TA7
  A5: {}
  StartActivity: A1
  StartActivityDialog: AD1
  CreateTime: 2012-11-23 11:45:12
  EntityID: P1
  ID: 94
  Layout:
    A1:
      left: 172
      top: 63
    A2:
      left: 402
      top: 156
    A3:
      left: 649
      top: 255
    A4:
      left: 774
      top: 391
    A5:
      left: 194
      top: 410
  Name: Book ordering
  State: Active
  StateEntityID: S1
  TransitionActions:
    - TA1
    - TA2
    - TA3
    - TA4
    - TA8
    - TA5
    - TA3
    - TA4
    - TA8
    - TA6
    - TA3
    - TA4
    - TA7
  Transitions:
    - T1
    - T2
    - T3
    - T4
    - T5
```

```
- T6
TransitionActions:
TA1:
  ChangeTime: 2012-11-23 16:01:37
  Config:
    Config:
      Queue: Management
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 15:50:59
  EntityID: TA1
  ID: 61
  Name: Move the process ticket into the "Management" queue
TA2:
  ChangeTime: 2012-11-23 16:02:12
  Config:
    Config:
      Responsible: manager
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet
  CreateTime: 2012-11-23 15:58:22
  EntityID: TA2
  ID: 62
  Name: Change ticket responsible to "manager"
TA3:
  ChangeTime: 2012-11-24 14:27:02
  Config:
    Config:
      Queue: Employees
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:02:54
  EntityID: TA3
  ID: 63
  Name: Move the process ticket into the "Employees" queue
TA4:
  ChangeTime: 2012-11-23 16:04:06
  Config:
    Config:
      Responsible: Employee
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet
  CreateTime: 2012-11-23 16:04:06
  EntityID: TA4
  ID: 64
  Name: Change ticket responsible to "Employee"
TA5:
  ChangeTime: 2012-12-06 02:18:34
  Config:
    Config:
      Queue: Purchasing
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:04:54
  EntityID: TA5
  ID: 65
  Name: Move process ticket into the "Purchasing" queue
TA6:
  ChangeTime: 2012-12-06 02:18:48
  Config:
    Config:
      Queue: Post office
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:06:20
  EntityID: TA6
  ID: 66
  Name: Move process ticket into the "Post office" queue
TA7:
  ChangeTime: 2012-12-06 02:29:55
  Config:
    Config:
      State: closed successful
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketStateSet
  CreateTime: 2012-12-06 02:29:27
  EntityID: TA7
  ID: 67
  Name: Close ticket successfully
```

```
TA8:
  ChangeTime: 2012-12-06 02:31:12
  Config:
    Config:
      State: closed unsuccessful
      Module: Kernel::System::ProcessManagement::TransitionAction::TicketStateSet
    CreateTime: 2012-12-06 02:31:12
    EntityID: TA8
    ID: 68
    Name: Close ticket unsuccessfully
Transitions:
  T1:
    ChangeTime: 2012-11-23 15:12:20
    Config:
      Condition:
        1:
          Fields:
            DynamicField_Status:
              Match: Approval
              Type: String
          Type: and
      ConditionLinking: and
    CreateTime: 2012-11-23 11:53:52
    EntityID: T1
    ID: 94
    Name: Approval
  T2:
    ChangeTime: 2012-11-23 15:12:50
    Config:
      Condition:
        1:
          Fields:
            DynamicField_Status:
              Match: Approval denied
              Type: String
          Type: and
      ConditionLinking: and
    CreateTime: 2012-11-23 11:54:26
    EntityID: T2
    ID: 95
    Name: Approval denied
  T3:
    ChangeTime: 2012-11-23 15:13:29
    Config:
      Condition:
        1:
          Fields:
            DynamicField_Status:
              Match: Approved
              Type: String
          Type: and
      ConditionLinking: and
    CreateTime: 2012-11-23 11:54:54
    EntityID: T3
    ID: 96
    Name: Approved
  T4:
    ChangeTime: 2012-11-23 15:14:08
    Config:
      Condition:
        1:
          Fields:
            DynamicField_Status:
              Match: Order denied
              Type: String
          Type: and
      ConditionLinking: and
    CreateTime: 2012-11-23 11:55:25
    EntityID: T4
    ID: 97
    Name: Order denied
  T5:
```

```

ChangeTime: 2012-11-23 18:30:33
Config:
  Condition:
    1:
      Fields:
        DynamicField_Status:
          Match: Order placed
          Type: String
      Type: and
      ConditionLinking: and
    CreateTime: 2012-11-23 11:56:15
    EntityID: T5
    ID: 98
    Name: Order placed
T6:
  ChangeTime: 2012-11-23 15:15:30
  Config:
    Condition:
      1:
        Fields:
          DynamicField_Status:
            Match: Shipment received
            Type: String
        Type: and
        ConditionLinking: and
      CreateTime: 2012-11-23 11:56:48
      EntityID: T6
      ID: 99
      Name: Shipment received
  
```

2.4. Process configuration reference

2.4.1. Процесс

Процесс моделирует путь прохождения задач процесса. Путевыми точками на этом пути могут быть Активности или Переходы, мы расскажем об этом позже.

2.4.1.1. Настройка Процесса

Конфигурация процесса может быть выполнена в файле Kernel/Config.pm, но мы настоятельно рекомендуем создать новый файл Kernel/Config/Files/MyProcess.pm. Заметьте, что генератор процессов в GUI создает файл Kernel/Config/File/ZZZProcessManagement, поэтому избегайте использование этого имени, иначе он будет перезаписан при синхронизации процессов. Рассмотрим пример конфигурации процесса (из кэш файла):

```

$self->{'Process'} = {
  'P1' => {
    Name           => 'Book order',
    CreateTime     => '16-02-2012 13:37:00',
    CreateBy      => '1',
    ChangeTime    => '17-02-2012 13:37:00',
    ChangeBy      => '1',
    State         => 'Active',
    StartActivity => 'A1',
    StartActivityDialog => 'AD1',
    Path => {
      'A1' => {
        'T1' => {
          ActivityEntityID => 'A2',
        },
      },
    },
    'A2' => {
      'T2' => {
        ActivityEntityID => 'A3',
      },
    },
  },
}
  
```

```

    },
  },
},
'P2' => {
  Name           => 'IT order',
  CreateTime     => '26-02-2012 13:37:00',
  CreateBy       => '1',
  ChangeTime     => '27-02-2012 13:37:00',
  ChangeBy       => '1',
  State          => 'Active',
  StartActivity  => 'A2',
  StartActivityDialog => 'AD2',
  Path => {
    'A2' => {
      'T3' => {
        ActivityEntityID => 'A4',
      },
    },
  },
},
};

```

2.4.1.2. Название

Имя Процесса, оно может быть выбрано агентом при создании новой процессной заявки.

2.4.1.3. CreateTime

Время, когда процесс был создан.

2.4.1.4. CreateBy

The UID of the user creating the process.

2.4.1.5. ChangeTime

The time when the process was changed.

2.4.1.6. ChangeBy

UID пользователя, который вносил последние изменения в процесс.

2.4.1.7. State

Задаёт состояние процесса. Возможные значения:

- 'Active' / Активный - все процессы, которые могут быть использованы в новых процессных заявках.
- 'FadeAway' / Исчезающие - это процессы, которые нельзя выбрать более для новых заявок, но существующие заявки могут продолжать их использовать.
- 'Inactive' / Неактивные - деактивированные процессы. Они не могут быть более использованы ни в новых, ни в существующих заявках.

2.4.1.8. StartActivity

При создании новой процессной заявки, StartActivity / Начальная Активность должна быть задана. Как только заявка будет создана, эта Активность будет инициализирована и использована как начальная точка для первых проверок переходов.

2.4.1.9. StartActivityDialog

For new process tickets, a StartActivityDialog must be defined. This will be shown when creating a new process ticket (after the process was selected). At this point, the ticket does not exist yet, it will be created after submitting the StartActivityDialog.

2.4.1.10. Путь

The Path contains the structure of the Activities, and the possible Transitions between them, for the current process. And also the Transition Actions that happens when transitioning . This controls the way that a process ticket can take. Example:

```
'A1' => {
  'T1' => {
    ActivityEntityID => 'A2',
  },
  'T2' => {
    ActivityEntityID => 'A3',
  },
  'T3' => {
    ActivityEntityID => 'A4',
    TransitionAction => ['TA1', 'TA2'],
  },
},
```

If a process ticket is in Activity 'A1', it has three possible ways to get to another Activity. In the Transitions 'T1' to 'T3', conditions are defined, that a process ticket must fulfill to move (transit) to another Activity.

If in this case all the values of the process ticket and its dynamic fields that are needed for the Transition 'T2' are correct, the ticket will be moved from Activity 'A1' to 'A3'. After an ActivityDialog is submitted, or any other change is made to a ticket, it will be checked for possible Transitions from the current Activity. If multiple Transitions are possible, the first one will be used (based on numerical sorting of the TransitionIDs).

Additionally, it is possible to assign Transition Actions to Transitions in the Path configuration. These are modules which are executed after a successful Transition. They have to be specified in array form as in the example, we'll talk about the details later.

2.4.2. АКТИВНОСТЬ

An Activity contains one or more Activity Dialogs and models a 'step' in the process. All Activity Dialogs of the current Activity are displayed in the ticket zoom and can be used until the conditions of a Transition are fulfilled.

2.4.2.1. Настройка Активности

Давайте рассмотрим пример конфигурации активности:

```
$Self->{'Process::Activity'} =
{
  'A1' => {
    Name      => 'Activity 1 optional',
    CreateTime => '16-02-2012 13:37:00',
    CreateBy  => '1',
    ChangeTime => '17-02-2012 13:37:00',
    ChangeBy  => '1',
    ActivityDialog => {
      1 => 'AD1',
    },
  },
},
```

```
'A2' => {
  Name      => 'Activity 2 optional',
  CreateTime => '16-02-2012 13:37:00',
  CreateBy  => '1',
  ChangeTime => '17-02-2012 13:37:00',
  ChangeBy  => '1',
  ActivityDialog => {
    1 => 'AD5',
    2 => 'AD6',
    3 => 'AD1',
  },
},
};
```

2.4.2.2. Название

Название активности.

2.4.2.3. CreateTime

Время создания.

2.4.2.4. CreateBy

UID of the user who created the Activity.

2.4.2.5. ChangeTime

The last time when it was changed.

2.4.2.6. ChangeBy

UID of the last user who changed the Activity.

2.4.2.7. ActivityDialog

Activity Dialog contains the list of Activity Dialogs which are available in this Activity. All Activity Dialogs of the current Activity are displayed in the ticket zoom. Their order is set by the order in the configuration, here 'AD5' is shown before 'AD6' and 'AD1'.

2.4.3. ActivityDialog

An Activity Dialog is a particular screen and can be used in different Activities.

2.4.3.1. ActivityDialog configuration

Let's see an example config

```
$Self->{'Process::ActivityDialog'} = {
  'AD1' => {
    Name      => 'ActivityDialog 1 optional',
    DescriptionShort => 'Basic info',
    DescriptionLong  => 'Please insert the necessary basic information for IT orders',
    CreateTime      => '28-02-2012 13:37:00',
    CreateBy        => '1',
    ChangeTime      => '29-02-2012 13:37:00',
    ChangeBy        => '1',
    Fields => {
      PriorityID => {
        DescriptionShort => 'Priority ID',
        DescriptionLong  => 'Enter the priority here',
      }
    }
  }
}
```

```

        Display      => 2,
    },
},
FieldOrder      => [ 'PriorityID' ],
SubmitAdviceText => 'Note: If you submit the form...',
SubmitButtonText => 'Send request',
},
'AD2' => {
    Name => 'ActivityDialog 2 optional',
    DescriptionShort => 'Basic info',
    DescriptionLong => 'Please insert the necesesary basic information for Book
orders',
    CreateTime      => '28-02-2012 13:37:00',
    CreateBy        => '1',
    ChangeTime      => '29-02-2012 13:37:00',
    ChangeBy        => '1',
    Fields => {
        StateID => {
            DescriptionShort => 'State ID',
            DescriptionLong  => 'Enter the state here',
            Display          => 2,
            DefaultValue     => '2',
        },
        Queue => {
            DescriptionShort => 'Queue ID',
            DescriptionLong  => 'Enter the queue here',
            Display          => 2,
            DefaultValue     => 'Raw',
        },
        Title => {
            DescriptionShort => 'Title',
            DescriptionLong  => 'Enter the title here',
            Display          => 1,
            DefaultValue     => 'Default Title',
        },
    },
    DynamicField_Anzahl => {
        DescriptionShort  => 'Amount',
        DescriptionLong   => 'Enter the amount here',
        Display           => 2,
        DefaultValue     => '4',
    },
},
FieldOrder      => [ 'DynamicField_Anzahl', 'StateID', 'Queue', 'Title' ],
SubmitAdviceText => 'Note: If you submit the form...',
SubmitButtonText => 'Send request',
},
};

```

2.4.3.2. Название

Имя Диалога Активности.

2.4.3.3. CreateTime

Время, когда он был создан.

2.4.3.4. CreateBy

UID пользователя, создавшего этот Диалог Активности.

2.4.3.5. ChangeTime

Время последнего изменения.

2.4.3.6. ChangeBy

UID пользователя, последним изменявшего этот Диалог Активности.

2.4.3.7. Поля

Содержит все поля, которые могут быть отображены в этом Диалоге Активности. Следующие поля могут быть использованы в данный момент:

```
Title
State
StateID
Priority
PriorityID
Lock
LockID
Queue
QueueID
Customer
CustomerID
CustomerNo
CustomerUserID
Owner
OwnerID
Type
TypeID
SLA
SLAID
Service
ServiceID
Responsible
ResponsibleID
PendingTime
DynamicField_FieldName # for all dynamic fields
```

Example of a single field configuration:

```
StateID => {
  DescriptionShort => 'State ID',
  DescriptionLong  => 'Enter the state here',
  Display         => 2,
  DefaultValue    => '2',
},
```

The field "Article" is a special case. If it is present in a "Fields" configuration, the Activity Dialog will contain a complete Richtext editor with subject field and attachment handling. The entered text will then be added to the ticket as an article and sent by email. Let's see an example Article field configuration:

```
Article => {
  DescriptionShort => 'Please insert your comment here.',
  DescriptionLong  => '',
  Display         => 1,
  Config          => {
    ArticleType   => 'note-internal',
    LabelSubject  => '',
    LabelBody     => '',
  },
},
```

Рассмотрим опции конфигурации полей:

2.4.3.7.1. Краткое описание

Необязательное краткое описание, которое может отображаться вместе с заголовком поля.

2.4.3.7.2. Полное описание

Необязательное полное описание поля, отображаемое при наведении мыши на поле, например, указания по заполнению этого поля.

2.4.3.7.3. Отобразить

Задаёт, должно ли поле отображаться и/или обязательно. Возможные значения:

- "0": поле не отображается. Это может быть полезно, если значения полей заносятся автоматически. В этом случае будет использовано значение, заданное по умолчанию.
- "1": поле отображается, но заполнение его необязательно.
- "2": поле отображается и обязательно для заполнения. Следующие поля могут быть только неотображаемыми и обязательными:

```
QueueID
Queue
State
StateID
Lock
LockID
Priority
PriorityID
Type
TypeID
```

Если поля заданы как необязательные, и никакие значения не вводятся пользователем, умалчиваемое значение будет присвоено этому полю, когда пользователь нажимает Отправить для этого Диалога Активности.

2.4.3.7.4. Значение по умолчанию

Для полей с суффиксом "ID" (таких как QueueID, OwnerID), система использует идентификатор в базе данных в качестве значения. Для других полей без "ID" (таких как Queue, Owner), значение по умолчанию должно содержать собственно значение. Например:

```
Queue => {
  DescriptionShort => 'Queue',
  DescriptionLong => 'Enter the queue here',
  Display         => 2,
  DefaultValue    => 'Raw',
},
```

2.4.3.8. Порядок полей

Здесь задается порядок отображения полей. ВАЖНО: Неотображаемые поля также должны быть заданы здесь, т.к. только описанные поля учитываются при сохранении. Значения полей не указанных здесь, не сохраняются.

2.4.3.9. Текст подсказки для кнопки

Необязательный текст, отображаемый поверх указателя мыши при наведении на кнопку, содержит дополнительную подсказку или поясняющий текст.

2.4.3.10. Текст надписи на кнопке

Необязательный/альтернативный пользовательский текст/название для кнопки Отправить.

2.4.4. Переход

Результат Перехода - основанный на заданных условиях - дальнейший путь Процесса, т.е. к какой Активности должна перейти процессная заявка.

2.4.4.1. Конфигурирование Перехода

Let's see an example:

```
$Self->{'Process::Transition'} = {
  'T1' => {
    Name => 'Transition 1',
    CreateTime => '14-03-2012 13:37:00', # optional
    CreateBy => '1', # optional
    ChangeTime => '15-03-2012 13:37:00', # optional
    ChangeBy => '15-03-2012 13:37:00', # optional
    Condition => {
      Cond1 => {
        Fields => {
          StateID => {
            Type => 'String',
            Match => '1',
          },
        },
      },
    },
  },
  'T2' => {
    Name => 'Transition 2 optional',
    CreateTime => 'DATE', # optional
    CreateBy => 'USERID', # optional
    ChangeTime => 'DATE', # optional
    ChangeBy => 'USERID', # optional
    Condition => {
      Cond1 => {
        Queue => 'Raw',
        DynamicField_Farbe => '2',
        DynamicField_Anzahl => '1',
      },
    },
  },
},
};
```

2.4.4.2. Название

Имя Перехода

2.4.4.3. CreateTime

Время, когда он был создан.

2.4.4.4. CreateBy

UID пользователя, создавшего этот Переход

2.4.4.5. ChangeTime

Время последнего изменения.

2.4.4.6. ChangeBy

UID пользователя, последним изменявшего этот Переход.

2.4.4.7. Условие

Содержит все условия, необходимые для осуществления этого Перехода. Пример:

```
Condition => {
  Type => 'and',
  Cond1 => {
    Type => 'and',
    Fields => {
      StateID => {
        Type => 'String',
        Match => '1',
      },
      DynamicField_Marke => {
        Type => 'String',
        Match => 'VW',
      },
    },
  },
  Cond2 => {
    Type => 'and',
    Fields => {
      Queue => {
        Type => 'String',
        Match => 'Raw',
      },
    },
  },
},
```

Давайте рассмотрим подробно конфигурирование Условий.

2.4.4.7.1. Тип (Условия)

Задаёт операторы для соединения элементов Условия. Возможные значения:

- "and": Умалчиваемое значение. Все условия должны быть верны для выполнения Перехода.
- "or"/"или": хотябы одно из условий должно быть верно.
- "xor": должно быть верным не более одного условия.

2.4.4.7.2. Cond1

Это имя одного из условий примера. Имя выбирается произвольно. Условия вычисляются в порядке очередности.

2.4.4.7.3. Type (Cond)

Задаёт способ, которым отдельные поля, проверяющие это условие, соединены друг с другом. Возможные значения:

- 'and': This is the default. All field tests must match for this condition to match.
- 'or': At least one field test must match.
- 'xor': Exactly one field test must match, not more.

2.4.4.7.4. Поля

Specifies the particular fields whose values should be tested. From our example:

```
Fields => {
  StateID => {
```

```
Type => 'String',  
Match => '1',  
},
```

2.4.4.7.5. StateID

Пример имени поля. Следующие имена полей заявки могут использоваться:

```
Title  
State  
StateID  
Priority  
PriorityID  
Lock  
LockID  
Queue  
QueueID  
Customer  
CustomerID  
CustomerNo  
CustomerUserID  
Owner  
OwnerID  
Type  
TypeID  
SLA  
SLAID  
Service  
ServiceID  
Responsible  
ResponsibleID  
DynamicField_FieldName # for all DynamicFields
```

Когда проверяются поля с ID в имени (такие как SLAID), используется ID поля из базы данных, для других полей (таких как SLA), используется его текущее значение.

2.4.4.7.6. Type

Определяет тип проверяемого поля. Возможные значения:

- 'String': Compares the field value with the string specified in 'Match'. Matches if they are exactly the same.
- 'Hash': Compares the field value (hash) with the hash specified in 'Match'. All hash values must be the same.
- 'Array': Compares the field value (array) with the array specified in 'Match'. Both lists must be the same.
- 'Regex': The field value can be tested with a regular expression. It is important that 'Match' contains `qr{ }xms` as a base condition. Between the braces the actual regular expression can be noted.
- 'Module': Allows you to use a perl module for condition checking. If it returns 1, the check was positive. You can find an example module in `Kernel/System/ProcessManagement/TransitionValidation/ValidateDemo.pm`.

2.4.5. Действия Перехода

Действия Перехода это действия которые могут быть выполнены после успешного Перехода (когда процессная заявка переходит от одной Активности к другой). Эти Действия Перехода могут быть использованы для выполнения различных изменений в заявке, например, изменить Очередь или Владельца заявки, и вы можете создать

свои собственные Действия Переходов для осуществления других сложных изменений.

2.4.5.1. Настройка Действий Перехода

Рассмотрим пример:

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Queue Move',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',  
    Config => {  
      Queue => 'Junk',  
      UserID => 123,  
    },  
  },  
};
```

2.4.5.2. Название

Имя Действия Перехода

2.4.5.3. Модуль

Задаёт используемый Perl модуль.

2.4.5.4. Config

Этот параметр содержит все настройки, требуемые для модуля. Их содержание зависит от конкретного используемого модуля Действия Перехода. Детали по конкретным модулям смотрите в документации. В нашем примере, только Очередь должна быть задана. Тем не менее мы также передаём UserID. Используя параметр UserID действие перехода будет выполняться от имени пользователя с этим UserID.

Использование UserID внутри параметра "Config" для Действия Перехода прелема для всех Действий Перехода (начиная с OTRS 3.2.4), в этом примере это может быть в частности важным, если пользователь, выполняющий Переход, не имеет прав для передачи заявки в очередь "Junk", тогда как пользователь с UserID = 123, может их иметь.

2.4.5.5. Повторное использование модулей Действий Переходов.

Для многократного использования модулей Действий Переходов, достаточно задать несколько Действий Переходов в ваших настройках. Пример:

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Queue Move Junk',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',  
    Config => {  
      Queue => 'Junk',  
    },  
  },  
  'TA2' => {  
    Name => 'Queue Move Raw',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',  
    Config => {  
      Queue => 'Raw',  
    },  
  },  
};
```

Здесь, один и тот же модуль используется для передачи процессной заявки в очередь "Raw", и в другой раз для перемещения в очередь "junk". Действие Перехода, используемое для конкретного Перехода определяется в параметре "Path" настроек Процесса.

2.4.5.6. Доступные Действия Перехода

OTRS стандартно имеет в составе несколько Действий Перехода, которые могут быть использованы в ваших процессах. Здесь вы можете найти их описание и способ настройки.

2.4.5.6.1. DynamicFieldSet

Задаёт одно или несколько динамических полей в процессной заявке. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set DynamicField Master to Master and Approved to 1',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::DynamicFieldSet',
    Config => {
      MasterSlave => 'Master',
      Approved => '1',
    },
  },
};
```

"Имя" задаёт имя конфигурируемого Действия Перехода.

'Главный/Подчиненный' и 'Утверждение' даны как пример имен динамических полей. Значения полей ("Главный" и "1") будут установлены этим Действием Перехода.

2.4.5.6.2. TicketArticleCreate

Создаёт сообщение/заметку и может быть использован для создания заметок или почтовых ответов. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Article Create Note Internal',
    Module =>
'Kernel::System::ProcessManagement::TransitionAction::TicketArticleCreate',
    Config => {
      ArticleType => 'note-internal', #
note-external|phone|fax|sms|... #
      excluding any email type #
      SenderType => 'agent', #
agent|system|customer
      ContentType => 'text/plain; charset=ISO-8859-15', # or
optional Charset & MimeType
      Subject => 'some short description', #
      required #
      Body => 'the message text', #
      required #
      HistoryType => 'OwnerUpdate', #
EmailCustomer|Move|AddNote|PriorityUpdate|WebRequestCustomer|...
      HistoryComment => 'Some free text!', #
      From => 'Some Agent <email@example.com>', #
      not required but useful #
      To => 'Some Customer A <customer-a@example.com>', #
      not required but useful #
      Cc => 'Some Customer B <customer-b@example.com>', #
      not required but useful #
    },
  },
};
```

```

ReplyTo          => 'Some Customer B <customer-b@example.com>',      #
not required
InReplyTo        => '<asdasdasd.12@example.com>',                    #
not required but useful
References        => '<asdasdasd.1@example.com> <asdasdasd.12@example.com>', #
not required but useful
NoAgentNotify    => 0,                                              # if
you don't want to send agent notifications
AutoResponseType => 'auto reply',                                  #
auto reject|auto follow up|auto reply/new ticket|auto remove

ForceNotificationToUserID => [ 1, 43, 56 ],                          # if
you want to force somebody
ExcludeNotificationToUserID => [ 43, 56 ],
    # if you want full exclude somebody from notfications,
    # will also be removed in To: line of article,
    # higher prio as ForceNotificationToUserID
ExcludeMuteNotificationToUserID => [ 43, 56 ],
    # the same as ExcludeNotificationToUserID but only the
    # sending gets muted, agent will still shown in To:
    # line of article
    },
  },
};

```

"Имя" задает имя настраиваемого Действия Перехода. Его можно выбрать произвольно, но желательно, чтобы оно отражало цель действия.

"Тип заметки/сообщения" задает тип создаваемой заметки. Возможные значения: phone, fax, sms, webrequest, note-internal, note-external and note-report.

Тип отправителя задает тип отправителя для заметки/сообщения. Возможные значения: агент, система, клиент.

"ContentType"/"Тип содержания" задает тип содержимого заметки/сообщения. Возможные значения: 'text/plain; charset=ISO-8859-15' или любой другая допустимая кодировка или mime тип.

Тема - задает заголовок сообщения/заметки. Обязательна.

"Тело" - задает содержимое сообщения/заметки. Указывать обязательно.

HistoryType - задает тип записи истории. Возможные значения: AddNote, ArchiveFlagUpdate, Bounce, CustomerUpdate, EmailAgent, EmailCustomer, EscalationResponseTimeNotifyBefore, EscalationResponseTimeStart, EscalationResponseTimeStop, EscalationSolutionTimeNotifyBefore, EscalationSolutionTimeStart, EscalationSolutionTimeStop, EscalationUpdateTimeNotifyBefore, EscalationUpdateTimeStart, EscalationUpdateTimeStop, FollowUp, Forward, Lock, LoopProtection, Merged, Misc, Move, NewTicket, OwnerUpdate, PhoneCallAgent, PhoneCallCustomer, PriorityUpdate, Remove, ResponsibleUpdate, SendAgentNotification, SendAnswer, SendAutoFollowUp, SendAutoReject, SendAutoReply, SendCustomerNotification, ServiceUpdate, SetPendingTime, SLAUpdate, StateUpdate, Subscribe, SystemRequest, TicketDynamicFieldUpdate, TicketLinkAdd, TicketLinkDelete, TimeAccounting, TypeUpdate, Unlock, Unsubscribe, WebRequestCustomer.

"HistoryComment" описывает содержимое записи истории.

'From', 'To', 'Cc' and 'ReplyTo' take email addresses in the notation specified above.

'InReplyTo' and 'References' take email message IDs.

"NoAgentNotify"/Не информировать агента - если установлено в "1", уведомление агенту посылаться не будет.

"AutoResponseType"/"Тип автоответа" - может принимать следующие значения: auto follow up/автоотклик, auto reject/автоотказ, auto remove/автоудаление, auto reply/автоответ, auto reply/new ticket/автоответ-новая заявка.

'ForceNotificationToUserID', 'ExcludeNotificationToUserID', 'ExcludeMuteNotificationToUserID' может содержать список UserIDs, которые либо всегда оповещаются, не оповещаются, либо помечаются как оповещенные, но на самом деле им не отправлено уведомление по электронной почте.

2.4.5.6.3. TicketCustomerSet

Задаёт клиента процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Customer Set Customer to test',
    Module => 'Kernel::System::Process::TransitionAction::TicketCustomerSet',
    Config => {
      No => 'test',
      User => 'client-user-123',
      # or in other words
      # CustomerID => 'client123',
      # CustomerUserID => 'client-user-123',
    },
  },
};
```

"Имя" задаёт имя конфигулируемого Действия Перехода.

No or CustomerID set the Customer ID of the customer.

User or CustomerUserID set the Username of the customer.

2.4.5.6.4. TicketLockSet

Устанавливает состояние блокирования процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set Lock to lock',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketLockSet',
    Config => {
      Lock => 'lock',
      # or
      LockID => 2,
    },
  },
};
```

"Имя" задаёт имя конфигулируемого Действия Перехода.

'Lock' defines the new lock of the process ticket.

'LockID' defines the internal ID of the new lock.

2.4.5.6.5. TicketOwnerSet

Изменяет владельца этой процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Owner Set root@localhost',
```

```
Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketOwnerSet',
Config => {
  Owner => 'root@localhost',
  # or
  OwnerID => 1,
},
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"Owner"/"Владелец" - задает логин нового владельца.

"OwnerID" задает внутренний ID нового владельца.

2.4.5.6.6. TicketQueueSet

Перемещает заявку в указанную очередь. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Queue Move Raw',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',
    Config => {
      Queue => 'Raw',
      # or
      # QueueID => '2',
    },
  },
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"Queue" задает имя очереди для перемещения.

"QueueID" задает внутренний ID для очереди для перемещения.

2.4.5.6.7. TicketResponsibleSet

Изменяет владельца процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Responsible Set root@localhost',
    Module =>
'Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet',
    Config => {
      Responsible => 'root@localhost',
      # or
      ResponsibleID => 1,
    },
  },
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"Responsible" задает логин нового ответственного.

"ResponsibleID" задает внутренний ID нового ответственного.

2.4.5.6.8. TicketServiceSet

Назначает сервис процессной заявке. Заявка должна иметь клиента и сервис должен быть назначен этому клиенту. Пример:

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Set MyService service',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketServiceSet',  
    Config => {  
      Service => 'MyService',  
      # or  
      ServiceID => 123,  
    },  
  },  
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"Service" задает новый сервис для процессной заявки. Требуется задать полное имя сервиса (например, GrandFatherService::FatherService::SonService).

"ServiceID" задает внутренний ID нового сервиса.

2.4.5.6.9. TicketSLASet

Задаёт SLA для процессной заявки. Заявке должен быть уже назначен сервис и SLA должен быть привязан к этому сервису. Пример:

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Set MySLA SLA',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketSLASet',  
    Config => {  
      SLA => 'MyService',  
      # or  
      SLAID => 123,  
    },  
  },  
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"SLA" задает новое соглашение об уровне сервиса для процессной заявки.

"SLAID" задает внутренний ID нового SLA.

2.4.5.6.10. TicketStateSet

Изменяет состояние процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Set State to open',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketStateSet',  
    Config => {  
      State => 'open',  
      # or  
      StateID => 4,  
      PendingTimeDiff => 123,  
    },  
  },  
};
```

"Имя" задает имя конфигулируемого Действия Перехода.

"State" задает новое состояние процессной заявки.

"StateID" задает внутренний ID нового состояния.

"PendingTimeDiff" используется только для состояний с ожиданием, задает относительную разность в секундах (относительно заданного времени выполнения Действия Перехода) для установки времени ожидания для заявки (например, 3600 означает что устанавливается ожидание в 1 час после заданного времени выполнения Действия Перехода).

2.4.5.6.11. TicketTitleSet

Задает Тему процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set Ticket Title to Ticket-title',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketTitleSet',
    Config => {
      Title => 'Ticket-title',
    },
  },
};
```

"Имя" задает имя конфигурируемого Действия Перехода.

"Title" задает новую тему заявки.

2.4.5.6.12. TicketTypeSet

Задает Тип процессной заявки. Пример:

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set Ticket Type to default',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketTypeSet',
    Config => {
      Type => 'default',
      # or
      # TypeID => '1',
    },
  },
};
```

"Имя" задает имя конфигурируемого Действия Перехода.

"Type"/"Тип" - задает название типа заявки.

"TypeID" задает внутренний ID для типа заявки.

2.4.6. Списки Контроля Доступа (ACLs)

С помощью ACL, вы можете ограничить выбор значений в процессной заявке. Смотрите также ACL reference для обзора полного синтаксиса ACL для заявок.

2.4.6.1. Настройка ACL

ACL могут быть заданы только в Kernel/Config.pm. /Правильнее с помощью редактора ACL в SysConfig/. Пример:

```
$Self->{TicketACL}->{'001-ACL-ProcessProperties'} = {
  Properties => {
    Process => {
```

```

        ProcessEntityID      => ['P1'],
        ActivityEntityID     => ['A1'],
        ActivityDialogEntityID => ['AD1'],
    }
},
Possible => {
    ActivityDialog => ['AD1', 'AD3'],
},
PossibleNot => {
    ActivityDialog => ['AD3'],
},
};

```

2.4.6.2. 001-ACL-ProcessProperties

Имя правила ACL. Для более полной информации обратитесь к the ACL manual.

2.4.6.3. Процесс

Эта секция предназначена для проверки может ли правило ACL быть применено. Если параметр имеет заданное значение, правило применяется. Могут быть использованы следующие значения:

2.4.6.3.1. ProcessEntityID

The ID of a process that the process. Matches if the ticket is assigned to this process.

2.4.6.3.2. ActivityEntityID

ID Активности, которой назначена процессная заявка в текущий момент.

2.4.6.3.3. ActivityDialogEntityID

ID Диалога Активности открытого в настоящий момент для процессной заявки.

2.4.6.4. Possible/PossibleNot Activity Dialog

Здесь вы можете задать список ID'ов Диалогов Активности. Этот список ограничивает Диалоги Активности, которые могут быть предложены пользователю при просмотре заявки.

"Possible" показывает список допустимых Диалогов Активности. Установка выше, допускает только 'AD1' и 'AD3' из списка заданных Диалогов Активности.

'PossibleNot' отображает список недопустимых Диалогов Активности. В примере выше, настройка удаляет 'AD3' из списка заданных Диалогов Активности.

Если оба 'Possible' и 'PossibleNot' заданы, список заданных Диалогов Активности сначала фильтруется используя 'Possible', оставляя 'AD1' и 'AD3' в нашем примере. Затем применяется фильтр 'PossibleNot', который удаляет 'AD3', оставляя таким образом только 'AD1', как доступный пользователю для использования Диалог Активности.

Если множество правил ACL удовлетворяются, их пересечение будет вычислено для определения доступных Диалогов Активности. Пример:

Заданные Диалоги Активности: 'AD1', 'AD2', 'AD3', 'AD4', 'AD5', 'AD6', 'AD7'.

```

$self->{TicketAcl}->{'001-ACL-Status'} = {
    Properties => {
        Ticket => {
            Status => 'new',
        }
    }
}

```

```

    },
    Possible => {
      ActivityDialog => ['AD1', 'AD2', 'AD3', 'AD6', 'AD7'],
    },
  };
$self->{TicketAcl}->{'002-ACL-Queue'} = {
  Properties => {
    Ticket => {
      Queue => ['Raw']
    }
  },
  Possible => {
    ActivityDialog => ['AD2', 'AD3', 'AD4', 'AD7'],
  },
};
$self->{TicketAcl}->{'003-ACL-Priority'} = {
  Properties => {
    Ticket => {
      Priority => ['3 normal']
    }
  },
  PossibleNot => {
    ActivityDialog => ['AD3', 'AD4'],
  },
};

```

Если заявка имеет состояние 'new', находится в очереди 'Raw' и имеет приоритет '3 normal', то все три правила ACL будут удовлетворяться.

Первое правило сокращает список Диалогов Активности с 'AD1', 'AD2', 'AD3', 'AD4', 'AD5', 'AD6', 'AD7' до 'AD1', 'AD2', 'AD3', 'AD6', 'AD7' и запрещает 'AD4' and 'AD5'.

Второе правило еще сокращает список оставшихся Диалогов Активности. В нашем примере, остаются 'AD2', 'AD3', 'AD7'.

Третье правило еще более сокращает список с помощью 'PossibleNot'. 'AD3' удаляется из списка. 'AD4' не будет удален, т.к. он не на первом месте в списке. И, наконец, 'AD2' и 'AD7' останутся, как доступные для выбора пользователем Диалоги Активности.

Также, можно ограничить список Процессов, которые будут отображаться на экране "Новая процессная заявка", способ аналогичен ограничению списка Диалогов Активности, за одним исключением: ACL должны базироваться на данных пользователя. Т.е. список привязывается к пользователю, создающему заявку.

Смотрите примеры ниже:

```

$self->{TicketAcl}->{'200-ACL-Process'} = {
  # match properties
  Properties => {
    User => {
      UserID => [2, 3],
    },
  },
  Possible => {
    Process => ['P1', 'P2', 'P3'],
  },
  PossibleNot => {
    Process => ['P4'],
  },
};

```

```

$self->{TicketAcl}->{'201-ACL-Process'} = {
  # match properties

```

```
Properties => {
  User => {
    Group_rw => [ 'MyGroup' ],
  },
},
Possible => {
  Process => ['P1', 'P2', 'P3'],
},
PossibleNot => {
  Process => ['P4'],
},
};
```

```
$Self->{TicketAcl}->{'202-ACL-Process'} = {
  # match properties
  Properties => {
    User => {
      Role => [ 'MyRole' ],
    },
  },
  Possible => {
    Process => ['P1', 'P2', 'P3'],
  },
  PossibleNot => {
    Process => ['P4'],
  },
};
```

3. Создание своих собственных тем (шаблонов)

Для OTRS можно создавать свои собственные темы и использовать в качестве веб-интерфейса тот шаблон, который вам нравится больше всего. Для создания собственной темы нужно настроить шаблоны вывода под свои нужды.

Более подробную информацию о синтаксисе и структуре шаблонов вывода можно найти в Руководстве Разработчика по ссылке <http://otrs.github.io/doc>, особенно в разделе [Шаблоны](#).

В качестве примера, выполните следующие шаги для создания новой темы, которая называется "Компания":

1. Создайте директорию и назовите ее `Kernel/Output/HTML/Company` и скопируйте все файлы, которые хотите изменить, из `Kernel/Output/HTML/Standard` в новую директорию.

Важно

Копируйте только те файлы, которые вы действительно планируете изменять. OTRS автоматически получит недостающие файлы из Стандартной темы. На более поздних этапах такой подход позволит сделать модернизацию более простой.

2. Настройте файлы в директории `Kernel/Output/HTML/Company` и измените шаблоны по своему вкусу.
3. To activate the new theme, add them in SysConfig under `Frontend::Themes`.

Теперь новая тема должна быть доступна. Вы можете выбрать эту тему перейдя по ссылке [предпочтения](#).

Предупреждение

Не изменяйте файлы темы, поставляемой с OTRS, поскольку эти изменения будут утеряны после обновления. Создавайте свои собственные темы только путем выполнения шагов, описанных выше.

4. Локализация интерфейса OTRS

OTRS предлагает многоязычную поддержку для своего веб-интерфейса.

Информацию о том, как локализовать OTRS-фреймворк, какие шаги нужно выполнить чтобы создать перевод для нового языка, или как можно создать настройки для перевода можете найти, перейдя по ссылке "[Локализация](#)" раздел в руководстве разработчика на <http://otrs.github.io/doc>.



Глава 6. Настройка Производительности

Ниже представлен перечень различных техник, которые можно использовать для получения максимально возможной производительности системы OTRS: настройка, кодирование, использование памяти и многое другое.

1. OTRS

Есть несколько возможных вариантов улучшения производительности OTRS.

1.1. TicketIndexModule

Есть два типа хранилища информации для хранения индексов заявок:

- `Kernel::System::Ticket::IndexAccelerator::RuntimeDB` (по умолчанию), "на лету" генерирует из таблицы заявок шаблон просмотра для каждой очереди. Пока в системе не больше 60, 000 открытых заявок, проблем с производительностью не будет.
- `Kernel::System::Ticket::IndexAccelerator::StaticDB` - один из самых мощных модулей, его следует использовать, когда у вас есть больше 80.000 открытых заявок. Он использует дополнительную таблицу `ticket_index`, которая работает как обычный шаблон. Используйте `bin/otrs.RebuildTicketIndex.pl` для получения начального индекса после создания хранилища информации.

Вы можете изменить `IndexAccelerator` через `SysConfig`.

1.2. TicketStorageModule

Существует два различных хранилища для хранения заявок/статей:

- Задайте `Kernel::System::Ticket::ArticleStorageDB` (по умолчанию) для хранения вложений, и т.д. в базе данных. Примечание: Не используйте его для систем с большим объемом данных.

Pro: Если пользователь вашего веб-сервера не является пользователем 'otrs', используйте этот модуль, чтобы избежать проблем с правами доступа.

Противопоказания: Не совсем целесообразно хранить вложения в базе данных. Будьте осторожны, при хранении больших объектов. Настройте конфигурационный параметр MySQL `"set-variable=max_allowed_packet=8M"` для хранения объектов размером 8 MB (по умолчанию это 2M).

- Configure `Kernel::System::Ticket::ArticleStorageFS` для хранения вложений и т.д. в локальной файловой системе. Примечание: Рекомендовано при больших установках.

Pro: Это быстро!

Противопоказания: Пользователь веб-сервера должен быть пользователем 'otrs'. Кроме того, если у вас есть несколько серверов, нужно убедиться в том, что файловая система разделяется между этими серверами. Так что разместите его на общедоступный NFS-ресурс или предпочтительнее SAN, или аналогичные решения.

Примечание: можно переключаться с одного бэк-енда на другой "на лету". Бэк-энды можно переключить в `SysConfig` и затем запустить в командной строке утилиту

otrs.ArticleStorageSwitch.pl, чтобы внести статьи из базы данных в файловую систему и наоборот. Можно использовать опции `-s` и `-d` для указания источника и конечного бэк-энда. Обратите внимание, что весь процесс может занять много времени, в зависимости от количества статей/сообщений/заметок, которые у вас есть и доступной мощности процессора и/или пропускной способности сети.

```
shell> bin/otrs.ArticleStorageSwitch.pl -s ArticleStorageDB -d ArticleStorageFS
```

Сценарий Переключение хранилища данных с базы данных на файловую систему.

1.3. Архивирование Заявок

Поскольку OTRS может использоваться в качестве системы аудита доказательств, то удаление закрытых заявок не очень хорошая идея. Именно по этому мы реализовали функцию архивирования заявок.

Заявки, которые удовлетворяют определенным критериям, могут быть отмечены как "архивированные". При обычном поиске заявок или при поиске заявок с помощью задания Generic Agent, они будут недоступны. Сама собой система не будет иметь дела с огромным количеством заявок, до тех пор, пока не понадобится обратиться к "очень старой" заявке. Это приводит к огромному приросту производительности на больших системах.

Для использования функции архивации выполните следующие действия:

1. Включение архивирования системы в SysConfig

В Панели Администрирования перейдите в SysConfig и выберите группу Заявка. В Core::Ticket найдите опцию Ticket::ArchiveSystem, по умолчанию установленную в значение "нет". Измените значение этой настройки на "да" и сохраните изменения.

2. Определение работы GenericAgent

В Панели Администратора выберите GenericAgent и добавьте новое задание (работу).

a. Настройки Задания

Введите имя для работы архивирования, и выберите надлежащие опции для планирования этой работы.

b. Ticket Filter

Фильтр заявок производит поиск заявок, которые отвечают выбранным критериям. Хорошей идеей будет заархивировать закрытые заявки, которые были закрыты за несколько месяцев до этого.

c. Действия над заявками

В этой части установите поле "Архивировать выбранные заявки" в "архив заявок".

d. Сохранить работу

В конце страницы у вас будет возможность сохранить работу (задание).

e. Обработанные заявки

Система отобразит все заявки, которые будут заархивированы при выполнении задания Generic Agent-ом.

3. Поиск Заявок

При поиске заявок, система по умолчанию производит поиск среди не архивированных заявок. Установите критерий поиска "поиск в архивах", если нужно чтобы поиск происходил также и в архивированных заявках.

2. База данных

Решения зависят от используемой базы данных. Изучайте документацию к используемой базе данных или же обратитесь за помощью к администратору.

2.1. MySQL

Если для MySQL-таблиц используется MyISAM (а по умолчанию так и есть) и из таблицы была удалена большая часть данных, или же производилось множество действий над таблицей, которая содержит строки переменной длины (таблицы, которые содержат поля типов VARCHAR, BLOB или TEXT), то нужно дефрагментировать файл (таблицу), или другими словами выполнить команду "оптимизировать".

Нужно попытаться сделать это, если сервер mysqld использует много процессорного времени. Оптимизация таблиц ticket_history и article (см Сценарий ниже).

```
shell$ mysql -u user -p database
mysql$ optimize table ticket;
mysql$ optimize table ticket_history;
mysql$ optimize table article;
```

Сценарий: Оптимизация таблиц базы данных.

2.2. PostgreSQL

PostgreSQL лучше настраивается путем изменений в файле postgresql.conf который находится в директории с установленным PostgreSQL. За советом, как это сделать, обратитесь к следующим статьям:

- <http://www.revsys.com/writings/postgresql-performance.html>
- <http://varlena.com/GeneralBits/Tidbits/perf.html>
- http://varlena.com/GeneralBits/Tidbits/annotated_conf_e.html

Если производительность остается на неудовлетворительном уровне, мы предлагаем вам присоединиться к спискам рассылки "PostgreSQL Performance" (<http://www.postgresql.org/community/lists/>) и задавать вопросы именно там. Люди в этом списке рассылке очень дружелюбны а всегда постараются вам помочь.

3. Веб-сервер

Конечно же нужно использовать mod_perl 2.0 (<http://perl.apache.org/>). Он быстрее (~ * 100) чем "чистый" cgi. Но он требует больше оперативной памяти (RAM).

3.1. Предустановленное соединение с базой данных

При запуске веб-сервера можно иметь заранее установленное соединение с базой данных. Такой подход существенно экономит время (см. README.webserver).

3.2. Предварительно загруженные модули - startup.pl

Чтобы все работало быстрее и занимало меньше памяти используйте сценарий запуска `scripts/apache2-perl-startup.pl` для предварительной загрузки/компиляции Perl модулей на сервере с `mod_perl` (см. `README.webserver`).

3.3. Перезагрузка Perl-модулей во время обновления с диска

По умолчанию `Apache::Reload` используется в `scripts/apache2-httpd.include.conf`. Отключив его вы получите больше 8% прироста скорости. Не забудьте перезапустить веб-сервер, если устанавливаете модули через OTRS Package Manager, или устанавливаете любые значения в `SysConfig` или в `Kernel/Config.pm`. Важно: это также будет означать, что у вас не будет возможности использовать OTRS Package Manager через пользовательский веб-интерфейс. В таком случае необходимо воспользоваться командной строкой - `bin/otrs.PackageManager.pl`.

3.4. Выбор Правильной Стратегии

Если у вас действительно большая система, более 1 000 новых заявок в день и больше 40 агентов, то советуем почитать статью "Производительность" в Руководстве пользователя по `mod_perl` (<http://perl.apache.org/docs/2.0/user/index.html>).

3.5. mod_gzip/mod_deflate

Используйте `mod_deflate` для Apache2, если у вас "слабая" полоса пропускания. При загрузке html-страницы размером в 45к, `mod_gzip/mod_deflate` сожмет ее и страница будет занимать около 7к. Недостатком такого подхода является дополнительная нагрузка на сервер.

Приложение А.

Дополнительные ресурсы

Мы стараемся снабжать вас самой последней информацией об OTRS и предоставляем вам возможность оставить нам свой отзыв.

1. OTRS Group вебсайт

Более подробную информацию о компании OTRS Group можно найти на сайте по ссылке <http://www.otrs.com>. Он содержит много информации о продукте - белые листы/white papers, информацию о новых релизах, учебных курсов и многое другое.

2. Списки рассылки

В Таблице А-1 приведен полный список рассылок для сообщества.

Таблица А.1. Списки рассылки

Название	Описание	Домашняя страница
announce@otrs.org	Низкий трафик, на английском языке, для объявлений о новых релизах OTRS и решениях о безопасности.	http://lists.otrs.org/cgi-bin/listinfo/announce
otrs@otrs.org	От среднего до высокого объема трафика, на английском языке, где вы можете найти все виды вопросов, касающихся поддержки и продукта.	http://lists.otrs.org/cgi-bin/listinfo/otrs
otrs-de@otrs.org	От среднего до высокого объема трафика, на немецком языке, где вы можете найти все виды вопросов, касающихся поддержки и продукта.	http://lists.otrs.org/cgi-bin/listinfo/otrs-de
dev@otrs.org	Средний трафик на английском языке, где OTRS-разработчики обсуждают различные планы и вопросы реализации.	http://lists.otrs.org/cgi-bin/listinfo/dev
i18n@otrs.org	Небольшое количество пользователей на английском языке с вопросами интернационализации и локализации. Если вы есть или только хотите стать переводчиком проекта OTRS или имеете проблемы с одним из наших приложений, это именно то место, куда вам нужно.	http://lists.otrs.org/cgi-bin/listinfo/i18n

Чтобы подписаться на любой из этих списков рассылки перейдите по ссылке: <http://lists.otrs.org/>.

3. Трекинг ошибок

Чтобы сообщить об ошибке перейдите по ссылке <http://bugs.otrs.org/> (см. Рисунок ниже). Пожалуйста, обратите внимание на разницу между ошибкой и вопросом о конфигурации. Вопросы о конфигурации это проблемы, с которыми вы сталкиваетесь при создании и настройке системы, или же это общие вопросы по использованию OTRS. Отчет об ошибке должен отправляться только для найденных ошибок которые относятся к исходному коду OTRS или относятся к проблемам усовершенствования OTRS. Все отчеты об ошибках будут приветствоваться.

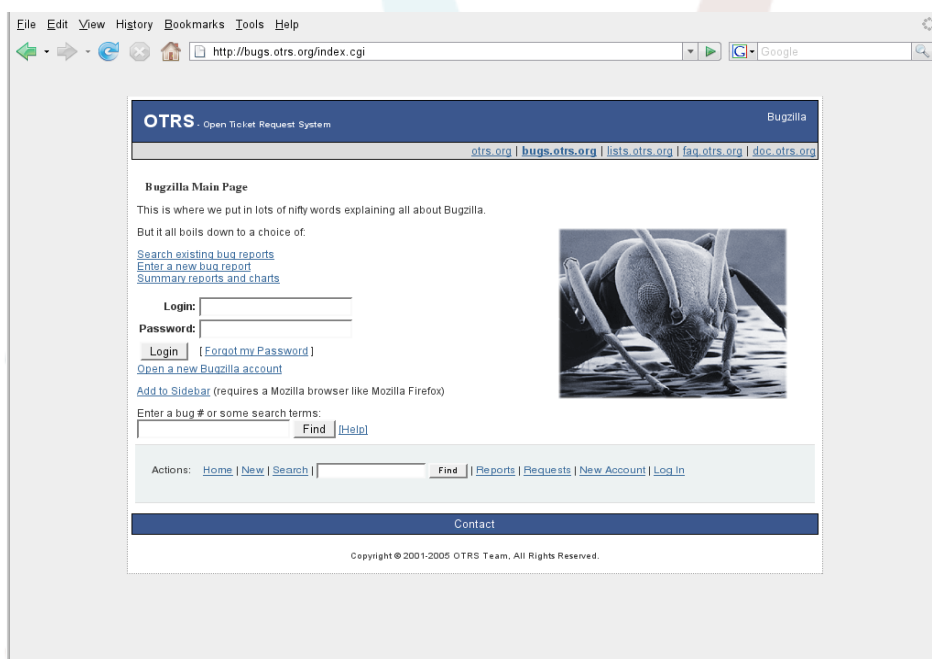


Рисунок: Трекинг ошибок.

По вопросам конфигурации используйте ссылку [коммерческая поддержка, которая доступна на сайте OTRS.com](#), или же публичные списки рассылки и форумы пользователей.

Вы можете помочь нам улучшить продукт если сообщите об ошибке. Мы ценим ваш вклад!

4. Коммерческая поддержка

Для услуг (поддержка, консалтинг, разработка и обучение) вы можете связаться с компанией разработчиком OTRS, OTRS AG. У нас есть офисы в Германии, США, Мексике, Нидерландах и других странах. Посетите наш сайт для поиска контактов: <http://www.otrs.com/en/corporate-navigation/contact/>

Приложение В. Configuration Options Reference

1. DynamicFields

1.1. DynamicFields::Driver::Registration

1.1.1. DynamicFields::Driver###Text

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'Text'} = { 'ConfigDialog' => 'AdminDynamicFieldText', 'DisplayName' => 'Text', 'Module' => 'Kernel::System::DynamicField::Driver::Text' };</pre>

1.1.2. DynamicFields::Driver###TextArea

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'TextArea'} = { 'ConfigDialog' => 'AdminDynamicFieldText', 'DisplayName' => 'Textarea', 'Module' => 'Kernel::System::DynamicField::Driver::TextArea' };</pre>

1.1.3. DynamicFields::Driver###Checkbox

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'Checkbox'} = { 'ConfigDialog' => 'AdminDynamicFieldCheckbox', 'DisplayName' => 'Checkbox', 'Module' => 'Kernel::System::DynamicField::Driver::Checkbox' };</pre>

1.1.4. DynamicFields::Driver###Dropdown

Description:	DynamicField backend registration.
Group:	DynamicFields

SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'Dropdown'} = { 'ConfigDialog' => 'AdminDynamicFieldDropdown', 'DisplayName' => 'Dropdown', 'Module' => 'Kernel::System::DynamicField::Driver::Dropdown' };</pre>

1.1.5. DynamicFields::Driver###DateTime

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'DateTime'} = { 'ConfigDialog' => 'AdminDynamicFieldDateTime', 'DisplayName' => 'Date / Time', 'Module' => 'Kernel::System::DynamicField::Driver::DateTime' };</pre>

1.1.6. DynamicFields::Driver###Date

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'Date'} = { 'ConfigDialog' => 'AdminDynamicFieldDateTime', 'DisplayName' => 'Date', 'Module' => 'Kernel::System::DynamicField::Driver::Date' };</pre>

1.1.7. DynamicFields::Driver###Multiselect

Description:	DynamicField backend registration.
Group:	DynamicFields
SubGroup:	DynamicFields::Driver::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::Driver'}->{'Multiselect'} = { 'ConfigDialog' => 'AdminDynamicFieldMultiselect', 'DisplayName' => 'Multiselect', 'ItemSeparator' => ', ', 'Module' => 'Kernel::System::DynamicField::Driver::Multiselect' };</pre>

1.2. DynamicFields::ObjectType::Registration

1.2.1. DynamicFields::ObjectType###Ticket

Description:	DynamicField object registration.
--------------	-----------------------------------

Group:	DynamicFields
SubGroup:	DynamicFields::ObjectType::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::ObjectType'}->{'Ticket'} = { 'DisplayName' => 'Ticket', 'Module' => 'Kernel::System::DynamicField::ObjectType::Ticket' };</pre>

1.2.2. DynamicFields::ObjectType###Article

Description:	DynamicField object registration.
Group:	DynamicFields
SubGroup:	DynamicFields::ObjectType::Registration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFields::ObjectType'}->{'Article'} = { 'DisplayName' => 'Article', 'Module' => 'Kernel::System::DynamicField::ObjectType::Article' };</pre>

1.3. Frontend::Admin::ModuleRegistration

1.3.1. Frontend::Module###AdminDynamicField

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminDynamicField'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.DynamicField.css'], 'JavaScript' => ['Core.Agent.Admin.DynamicField.js'] }, 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage dynamic fields.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Dynamic Fields', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Dynamic Fields GUI' };</pre>

1.3.2. Frontend::Module###AdminDynamicFieldText

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields

SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminDynamicFieldText'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'JavaScript' => ['Core.Agent.Admin.DynamicField.js'] }, 'Title' => 'Dynamic Fields Text Backend GUI' }; </pre>

1.3.3. Frontend::Module###AdminDynamicFieldCheckbox

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminDynamicFieldCheckbox'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'JavaScript' => ['Core.Agent.Admin.DynamicField.js'] }, 'Title' => 'Dynamic Fields Checkbox Backend GUI' }; </pre>

1.3.4. Frontend::Module###AdminDynamicFieldDropdown

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminDynamicFieldDropdown'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.DynamicField.css'], 'JavaScript' => ['Core.Agent.Admin.DynamicField.js', 'Core.Agent.Admin.DynamicFieldDropdown.js'] }, }; </pre>

```
'Title' => 'Dynamic Fields Drop-down Backend GUI'
};
```

1.3.5. Frontend::Module###AdminDynamicFieldDateTime

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminDynamicFieldDateTime'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.DynamicField.css'], 'JavaScript' => ['Core.Agent.Admin.DynamicField.js', 'Core.Agent.Admin.DynamicFieldDateTime.js'] }, 'Title' => 'Dynamic Fields Date Time Backend GUI' };</pre>

1.3.6. Frontend::Module###AdminDynamicFieldMultiselect

Description:	Frontend module registration for the agent interface.
Group:	DynamicFields
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminDynamicFieldMultiselect'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.DynamicField.css'], 'JavaScript' => ['Core.Agent.Admin.DynamicField.js', 'Core.Agent.Admin.DynamicFieldMultiselect.js'] }, 'Title' => 'Dynamic Fields Multiselect Backend GUI' };</pre>

1.4. Frontend::Agent::Preferences

1.4.1. PreferencesGroups###DynamicField

Description:	Defines the config parameters of this item, to be shown in the preferences view.
--------------	--

Group:	DynamicFields
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'DynamicField'} = { 'Active' => '1', 'Block' => 'Input', 'Column' => 'Other Settings', 'Data' => '\$Env{"UserDynamicField_NameX"}', 'Key' => 'Default value for NameX', 'Label' => 'NameX', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserDynamicField_NameX', 'Prio' => '7000' }; </pre>

1.4.2. PreferencesGroups###DynamicFieldsOverviewPageShown

Description:	Parameters for the pages (in which the dynamic fields are shown) of the dynamic fields overview.
Group:	DynamicFields
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'DynamicFieldsOverviewPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '25', 'Key' => 'Dynamic fields limit per page for Dynamic Fields Overview', 'Label' => 'Dynamic Fields Overview Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'AdminDynamicFieldsOverviewPageShown', 'Prio' => '8000' }; </pre>

2. Framework

2.1. Core

2.1.1. SecureMode

Description:	Disables the web installer (http://yourhost.example.com/otrs/installer.pl), to prevent the system from being hijacked. If set to "No", the system can be reinstalled and the current basic configuration will be used to pre-populate the questions within the installer script. If not active, it also disables the GenericAgent, PackageManager and SQL Box.
--------------	--

Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SecureMode'} = '0';</code>

2.1.2. Frontend::DebugMode

Description:	Enables or disable the debug mode over frontend interface.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::DebugMode'} = '0';</code>

2.1.3. ConfigLevel

Description:	Sets the configuration level of the administrator. Depending on the config level, some sysconfig options will be not shown. The config levels are in in ascending order: Expert, Advanced, Beginner. The higher the config level is (e.g. Beginner is the highest), the less likely is it that the user can accidentally configure the system in a way that it is not usable any more.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ConfigLevel'} = '100';</code>

2.1.4. ProductName

Description:	Defines the name of the application, shown in the web interface, tabs and title bar of the web browser.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ProductName'} = 'OTRS';</code>

2.1.5. SystemID

Description:	Defines the system identifier. Every ticket number and http session string contains this ID. This ensures that only tickets which belong to your system will be processed as follow-ups (useful when communicating between two instances of OTRS).
Group:	Framework
SubGroup:	Core

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SystemID'} = '10';</code>

2.1.6. FQDN

Description:	Defines the fully qualified domain name of the system. This setting is used as a variable, OTRS_CONFIG_FQDN which is found in all forms of messaging used by the application, to build links to the tickets within your system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'FQDN'} = 'yourhost.example.com';</code>

2.1.7. SupportDataCollector::HTTPHostname

Description:	Defines the HTTP hostname for the support data collection with the public module 'PublicSupportDataCollector' (e.g. used from the OTRS Daemon).
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SupportDataCollector::HTTPHostname'} = '';</code>

2.1.8. HttpType

Description:	Defines the type of protocol, used by the web server, to serve the application. If https protocol will be used instead of plain http, it must be specified here. Since this has no affect on the web server's settings or behavior, it will not change the method of access to the application and, if it is wrong, it will not prevent you from logging into the application. This setting is only used as a variable, OTRS_CONFIG_HttpType which is found in all forms of messaging used by the application, to build links to the tickets within your system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'HttpType'} = 'http';</code>

2.1.9. ScriptAlias

Description:	Sets the prefix to the scripts folder on the server, as configured on the web server. This setting is used as a variable,
--------------	---

	OTRS_CONFIG_ScriptAlias which is found in all forms of messaging used by the application, to build links to the tickets within the system.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ScriptAlias'} = 'otrs/';</code>

2.1.10. AdminEmail

Description:	Defines the system administrator's email address. It will be displayed in the error screens of the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AdminEmail'} = 'admin@example.com';</code>

2.1.11. Organization

Description:	Company name which will be included in outgoing emails as an X-Header.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Organization'} = 'Example Company';</code>

2.1.12. DefaultLanguage

Description:	Defines the default front-end language. All the possible values are determined by the available language files on the system (see the next setting).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultLanguage'} = 'en';</code>

2.1.13. DefaultUsedLanguages

Description:	Defines all the languages that are available to the application. The Key/Content pair links the front-end display name to the appropriate language PM file. The "Key" value should be the base-name of the PM file (i.e. de.pm is the file, then de is the "Key" value). The "Content" value should be the display
--------------	--

	name for the front-end. Specify any own-defined language here (see the developer documentation http://doc.otrs.org/ for more information). Please remember to use the HTML equivalents for non-ASCII characters (i.e. for the German oe = o umlaut, it is necessary to use the ö symbol).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre> \$self->{'DefaultUsedLanguages'} = { 'ar_SA' => 'Arabic (Saudi Arabia)', 'bg' => 'Bulgarian (&#x0411;&#x044a;&#x043b;&#x0433;&#x0430;&#x0440;&#x0441;&#x043a;&#x0438;)', 'ca' => 'Catal&agrave;', 'cs' => 'Czech (&#x010c;esky)', 'da' => 'Dansk', 'de' => 'Deutsch', 'el' => 'Greek (&#x0395;&#x03bb;&#x03b7;&#x03bd;&#x03b9;&#x03ba;&#x03ac;)', 'en' => 'English (United States)', 'en_CA' => 'English (Canada)', 'en_GB' => 'English (United Kingdom)', 'es' => 'Espa&ntilde;ol', 'es_CO' => 'Espa&ntilde;ol (Colombia)', 'es_MX' => 'Espa&ntilde;ol (M&eacute;xico)', 'et' => 'Eesti', 'fa' => 'Persian (&#x0641;&#x0627;&#x0631;&#x0633;&#x0649;)', 'fi' => 'Suomi', 'fr' => 'Fran&ccedil;ais', 'fr_CA' => 'Fran&ccedil;ais (Canada)', 'he' => 'Hebrew (#####)', 'hi' => 'Hindi', 'hr' => 'Hrvatski', 'hu' => 'Magyar', 'it' => 'Italiano', 'ja' => 'Japanese (&#x65e5;&#x672c;&#x8a9e)', 'lt' => 'Lietuvių kalba', 'lv' => 'Latvijas', 'ms' => 'Malay', 'nb_NO' => 'Norsk bokm&aring;l', 'nl' => 'Nederlands', 'pl' => 'Polski', 'pt' => 'Portugu&ecirc;s', 'pt_BR' => 'Portugu&ecirc;s Brasileiro', 'ru' => 'Russian (&#x0420;&#x0443;&#x0441;&#x044a;&#x0438;&#x0439;)', 'sk_SK' => 'Slovak (Sloven&#x010d;ina)', 'sl' => 'Slovenian (Sloven&#x0161;ina)', 'sr_Cyrl' => 'Serbian Cyrillic (srpski)', 'sr_Latn' => 'Serbian Latin (Srpski)', 'sv' => 'Svenska', 'tr' => 'T&uuml;rks&ccedil;e', 'uk' => 'Ukrainian (&#x0423;&#x0430;&#x0440;&#x0430;&#x0457;&#x043d;&#x0441;&#x044c;&#x0430;&#x0430;)', 'vi_VN' => 'Vietnam (Vi&#x0246;t Nam)', 'zh_CN' => 'Chinese (Sim.) (&#x7b80;&#x4f53;&#x4e2d;&#x6587;)', 'zh_TW' => 'Chinese (Tradi.) (&#x6b63;&#x9ad4;&#x4e2d;&#x6587;)' }; </pre>

2.1.14. DefaultTheme

Description:	Defines the default front-end (HTML) theme to be used by the agents and customers. If you like, you can add your own theme. Please refer the administrator manual located at http://doc.otrs.org/ .
--------------	--

Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultTheme'} = 'Standard';</code>

2.1.15. DefaultTheme::HostBased

Description:	It is possible to configure different themes, for example to distinguish between agents and customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid theme on your system. Please see the example entries for the proper form of the regex.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'DefaultTheme::HostBased'} = { 'host1\\.example\\.com' => 'SomeTheme1', 'host2\\.example\\.com' => 'SomeTheme2' };</code>

2.1.16. CheckMXRecord

Description:	Makes the application check the MX record of email addresses before sending an email or submitting a telephone or email ticket.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckMXRecord'} = '1';</code>

2.1.17. CheckMXRecord::Nameserver

Description:	Defines the address of a dedicated DNS server, if necessary, for the "CheckMXRecord" look-ups.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CheckMXRecord::Nameserver'} = 'ns.example.com';</code>

2.1.18. CheckEmailAddresses

Description:	Makes the application check the syntax of email addresses.
Group:	Framework

SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailAddresses'} = '1';</code>

2.1.19. CheckEmailValidAddress

Description:	Defines a regular expression that excludes some addresses from the syntax check (if "CheckEmailAddresses" is set to "Yes"). Please enter a regex in this field for email addresses, that aren't syntactically valid, but are necessary for the system (i.e. "root@localhost").
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailValidAddress'} = '^(root@localhost admin@localhost)\$';</code>

2.1.20. CheckEmailInvalidAddress

Description:	Defines a regular expression that filters all email addresses that should not be used in the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CheckEmailInvalidAddress'} = '@(example)\.\.(...)\$';</code>

2.1.21. CGILogPrefix

Description:	Specifies the text that should appear in the log file to denote a CGI script entry.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CGILogPrefix'} = 'OTRS-CGI';</code>

2.1.22. DemoSystem

Description:	Runs the system in "Demo" mode. If set to "Yes", agents can change preferences, such as selection of language and theme via the agent web interface. These changes are only valid for the current session. It will not be possible for agents to change their passwords.
Group:	Framework
SubGroup:	Core

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DemoSystem'} = '0';</code>

2.1.23. SwitchToUser

Description:	Allows the administrators to login as other users, via the users administration panel.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SwitchToUser'} = '0';</code>

2.1.24. SwitchToCustomer

Description:	Allows the administrators to login as other customers, via the customer user administration panel.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SwitchToCustomer'} = '0';</code>

2.1.25. SwitchToCustomer::PermissionGroup

Description:	Specifies the group where the user needs rw permissions so that he can access the "SwitchToCustomer" feature.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SwitchToCustomer::PermissionGroup'} = 'admin';</code>

2.1.26. NotificationSenderName

Description:	Specifies the name that should be used by the application when sending notifications. The sender name is used to build the complete display name for the notification master (i.e. "OTRS Notification Master" otrs@your.example.com). Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSenderName'} = 'OTRS Notification Master';</code>

2.1.27. NotificationSenderEmail

Description:	Specifies the email address that should be used by the application when sending notifications. The email address is used to build the complete display name for the notification master (i.e. "OTRS Notification Master" otrs@your.example.com). You can use the OTRS_CONFIG_FQDN variable as set in your configuration, or choose another email address. Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSenderEmail'} = 'otrs@<OTRS_CONFIG_FQDN>';</code>

2.1.28. System::Customer::Permission

Description:	Defines the standard permissions available for customers within the application. If more permissions are needed, you can enter them here. Permissions must be hard coded to be effective. Please ensure, when adding any of the afore mentioned permissions, that the "rw" permission remains the last entry.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'System::Customer::Permission'} = ['ro', 'rw'];</code>

2.1.29. LanguageDebug

Description:	Debugs the translation set. If this is set to "Yes" all strings (text) without translations are written to STDERR. This can be helpful when you are creating a new translation file. Otherwise, this option should remain set to "No".
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'LanguageDebug'} = '0';</code>

2.1.30. Secure::DisableBanner

Description:	If enabled, the OTRS version tag will be removed from the Webinterface, the HTTP headers and the X-Headers of outgoing mails.
Group:	Framework
SubGroup:	Core

Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Secure::DisableBanner'} = '0';</code>

2.1.31. StandardTemplate2QueueByCreating

Description:	List of default Standard Templates which are assigned automatically to new Queues upon creation.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'StandardTemplate2QueueByCreating'} = [</code> <code> ,</code> <code>];</code>

2.2. Core::Cache

2.2.1. Cache::Module

Description:	Selects the cache backend to use.
Group:	Framework
SubGroup:	Core::Cache
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Cache::Module'} = 'Kernel::System::Cache::FileStorable';</code>

2.2.2. Cache::SubdirLevels

Description:	Specify how many sub directory levels to use when creating cache files. This should prevent too many cache files being in one directory.
Group:	Framework
SubGroup:	Core::Cache
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Cache::SubdirLevels'} = '2';</code>

2.3. Core::CustomerCompany

2.3.1. CustomerCompany::EventModulePost###100-UpdateCustomerUsers

Description:	Event module that updates customer users after an update of the Customer Company.
Group:	Framework
SubGroup:	Core::CustomerCompany
Valid:	1

Required:	0
Config-Setting:	<pre>\$Self->{'CustomerCompany::EventModulePost'}->{'100-UpdateCustomerUsers'} = { 'Event' => 'CustomerCompanyUpdate', 'Module' => 'Kernel::System::CustomerCompany::Event::CustomerUserUpdate', 'Transaction' => '0' };</pre>

2.4. Core::CustomerUser

2.4.1. CustomerUser::EventModulePost###100-UpdateServiceMembership

Description:	Event module that updates customer user service membership if login changes.
Group:	Framework
SubGroup:	Core::CustomerUser
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerUser::EventModulePost'}->{'100-UpdateServiceMembership'} = { 'Event' => 'CustomerUserUpdate', 'Module' => 'Kernel::System::CustomerUser::Event::ServiceMemberUpdate', 'Transaction' => '0' };</pre>

2.5. Core::LinkObject

2.5.1. LinkObject::ViewMode

Description:	Determines the way the linked objects are displayed in each zoom mask.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::ViewMode'} = 'Simple';</pre>

2.5.2. LinkObject::Type###Normal

Description:	Defines the link type 'Normal'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::Type'}->{'Normal'} = { 'SourceName' => 'Normal', 'TargetName' => 'Normal' };</pre>

2.5.3. LinkObject::Type###ParentChild

Description:	Defines the link type 'ParentChild'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::Type'}->{'ParentChild'} = { 'SourceName' => 'Parent', 'TargetName' => 'Child' };</pre>

2.5.4. LinkObject::TypeGroup###0001

Description:	Defines the link type groups. The link types of the same group cancel one another. Example: If ticket A is linked per a 'Normal' link with ticket B, then these tickets could not be additionally linked with link of a 'ParentChild' relationship.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::TypeGroup'}->{'0001'} = ['Normal', 'ParentChild'];</pre>

2.6. Core::Log

2.6.1. LogModule

Description:	Defines the log module for the system. "File" writes all messages in a given logfile, "SysLog" uses the syslog daemon of the system, e.g. syslogd.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LogModule'} = 'Kernel::System::Log::SysLog';</pre>

2.6.2. LogModule::SysLog::Facility

Description:	If "SysLog" was selected for LogModule, a special log facility can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'LogModule::SysLog::Facility'} = 'user';</code>
-----------------	---

2.6.3. LogModule::SysLog::LogSock

Description:	If "SysLog" was selected for LogModule, a special log sock can be specified (on solaris you may need to use 'stream').
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::LogSock'} = 'unix';</code>

2.6.4. LogModule::SysLog::Charset

Description:	If "SysLog" was selected for LogModule, the charset that should be used for logging can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::Charset'} = 'utf-8';</code>

2.6.5. LogModule::LogFile

Description:	If "file" was selected for LogModule, a logfile must be specified. If the file doesn't exist, it will be created by the system.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::LogFile'} = '/tmp/otrs.log';</code>

2.6.6. LogModule::LogFile::Date

Description:	Adds a suffix with the actual year and month to the OTRS log file. A logfile for every month will be created.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::LogFile::Date'} = '0';</code>

2.7. Core::MIME-Viewer

2.7.1. MIME-Viewer###application/excel

Description:	Specifies the path to the converter that allows the view of Microsoft Excel files, in the web interface.
--------------	--

Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/excel'} = 'xlhtml';</code>

2.7.2. MIME-Viewer###application/msword

Description:	Specifies the path to the converter that allows the view of Microsoft Word files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/msword'} = 'wvWare';</code>

2.7.3. MIME-Viewer###application/pdf

Description:	Specifies the path to the converter that allows the view of PDF documents, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/pdf'} = 'pdftohtml -stdout -i';</code>

2.7.4. MIME-Viewer###text/xml

Description:	Specifies the path to the converter that allows the view of XML files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'text/xml'} = '<OTRS_CONFIG_Home>/scripts/tools/xml2html.pl';</code>

2.8. Core::MirrorDB

2.8.1. Core::MirrorDB::DSN

Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, specify the DSN to this database.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0

Required:	0
Config-Setting:	<code>\$Self->{'Core::MirrorDB::DSN'} = 'DBI:mysql:database=mirrordb;host=mirrordbhost';</code>

2.8.2. Core::MirrorDB::User

Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the user to authenticate to this database can be specified.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Core::MirrorDB::User'} = 'some_user';</code>

2.8.3. Core::MirrorDB::Password

Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the password to authenticate to this database can be specified.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Core::MirrorDB::Password'} = 'some_password';</code>

2.9. Core::PDF

2.9.1. PDF

Description:	Enables PDF output. The CPAN module PDF::API2 is required, if not installed, PDF output will be disabled.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF'} = '1';</code>

2.9.2. PDF::LogoFile

Description:	Specifies the path of the file for the logo in the page header (gif jpg png, 700 x 100 pixel).
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::LogoFile'} = '<OTRS_CONFIG_Home>/var/logo-otrs.png';</code>

2.9.3. PDF::PageSize

Description:	Defines the standard size of PDF pages.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::PageSize'} = 'a4';</code>

2.9.4. PDF::MaxPages

Description:	Defines the maximum number of pages per PDF file.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::MaxPages'} = '100';</code>

2.9.5. PDF::TTFontFile###Proportional

Description:	Defines the path and TTF-File to handle proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'Proportional'} = 'DejaVuSans.ttf';</code>

2.9.6. PDF::TTFontFile###ProportionalBold

Description:	Defines the path and TTF-File to handle bold proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalBold'} = 'DejaVuSans-Bold.ttf';</code>

2.9.7. PDF::TTFontFile###ProportionalItalic

Description:	Defines the path and TTF-File to handle italic proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalItalic'} = 'DejaVuSans-Oblique.ttf';</code>
-----------------	--

2.9.8. PDF::TTFontFile###ProportionalBoldItalic

Description:	Defines the path and TTF-File to handle bold italic proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalBoldItalic'} = 'DejaVuSans-BoldOblique.ttf';</code>

2.9.9. PDF::TTFontFile###Monospaced

Description:	Defines the path and TTF-File to handle monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'Monospaced'} = 'DejaVuSansMono.ttf';</code>

2.9.10. PDF::TTFontFile###MonospacedBold

Description:	Defines the path and TTF-File to handle bold monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedBold'} = 'DejaVuSansMono-Bold.ttf';</code>

2.9.11. PDF::TTFontFile###MonospacedItalic

Description:	Defines the path and TTF-File to handle italic monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedItalic'} = 'DejaVuSansMono-Oblique.ttf';</code>

2.9.12. PDF::TTFontFile###MonospacedBoldItalic

Description:	Defines the path and TTF-File to handle bold italic monospaced font in PDF documents.
--------------	---

Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'MonospacedBoldItalic'} = 'DejaVuSansMono-BoldOblique.ttf';</code>

2.10. Core::Package

2.10.1. Package::FileUpload

Description:	Enables file upload in the package manager frontend.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Package::FileUpload'} = '1';</code>

2.10.2. Package::RepositoryRoot

Description:	Defines the location to get online repository list for additional packages. The first available result will be used.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryRoot'} = ['http://ftp.otrs.org/pub/otrs/misc/packages/repository.xml'];</code>

2.10.3. Package::RepositoryList

Description:	Defines the list of online repositories. Another installations can be used as repository, for example: Key="http://example.com/otrs/public.pl?Action=PublicRepository;File=" and Content="Some Name".
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryList'} = { 'ftp://ftp.example.com/pub/otrs/misc/packages/' => '[Example] ftp://ftp.example.com/' };</code>

2.10.4. Package::RepositoryAccessRegExp

Description:	Defines the IP regular expression for accessing the local repository. You need to enable this to have access to your local repository and the package::RepositoryList is required on the remote host.
--------------	---

Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Package::RepositoryAccessRegExp'} = '127\\.0\\.0\\.1';</code>

2.10.5. Package::Timeout

Description:	Sets the timeout (in seconds) for package downloads. Overwrites "WebUserAgent::Timeout".
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Package::Timeout'} = '15';</code>

2.10.6. Package::Proxy

Description:	Fetches packages via proxy. Overwrites "WebUserAgent::Proxy".
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Package::Proxy'} = 'http://proxy.sn.no:8001/';</code>

2.10.7. Package::ShowFeatureAddons

Description:	Toggles display of OTRS FeatureAddons list in PackageManager.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Package::ShowFeatureAddons'} = '1';</code>

2.10.8. Package::EventModule###99-SupportDataSend

Description:	Package event module file a scheduler task for update registration.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Package::EventModule'}->{'99-SupportDataSend'} = { 'Event' => '(PackageInstall PackageReinstall PackageUpgrade PackageUninstall)', 'Module' => 'Kernel::System::Package::Event::SupportDataSend', 'Transaction' => '1'</code>

```
};
```

2.11. Core::PerformanceLog

2.11.1. PerformanceLog

Description:	Enables performance log (to log the page response time). It will affect the system performance. Frontend::Module###AdminPerformanceLog must be enabled.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'PerformanceLog'} = '0';</code>

2.11.2. PerformanceLog::File

Description:	Specifies the path of the file for the performance log.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PerformanceLog::File'} = '<OTRS_CONFIG_Home>/var/log/Performance.log';</code>

2.11.3. PerformanceLog::FileMax

Description:	Defines the maximum size (in MB) of the log file.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PerformanceLog::FileMax'} = '25';</code>

2.12. Core::ReferenceData

2.12.1. ReferenceData::OwnCountryList

Description:	This setting allows you to override the built-in country list with your own list of countries. This is particularly handy if you just want to use a small select group of countries.
Group:	Framework
SubGroup:	Core::ReferenceData
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'ReferenceData::OwnCountryList'} = { 'AT' => 'Austria', 'CH' => 'Switzerland', 'DE' => 'Germany'</code>

```
};
```

2.13. Core::SOAP

2.13.1. SOAP::User

Description:	Defines the username to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::User'} = 'some_user';</code>

2.13.2. SOAP::Password

Description:	Defines the password to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::Password'} = 'some_pass';</code>

2.13.3. SOAP::Keep-Alive

Description:	Enable keep-alive connection header for SOAP responses.
Group:	Framework
SubGroup:	Core::SOAP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SOAP::Keep-Alive'} = '0';</code>

2.14. Core::Sendmail

2.14.1. SendmailModule

Description:	Defines the module to send emails. "Sendmail" directly uses the sendmail binary of your operating system. Any of the "SMTP" mechanisms use a specified (external) mailserver. "DoNotSendEmail" doesn't send emails and it is useful for test systems.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule'} = 'Kernel::System::Email::Sendmail';</code>

2.14.2. SendmailModule::CMD

Description:	If "Sendmail" was selected as SendmailModule, the location of the sendmail binary and the needed options must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::CMD'} = '/usr/sbin/sendmail -i -f';</code>

2.14.3. SendmailModule::Host

Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the mailhost that sends out the mails must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::Host'} = 'mail.example.com';</code>

2.14.4. SendmailModule::Port

Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the port where your mailserver is listening for incoming connections must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::Port'} = '25';</code>

2.14.5. SendmailModule::AuthUser

Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, an username must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthUser'} = 'MailserverLogin';</code>

2.14.6. SendmailModule::AuthPassword

Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, a password must be specified.
--------------	--

Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthPassword'} = 'MailserverPassword';</code>

2.14.7. SendmailBcc

Description:	Sends all outgoing email via bcc to the specified address. Please use this only for backup reasons.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'SendmailBcc'} = '';</code>

2.14.8. SendmailEnvelopeFrom

Description:	If set, this address is used as envelope sender in outgoing messages (not notifications - see below). If no address is specified, the envelope sender is equal to queue e-mail address.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailEnvelopeFrom'} = '';</code>

2.14.9. SendmailNotificationEnvelopeFrom

Description:	If set, this address is used as envelope sender header in outgoing notifications. If no address is specified, the envelope sender header is empty.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailNotificationEnvelopeFrom'} = '';</code>

2.14.10. SendmailEncodingForce

Description:	Forces encoding of outgoing emails (7bit 8bit quoted-printable base64).
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailEncodingForce'} = 'base64';</code>

2.15. Core::Session

2.15.1. SessionModule

Description:	Defines the module used to store the session data. With "DB" the frontend server can be splitted from the db server. "FS" is faster.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionModule'} = 'Kernel::System::AuthSession::DB';</code>

2.15.2. SessionName

Description:	Defines the name of the session key. E.g. Session, SessionID or OTRS.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionName'} = 'OTRSAgentInterface';</code>

2.15.3. CustomerPanelSessionName

Description:	Defines the name of the key for customer sessions.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSessionName'} = 'OTRSCustomerInterface';</code>

2.15.4. SessionCheckRemoteIP

Description:	Turns on the remote ip address check. It should be set to "No" if the application is used, for example, via a proxy farm or a dialup connection, because the remote ip address is mostly different for the requests.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCheckRemoteIP'} = '1';</code>

2.15.5. SessionDeleteIfNotRemoteID

Description:	Deletes a session if the session id is used with an invalid remote IP address.
Group:	Framework

SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDeleteIfNotRemoteID'} = '1';</code>

2.15.6. SessionMaxTime

Description:	Defines the maximal valid time (in seconds) for a session id.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxTime'} = '57600';</code>

2.15.7. SessionMaxIdleTime

Description:	Sets the inactivity time (in seconds) to pass before a session is killed and a user is logged out.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxIdleTime'} = '21600';</code>

2.15.8. SessionActiveTime

Description:	Sets the time (in seconds) a user is marked as active.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionActiveTime'} = '600';</code>

2.15.9. SessionDeleteIfTimeToOld

Description:	Deletes requested sessions if they have timed out.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDeleteIfTimeToOld'} = '1';</code>

2.15.10. SessionUseCookie

Description:	Makes the session management use html cookies. If html cookies are disabled or if the client browser disabled html cookies, then the system will work as usual and append the session id to the links.
--------------	--

Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookie'} = '1';</code>

2.15.11. SessionUseCookieAfterBrowserClose

Description:	Stores cookies after the browser has been closed.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookieAfterBrowserClose'} = '0';</code>

2.15.12. SessionCSRFProtection

Description:	Protection against CSRF (Cross Site Request Forgery) exploits (for more info see http://en.wikipedia.org/wiki/Cross-site_request_forgery).
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCSRFProtection'} = '1';</code>

2.15.13. AgentSessionLimit

Description:	Sets the maximum number of active agents within the timespan defined in SessionActiveTime.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'AgentSessionLimit'} = '100';</code>

2.15.14. CustomerSessionLimit

Description:	Sets the maximum number of active customers within the timespan defined in SessionActiveTime.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'CustomerSessionLimit'} = '100';</code>

2.15.15. SessionDir

Description:	If "FS" was selected for SessionModule, a directory where the session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDir'} = '<OTRS_CONFIG_Home>/var/sessions';</code>

2.15.16. SessionTable

Description:	If "DB" was selected for SessionModule, a table in database where session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTable'} = 'sessions';</code>

2.16. Core::SpellChecker

2.16.1. SpellChecker

Description:	Enables spell checker support.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellChecker'} = '0';</code>

2.16.2. SpellCheckerBin

Description:	Install ispell or aspell on the system, if you want to use a spell checker. Please specify the path to the aspell or ispell binary on your operating system.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellCheckerBin'} = '/usr/bin/ispell';</code>

2.16.3. SpellCheckerDictDefault

Description:	Defines the default spell checker dictionary.
Group:	Framework
SubGroup:	Core::SpellChecker

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellCheckerDictDefault'} = 'english';</code>

2.16.4. SpellCheckerIgnore

Description:	Defines a default list of words, that are ignored by the spell checker.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellCheckerIgnore'} = ['www', 'webmail', 'https', 'http', 'html', 'rfc'];</code>

2.17. Core::Stats

2.17.1. Stats::StatsHook

Description:	Sets the stats hook.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::StatsHook'} = 'Stat#';</code>

2.17.2. Stats::StatsStartNumber

Description:	Start number for statistics counting. Every new stat increments this number.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::StatsStartNumber'} = '10000';</code>

2.17.3. Stats::MaxXaxisAttributes

Description:	Defines the default maximum number of X-axis attributes for the time scale.
Group:	Framework
SubGroup:	Core::Stats
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Stats::MaxXaxisAttributes'} = '1000';</code>
-----------------	---

2.18. Core::Stats::Graph

2.18.1. Stats::Graph::t_margin

Description:	Specifies the top margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::t_margin'} = '10';</code>

2.18.2. Stats::Graph::l_margin

Description:	Specifies the left margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::l_margin'} = '10';</code>

2.18.3. Stats::Graph::b_margin

Description:	Specifies the bottom margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::b_margin'} = '10';</code>

2.18.4. Stats::Graph::r_margin

Description:	Specifies the right margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::r_margin'} = '20';</code>

2.18.5. Stats::Graph::bgclr

Description:	Specifies the background color of the picture.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Stats::Graph::bgclr'} = 'white';</code>
-----------------	--

2.18.6. Stats::Graph::transparent

Description:	Makes the picture transparent.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::transparent'} = '0';</code>

2.18.7. Stats::Graph::fgclr

Description:	Specifies the border color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::fgclr'} = 'black';</code>

2.18.8. Stats::Graph::boxclr

Description:	Specifies the background color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::boxclr'} = 'white';</code>

2.18.9. Stats::Graph::accentclr

Description:	Specifies the border color of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::accentclr'} = 'black';</code>

2.18.10. Stats::Graph::legendclr

Description:	Specifies the text color of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legendclr'} = 'black';</code>

2.18.11. Stats::Graph::textclr

Description:	Specifies the text color of the chart (e. g. caption).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::textclr'} = 'black';</code>

2.18.12. Stats::Graph::dclrs

Description:	Defines the colors for the graphs.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Stats::Graph::dclrs'} = ['red', 'green', 'blue', 'yellow', 'purple', 'orange', 'pink', 'marine', 'cyan', 'lgray', 'lblue', 'lyellow', 'lgreen', 'lred', 'lpurple', 'lorange', 'lbrown']; </pre>

2.18.13. Stats::Graph::TitleFont

Description:	Defines the title font in graphs (place custom fonts in var/fonts).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::TitleFont'} = 'DejaVuSans-Bold.ttf';</code>

2.18.14. Stats::Graph::LegendFont

Description:	Defines the legend font in graphs (place custom fonts in var/fonts).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::LegendFont'} = 'DejaVuSans.ttf';</code>

2.18.15. Stats::Graph::line_width

Description:	Defines the boldness of the line drawn by the graph.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::line_width'} = '1';</code>

2.18.16. Stats::Graph::legend_placement

Description:	Defines the placement of the legend. This should be a two letter key of the form: 'B[LCR]]R[TCB]'. The first letter indicates the placement (Bottom or Right), and the second letter the alignment (Left, Right, Center, Top, or Bottom).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_placement'} = 'BC';</code>

2.18.17. Stats::Graph::legend_spacing

Description:	Defines the spacing of the legends.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_spacing'} = '4';</code>

2.18.18. Stats::Graph::legend_marker_width

Description:	Defines the width of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_width'} = '12';</code>

2.18.19. Stats::Graph::legend_marker_height

Description:	Defines the height of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_height'} = '8';</code>
-----------------	---

2.19. Core::Time

2.19.1. TimeInputFormat

Description:	Defines the date input format used in forms (option or input fields).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeInputFormat'} = 'Option';</code>

2.19.2. TimeShowAlwaysLong

Description:	Shows time in long format (days, hours, minutes), if set to "Yes"; or in short format (days, hours), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeShowAlwaysLong'} = '0';</code>

2.19.3. TimeZone

Description:	Sets the system time zone (required a system with UTC as system time). Otherwise this is a diff time to the local time.
Group:	Framework
SubGroup:	Core::Time
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'TimeZone'} = '+0';</code>

2.19.4. TimeZoneUser

Description:	Sets the user time zone per user (required a system with UTC as system time and UTC under TimeZone). Otherwise this is a diff time to the local time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUser'} = '0';</code>

2.19.5. TimeZoneUserBrowserAutoOffset

Description:	Sets the user time zone per user based on java script / browser time zone offset feature at login time.
--------------	---

Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUserBrowserAutoOffset'} = '1';</code>

2.19.6. CalendarWeekDayStart

Description:	Define the start day of the week for the date picker.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CalendarWeekDayStart'} = '1';</code>

2.19.7. TimeVacationDays

Description:	Adds the permanent vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.19.8. TimeVacationDaysOneTime

Description:	Adds the one time vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

2.19.9. TimeWorkingHours

Description:	Defines the hours and week days to count the working time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre> \$self->{'TimeWorkingHours'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', </pre>

```

    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
];

```

2.19.10. TimeShowCompleteDescription

Description:	Shows time use complete description (days, hours, minutes), if set to "Yes"; or just first letter (d, h, m), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeShowCompleteDescription'} = '0';</code>

2.20. Core::Time::Calendar1

2.20.1. TimeZone::Calendar1Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar1Name'} = 'Calendar Name 1';</code>

2.20.2. TimeZone::Calendar1

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar1'} = '0';</code>

2.20.3. TimeVacationDays::Calendar1

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
--------------	--

Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar1'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } }</pre>

2.20.4. TimeVacationDaysOneTime::Calendar1

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar1'} = { '2004' => { '1' => { '1' => 'test' } } }</pre>

2.20.5. TimeWorkingHours::Calendar1

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar1'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], }</pre>


```

'Mon' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
],
'Sat' => [],
'Sun' => [],
'Thu' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
],
'Tue' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
],
'Wed' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
]
];

```

2.21. Core::Time::Calendar2

2.21.1. TimeZone::Calendar2Name

Description:	Defines the name of the indicated calendar.
--------------	---

Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar2Name'} = 'Calendar Name 2';</code>

2.21.2. TimeZone::Calendar2

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar2'} = '0';</code>

2.21.3. TimeVacationDays::Calendar2

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar2'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.21.4. TimeVacationDaysOneTime::Calendar2

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar2'} = { '2004' => { '1' => {</pre>

```
'1' => 'test'
}
}
};
```

2.21.5. TimeWorkingHours::Calendar2

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar2'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12',</pre>

```

    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
];

```

2.22. Core::Time::Calendar3

2.22.1. TimeZone::Calendar3Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3Name'} = 'Calendar Name 3';</code>

2.22.2. TimeZone::Calendar3

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3'} = '0';</code>

2.22.3. TimeVacationDays::Calendar3

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar3

Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar3'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.22.4. TimeVacationDaysOneTime::Calendar3

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar3'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

2.22.5. TimeWorkingHours::Calendar3

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar3'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8',</pre>

```

'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Sat' => [],
'Sun' => [],
'Thu' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Tue' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
]
};

```

2.23. Core::Time::Calendar4

2.23.1. TimeZone::Calendar4Name

Description:	Defines the name of the indicated calendar.
Group:	Framework

SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar4Name'} = 'Calendar Name 4';</code>

2.23.2. TimeZone::Calendar4

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar4'} = '0';</code>

2.23.3. TimeVacationDays::Calendar4

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar4'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.23.4. TimeVacationDaysOneTime::Calendar4

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar4'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

```

    }
  }
};

```

2.23.5. TimeWorkingHours::Calendar4

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar4'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', </pre>


```
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
};
```

2.24. Core::Time::Calendar5

2.24.1. TimeZone::Calendar5Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5Name'} = 'Calendar Name 5';</code>

2.24.2. TimeZone::Calendar5

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5'} = '0';</code>

2.24.3. TimeVacationDays::Calendar5

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1

Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar5'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.24.4. TimeVacationDaysOneTime::Calendar5

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar5'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

2.24.5. TimeWorkingHours::Calendar5

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar5'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10',</pre>

```

'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Sat' => [],
'Sun' => [],
'Thu' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Tue' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
];

```

2.25. Core::Time::Calendar6

2.25.1. TimeZone::Calendar6Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar6

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar6Name'} = 'Calendar Name 6';</code>

2.25.2. TimeZone::Calendar6

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar6'} = '0';</code>

2.25.3. TimeVacationDays::Calendar6

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar6'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.25.4. TimeVacationDaysOneTime::Calendar6

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar6'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

```
};
```

2.25.5. TimeWorkingHours::Calendar6

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar6'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15',</pre>

```

    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
];

```

2.26. Core::Time::Calendar7

2.26.1. TimeZone::Calendar7Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar7Name'} = 'Calendar Name 7';</code>

2.26.2. TimeZone::Calendar7

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar7'} = '0';</code>

2.26.3. TimeVacationDays::Calendar7

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1

Config-Setting:	<pre> \$Self->{'TimeVacationDays::Calendar7'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } } }; </pre>
-----------------	---

2.26.4. TimeVacationDaysOneTime::Calendar7

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime::Calendar7'} = { '2004' => { '1' => { '1' => 'test' } } } }; </pre>

2.26.5. TimeWorkingHours::Calendar7

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar7'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', </pre>

```

    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Sat' => [],
  'Sun' => [],
  'Thu' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Tue' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
];

```

2.27. Core::Time::Calendar8

2.27.1. TimeZone::Calendar8Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar8Name'} = 'Calendar Name 8';</code>

2.27.2. TimeZone::Calendar8

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar8'} = '0';</code>

2.27.3. TimeVacationDays::Calendar8

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDays::Calendar8'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

2.27.4. TimeVacationDaysOneTime::Calendar8

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar8'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

2.27.5. TimeWorkingHours::Calendar8

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre> \$self->{'TimeWorkingHours::Calendar8'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Sat' => [], 'Sun' => [], 'Thu' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Tue' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18',] } </pre>

```

    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
};

```

2.28. Core::Time::Calendar9

2.28.1. TimeZone::Calendar9Name

Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar9Name'} = 'Calendar Name 9';</code>

2.28.2. TimeZone::Calendar9

Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar9'} = '0';</code>

2.28.3. TimeVacationDays::Calendar9

Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeVacationDays::Calendar9'} = { '1' => { '1' => 'New Year\'s Day' }, '12' => { '24' => 'Christmas Eve', </code>

	<pre>'25' => 'First Christmas Day', '26' => 'Second Christmas Day', '31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>
--	--

2.28.4. TimeVacationDaysOneTime::Calendar9

Description:	Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar9'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

2.28.5. TimeWorkingHours::Calendar9

Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar9'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', </pre>

```

    '20'
  ],
  'Sat' => [],
  'Sun' => [],
  'Thu' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Tue' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
]
};

```

2.29. Core::Web

2.29.1. Frontend::WebPath

Description:	Defines the URL base path of icons, CSS and Java Script.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::WebPath'} = '/otrs-web/';</code>

2.29.2. Frontend::ImagePath

Description:	Defines the URL image path of icons for navigation.
--------------	---

Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::ImagePath'} = '<OTRS_CONFIG_Frontend::WebPath>skins/Agent/default/img/';</code>

2.29.3. Frontend::CSSPath

Description:	Defines the URL CSS path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CSSPath'} = '<OTRS_CONFIG_Frontend::WebPath>css/';</code>

2.29.4. Frontend::JavaScriptPath

Description:	Defines the URL java script path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::JavaScriptPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/';</code>

2.29.5. Frontend::RichText

Description:	Uses richtext for viewing and editing: articles, salutations, signatures, standard templates, auto responses and notifications.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText'} = '1';</code>

2.29.6. Frontend::RichTextPath

Description:	Defines the URL rich text editor path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/thirdparty/ckeditor-4.4.5/';</code>

2.29.7. Frontend::RichTextWidth

Description:	Defines the width for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextWidth'} = '620';</code>

2.29.8. Frontend::RichTextHeight

Description:	Defines the height for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextHeight'} = '320';</code>

2.29.9. Frontend::RichText::DefaultCSS

Description:	Defines the default CSS used in rich text editors.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText::DefaultCSS'} = 'font-family:Geneva,Helvetica,Arial,sans-serif; font-size: 12px;';</code>

2.29.10. Frontend::RichText::EnhancedMode

Description:	Defines if the enhanced mode should be used (enables use of table, replace, subscript, superscript, paste from word, etc.).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText::EnhancedMode'} = '0';</code>

2.29.11. DisableMSIFrameSecurityRestricted

Description:	Disable restricted security for IFrames in IE. May be required for SSO to work in IE8.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'DisableMSIFrameSecurityRestricted'} = '0';</code>
-----------------	--

2.29.12. DisableContentSecurityPolicy

Description:	Disable HTTP header "Content-Security-Policy" to allow loading of external script contents. Disabling this HTTP header can be a security issue! Only disable it, if you know what you are doing!
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'DisableContentSecurityPolicy'} = '0';</code>

2.29.13. DefaultViewNewLine

Description:	Automated line break in text messages after x number of chars.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultViewNewLine'} = '90';</code>

2.29.14. DefaultViewLines

Description:	Sets the number of lines that are displayed in text messages (e.g. ticket lines in the QueueZoom).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultViewLines'} = '6000';</code>

2.29.15. Frontend::AnimationEnabled

Description:	Turns on the animations used in the GUI. If you have problems with these animations (e.g. performance issues), you can turn them off here.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::AnimationEnabled'} = '1';</code>

2.29.16. AttachmentDownloadType

Description:	Allows choosing between showing the attachments of a ticket in the browser (inline) or just make them downloadable (attachment).
--------------	--

Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AttachmentDownloadType'} = 'attachment';</code>

2.29.17. WebMaxFileUpload

Description:	Defines the maximal size (in bytes) for file uploads via the browser. Warning: Setting this option to a value which is too low could cause many masks in your OTRS instance to stop working (probably any mask which takes input from the user).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebMaxFileUpload'} = '24000000';</code>

2.29.18. WebUploadCacheModule

Description:	Selects the module to handle uploads via the web interface. "DB" stores all uploads in the database, "FS" uses the file system.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebUploadCacheModule'} = 'Kernel::System::Web::UploadCache::DB';</code>

2.29.19. Frontend::Output::FilterText###AAAURL

Description:	Defines the filter that processes the text in the articles, in order to highlight URLs.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::Output::FilterText'}->{'AAAURL'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextURL', 'Templates' => { 'AgentTicketZoom' => '1' } };</code>

2.29.20. Frontend::Themes

Description:	Activates the available themes on the system. Value 1 means active, 0 means inactive.
Group:	Framework
SubGroup:	Core::Web

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Themes'} = { 'Lite' => '0', 'Standard' => '1' };</pre>

2.29.21.

Frontend::Output::FilterText###OutputFilterTextAutoLink

Description:	Defines a filter to process the text in the articles, in order to highlight predefined keywords.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::FilterText'}->{'OutputFilterTextAutoLink'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextAutoLink', 'Templates' => { 'AgentTicketZoom' => '1' } };</pre>

2.29.22.

Frontend::Output::OutputFilterTextAutoLink###CVE

Description:	Defines a filter for html output to add links behind CVE numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'CVE'} = { 'RegExp' => ['(CVE CAN)\{3,4}\{2,}'], 'Templates' => { 'AgentTicketZoom' => '1' }, 'URL1' => { 'Description' => 'Mitre', 'Image' => 'http://cve.mitre.org/favicon.ico', 'Target' => '_blank', 'URL' => 'http://cve.mitre.org/cgi-bin/cvename.cgi?name=<MATCH1>-<MATCH2>-<MATCH3>' }, 'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=<MATCH1>-<MATCH2>-<MATCH3>' }, 'URL3' => { 'Description' => 'US-CERT NVD', 'Image' => 'http://nvd.nist.gov/favicon.ico', } };</pre>

	<pre>'Target' => '_blank', 'URL' => 'http://nvd.nist.gov/nvd.cfm?cvename=<MATCH1>-<MATCH2>-<MATCH3>' } };</pre>
--	---

2.29.23.

Frontend::Output::OutputFilterTextAutoLink###Bugtraq

Description:	Defines a filter for html output to add links behind bugtraq numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Bugtraq'} = { 'RegExp' => ['Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})', 'Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?(\\d{2,8})', 'Bugtraq[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})', 'Bugtraq[\\s\\w\\t]*?(\\d{2,8})', 'BID[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})', 'BID[\\s\\w\\t]*?(\\d{2,8})'], 'Templates' => { 'AgentTicketZoom' => '1' }, }, 'URL1' => { 'Description' => 'Security Focus', 'Image' => 'http://www.securityfocus.com/favicon.ico', 'Target' => '_blank', 'URL' => 'http://www.securityfocus.com/bid/<MATCH1>/info' }, 'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=<MATCH>' } } };</pre>

2.29.24.

Frontend::Output::OutputFilterTextAutoLink###MSBulletins

Description:	Defines a filter for html output to add links behind MSBulletin numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'MSBulletins'} = { 'RegExp' => ['MS[^A-Za-z]{0,5}(\\d\\d)?.?(\\d{2,4})'], },</pre>

	<pre>'Templates' => { 'AgentTicketZoom' => '1' }, 'URL1' => { 'Description' => 'Microsoft Technet', 'Image' => 'http://www.microsoft.com/favicon.ico', 'Target' => '_blank', 'URL' => 'http://www.microsoft.com/technet/security/bulletin/ MS<MATCH1>-<MATCH2>.mspx' }, 'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=MS<MATCH1>-<MATCH2>' } };</pre>
--	--

2.29.25. Frontend::Output::OutputFilterTextAutoLink###Setting1

Description:	Define a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Setting1'} = { 'RegExp' => ['RegExp'], 'Templates' => { 'AgentTicketZoom' => '1' }, 'URL1' => { 'Description' => 'Description', 'Image' => 'right-small.png', 'Target' => '_blank', 'URL' => 'URL' }, 'URL2' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' } };</pre>

2.29.26. Frontend::Output::OutputFilterTextAutoLink###Setting2

Description:	Defines a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.
Group:	Framework
SubGroup:	Core::Web
Valid:	0

Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Setting2'} = { 'RegExp' => ['RegExp'], 'Templates' => { 'AgentTicketZoom' => '1' }, 'URL1' => { 'Description' => 'Description', 'Image' => 'right-small.png', 'Target' => '_blank', 'URL' => 'URL' }, 'URL2' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' }, 'URL3' => { 'Description' => 'Description', 'Image' => 'Image', 'Target' => '_blank', 'URL' => 'URL' } }; </pre>

2.29.27. Loader::Enabled::CSS

Description:	If enabled, OTRS will deliver all CSS files in minified form. WARNING: If you turn this off, there will likely be problems in IE 7, because it cannot load more than 32 CSS files.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::CSS'} = '1';</code>

2.29.28. Loader::Enabled::JS

Description:	If enabled, OTRS will deliver all JavaScript files in minified form.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::JS'} = '1';</code>

2.29.29. Loader::Agent::CommonCSS###000-Framework

Description:	List of CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1

Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonCSS'}->{'000-Framework'} = ['Core.Reset.css', 'Core.Default.css', 'Core.Header.css', 'Core.OverviewControl.css', 'Core.OverviewSmall.css', 'Core.OverviewMedium.css', 'Core.OverviewLarge.css', 'Core.Footer.css', 'Core.PageLayout.css', 'Core.Form.css', 'Core.Table.css', 'Core.Widget.css', 'Core.WidgetMenu.css', 'Core.TicketDetail.css', 'Core.Tooltip.css', 'Core.Dialog.css', 'Core.Print.css', 'thirdparty/fontawesome/font-awesome.css'];</pre>

2.29.30. Loader::Agent::CommonCSS::IE8###000-Framework

Description:	List of IE8-specific CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonCSS::IE8'}->{'000-Framework'} = ['Core.OverviewSmall.IE8.css'];</pre>

2.29.31. Loader::Agent::CommonJS###000-Framework

Description:	List of JS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonJS'}->{'000-Framework'} = ['thirdparty/jquery-1.10.0/jquery.js', 'thirdparty/jquery-browser-detection/jquery-browser-detection.js', 'thirdparty/jquery-ui-1.10.3/jquery-ui.js', 'thirdparty/jquery-validate-1.11.1/jquery.validate.js', 'thirdparty/stacktrace-0.4/stacktrace.js', 'thirdparty/jquery-pubsub/pubsub.js', 'thirdparty/jquery-jstree-v.pre1.0/jquery.jstree.js', 'thirdparty/jquery-jstree-v.pre1.0/_lib/jquery.hotkeys.js', 'Core.JavaScriptEnhancements.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Config.js', 'Core.Exception.js', 'Core.JSON.js', 'Core.App.js', 'Core.AJAX.js', 'Core.UI.js',];</pre>

	<pre>'Core.UI.Accordion.js', 'Core.UI.Datepicker.js', 'Core.UI.Resizable.js', 'Core.UI.Table.js', 'Core.UI.Accessibility.js', 'Core.UI.RichTextEditor.js', 'Core.UI.Dialog.js', 'Core.UI.ActionRow.js', 'Core.UI.Popup.js', 'Core.UI.TreeSelection.js', 'Core.UI.Autocomplete.js', 'Core.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.Agent.js', 'Core.Agent.Search.js', 'Core.Agent.CustomerInformationCenterSearch.js'];</pre>
--	---

2.29.32. Loader::Agent::CommonJS###001-JQueryMigrate

Description:	List of JS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonJS'}->{'001-JQueryMigrate'} = ['thirdparty/jquery-migrate-1.2.1/jquery-migrate.js'];</pre>

2.29.33. Loader::Customer::CommonCSS###000-Framework

Description:	List of CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS'}->{'000-Framework'} = ['Core.Reset.css', 'Core.Default.css', 'Core.Form.css', 'Core.Dialog.css', 'Core.Tooltip.css', 'Core.Login.css', 'Core.Control.css', 'Core.Table.css', 'Core.TicketZoom.css', 'Core.Print.css', 'thirdparty/fontawesome/font-awesome.css'];</pre>

2.29.34. Loader::Customer::CommonCSS::IE8###000-Framework

Description:	List of IE8-specific CSS files to always be loaded for the customer interface.
--------------	--

Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Customer::CommonCSS::IE8'}->{'000-Framework'} = [];</code>

2.29.35. Loader::Customer::CommonJS###000-Framework

Description:	List of JS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Customer::CommonJS'}->{'000-Framework'} = ['thirdparty/jquery-1.10.0/jquery.js', 'thirdparty/jquery-browser-detection/jquery-browser-detection.js', 'thirdparty/jquery-validate-1.11.1/jquery.validate.js', 'thirdparty/jquery-ui-1.10.3/jquery-ui.js', 'thirdparty/stacktrace-0.4/stacktrace.js', 'thirdparty/jquery-pubsub/pubsub.js', 'thirdparty/jquery-jstree-v.pre1.0/jquery.jstree.js', 'thirdparty/jquery-jstree-v.pre1.0/_lib/jquery.hotkeys.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Exception.js', 'Core.JSON.js', 'Core.JavaScriptEnhancements.js', 'Core.Config.js', 'Core.App.js', 'Core.AJAX.js', 'Core.UI.js', 'Core.UI.Accessibility.js', 'Core.UI.Dialog.js', 'Core.UI.RichTextEditor.js', 'Core.UI.Datepicker.js', 'Core.UI.Popup.js', 'Core.UI.TreeSelection.js', 'Core.UI.Autocomplete.js', 'Core.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.Customer.js'];</code>

2.29.36. Loader::Customer::CommonJS###001-JQueryMigrate

Description:	List of JS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Loader::Customer::CommonJS'}->{'001-JQueryMigrate'} = ['thirdparty/jquery-migrate-1.2.1/jquery-migrate.js'];</code>

2.29.37. Loader::Agent::DefaultSelectedSkin

Description:	The agent skin's InternalName which should be used in the agent interface. Please check the available skins in Frontend::Agent::Skins.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Agent::DefaultSelectedSkin'} = 'default';</code>

2.29.38. Loader::Customer::SelectedSkin::HostBased

Description:	It is possible to configure different skins, for example to distinguish between different customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Loader::Customer::SelectedSkin::HostBased'} = { 'host1\\.example\\.com' => 'Someskin1', 'host2\\.example\\.com' => 'Someskin2' };</code>

2.30. Core::WebUserAgent

2.30.1. WebUserAgent::Timeout

Description:	Sets the timeout (in seconds) for http/ftp downloads.
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebUserAgent::Timeout'} = '15';</code>

2.30.2. WebUserAgent::Proxy

Description:	Defines the connections for http/ftp, via a proxy.
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'WebUserAgent::Proxy'} = 'http://proxy.sn.no:8001/';</code>

2.30.3. WebUserAgent::DisableSSLVerification

Description:	Turns off SSL certificate validation, for example if you use a transparent HTTPS proxy. Use at your own risk!
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'WebUserAgent::DisableSSLVerification'} = '0';</code>

2.31. Crypt::PGP

2.31.1. PGP

Description:	Enables PGP support. When PGP support is enabled for signing and encrypting mail, it is HIGHLY recommended that the web server runs as the OTRS user. Otherwise, there will be problems with the privileges when accessing .gnupg folder.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP'} = '0';</code>

2.31.2. PGP::Bin

Description:	Defines the path to PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Bin'} = '/usr/bin/gpg';</code>

2.31.3. PGP::Options

Description:	Sets the options for PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Options'} = '--homedir /opt/otrs/.gnupg/ --batch --no-tty --yes';</code>

2.31.4. PGP::Key::Password

Description:	Sets the password for private PGP key.
--------------	--

Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PGP::Key::Password'} = { '488A0B8F' => 'SomePassword', 'D2DF79FA' => 'SomePassword' };</pre>

2.31.5. PGP::TrustedNetwork

Description:	Set this to yes if you trust in all your public and private pgp keys, even if they are not certified with a trusted signature.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PGP::TrustedNetwork'} = '0';</pre>

2.31.6. PGP::Log

Description:	Configure your own log text for PGP.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PGP::Log'} = { 'BADSIG' => 'The PGP signature with the keyid has not been verified successfully.', 'ERRSIG' => 'It was not possible to check the PGP signature, this may be caused by a missing public key or an unsupported algorithm.', 'EXPKEYSIG' => 'The PGP signature was made by an expired key.', 'GOODSIG' => 'Good PGP signature.', 'KEYREVOKED' => 'The PGP signature was made by a revoked key, this could mean that the signature is forged.', 'NODATA' => 'No valid OpenPGP data found.', 'NO_PUBKEY' => 'No public key found.', 'REVKEYSIG' => 'The PGP signature was made by a revoked key, this could mean that the signature is forged.', 'SIGEXPIRED' => 'The PGP signature is expired.', 'SIG_ID' => 'Signature data.', 'TRUST_UNDEFINED' => 'This key is not certified with a trusted signature!.', 'VALIDSIG' => 'The PGP signature with the keyid is good.' };</pre>

2.31.7. PGP::StoreDecryptedData

Description:	If this option is enabled, then the decrypted data will be stored in the database if they are displayed in AgentTicketZoom.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'PGP::StoreDecryptedData'} = '1';</code>

2.32. Crypt::SMIME

2.32.1. SMIME

Description:	Enables S/MIME support.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME'} = '0';</code>

2.32.2. SMIME::Bin

Description:	Defines the path to open ssl binary. It may need a HOME env (<code>\$ENV{HOME} = '/var/lib/wwwrun';</code>).
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::Bin'} = '/usr/bin/openssl';</code>

2.32.3. SMIME::CertPath

Description:	Specifies the directory where SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::CertPath'} = '/etc/ssl/certs';</code>

2.32.4. SMIME::PrivatePath

Description:	Specifies the directory where private SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::PrivatePath'} = '/etc/ssl/private';</code>

2.32.5. SMIME::CacheTTL

Description:	Cache time in seconds for the SSL certificate attributes.
--------------	---

Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::CacheTTL'} = '86400';</code>

2.32.6. SMIME::StoreDecryptedData

Description:	If this option is enabled, then the decrypted data will be stored in the database if they are displayed in AgentTicketZoom.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::StoreDecryptedData'} = '1';</code>

2.33. CustomerInformationCenter

2.33.1.

AgentCustomerInformationCenter::MainMenu###010-EditCustomerID

Description:	Main menu registration.
Group:	Framework
SubGroup:	CustomerInformationCenter
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'AgentCustomerInformationCenter::MainMenu'}->{'010-EditCustomerID'} = { 'Link' => '\$Env{"Baselink"}Action=AdminCustomerCompany;Subaction=Change;CustomerID= \$LQData{"CustomerID"};Nav=0', 'Name' => 'Edit customer company' };</code>

2.34. Frontend::Admin

2.34.1. Events###Package

Description:	List of all Package events to be displayed in the GUI.
Group:	Framework
SubGroup:	Frontend::Admin
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Events'}->{'Package'} = ['PackageInstall', 'PackageReinstall', 'PackageUpgrade',]</code>

	<code>'PackageUninstall'</code> <code>];</code>
--	--

2.34.2. Events###CustomerUser

Description:	List of all CustomerUser events to be displayed in the GUI.
Group:	Framework
SubGroup:	Frontend::Admin
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Events'}->{'CustomerUser'} = [</code> <code> 'CustomerUserAdd',</code> <code> 'CustomerUserUpdate'</code> <code>];</code>

2.34.3. Events###CustomerCompany

Description:	List of all CustomerCompany events to be displayed in the GUI.
Group:	Framework
SubGroup:	Frontend::Admin
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Events'}->{'CustomerCompany'} = [</code> <code> 'CustomerCompanyAdd',</code> <code> 'CustomerCompanyUpdate'</code> <code>];</code>

2.35. Frontend::Admin::AdminCustomerUser

2.35.1. AdminCustomerUser::RunInitialWildcardSearch

Description:	Runs an initial wildcard search of the existing customer users when accessing the AdminCustomerUser module.
Group:	Framework
SubGroup:	Frontend::Admin::AdminCustomerUser
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AdminCustomerUser::RunInitialWildcardSearch'} = '1';</code>

2.36. Frontend::Admin::AdminSelectBox

2.36.1. AdminSelectBox::AllowDatabaseModification

Description:	Controls if the admin is allowed to make changes to the database via AdminSelectBox.
Group:	Framework
SubGroup:	Frontend::Admin::AdminSelectBox
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'AdminSelectBox::AllowDatabaseModification'} = '0';</code>

2.37. Frontend::Admin::ModuleRegistration

2.37.1. Frontend::Module###Admin

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'Admin'} = { 'Description' => 'Admin-Area', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.css'], 'JavaScript' => ['Core.Agent.Admin.SysConfig.js'] }, 'NavBar' => [{ 'AccessKey' => 'a', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=Admin', 'LinkOption' => '', 'Name' => 'Admin', 'NavBar' => 'Admin', 'Prio' => '10000', 'Type' => 'Menu' }], 'NavBarModule' => { 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin' }, 'NavBarName' => 'Admin', 'Title' => '' }; </pre>

2.37.2. Frontend::Module###AdminInit

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminInit'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarName' => '', 'Title' => 'Init' }; </pre>

2.37.3. Frontend::Module###AdminUser

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminUser'} = { 'Description' => 'Create and manage agents.', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Agents' };</pre>

2.37.4. Frontend::Module###AdminGroup

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Groups', 'Prio' => '150' }, 'NavBarName' => 'Admin', 'Title' => 'Groups' };</pre>

2.37.5. Frontend::Module###AdminUserGroup

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], };</pre>


```

'NavBarModule' => {
  'Block' => 'Agent',
  'Description' => 'Link agents to groups.',
  'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin',
  'Name' => 'Agents <-> Groups',
  'Prio' => '200'
},
'NavBarName' => 'Admin',
'Title' => 'Agents <-> Groups'
};

```

2.37.6. Frontend::Module###AdminCustomerUser

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminCustomerUser'} = { 'Description' => 'Edit Customer Users', 'Group' => ['admin', 'users'], 'GroupRo' => [''], 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => '', 'Block' => 'ItemArea', 'Description' => 'Customer User Administration.', 'Link' => 'Action=AdminCustomerUser;Nav=Agent', 'LinkOption' => '', 'Name' => 'Customer User Administration', 'NavBar' => 'Customers', 'Prio' => '9000', 'Type' => '' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage customer users.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customer User', 'Prio' => '300' }, 'NavBarName' => 'Customers', 'Title' => 'Customer Users' }; </pre>

2.37.7. Frontend::Module###AdminCustomerCompany

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerCompany'} = { 'Description' => 'Admin', 'Group' => ['admin', 'users'], 'GroupRo' => [''], 'NavBar' => [{ 'AccessKey' => '', 'Block' => 'ItemArea', 'Description' => 'Customer Company Administration.', 'Link' => 'Action=AdminCustomerCompany;Nav=Agent', 'LinkOption' => '', 'Name' => 'Customer Company Administration', 'NavBar' => 'Customers', 'Prio' => '9100', 'Type' => '' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage customers.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers', 'Prio' => '310' }, 'NavBarName' => 'Admin', 'Title' => 'Customer Companies' }; </pre>
-----------------	---

2.37.8. Frontend::Module###AdminCustomerUserGroup

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Link customer user to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customer User <-> Groups', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Customers <-> Groups' }; </pre>

2.37.9. Frontend::Module###AdminCustomerUserService

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminCustomerUserService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Link customer user to services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customer User <-> Services', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Customer User <-> Services' };</pre>

2.37.10. Frontend::Module###AdminRole

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminRole'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Roles' };</pre>

2.37.11. Frontend::Module###AdminRoleUser

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminRoleUser'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link agents to roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents <-> Roles', 'Prio' => '700' }, 'NavBarName' => 'Admin', };</pre>

	<pre>'Title' => 'Agents <-> Roles' };</pre>
--	--

2.37.12. Frontend::Module###AdminRoleGroup

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminRoleGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link roles to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles <-> Groups', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Roles <-> Groups' };</pre>

2.37.13. Frontend::Module###AdminSMIME

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSMIME'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage S/MIME certificates for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'S/MIME Certificates', 'Prio' => '1100' }, 'NavBarName' => 'Admin', 'Title' => 'S/MIME Management' };</pre>

2.37.14. Frontend::Module###AdminPGP

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPGP'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage PGP keys for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PGP Keys', 'Prio' => '1200' }, 'NavBarName' => 'Admin', 'Title' => 'PGP Key Management' }; </pre>
-----------------	--

2.37.15. Frontend::Module###AdminMailAccount

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminMailAccount'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage POP3 or IMAP accounts to fetch email from.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PostMaster Mail Accounts', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Mail Accounts' }; </pre>

2.37.16. Frontend::Module###AdminPostMasterFilter

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPostMasterFilter'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Filter incoming emails.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PostMaster Filters', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'PostMaster Filters' }; </pre>

2.37.17. Frontend::Module###AdminEmail

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminEmail'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Send notifications to users.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Admin Notification', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Admin Notification' };</pre>

2.37.18. Frontend::Module###AdminSession

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSession'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage existing sessions.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Session Management', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Session Management' };</pre>

2.37.19. Frontend::Module###AdminPerformanceLog

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminPerformanceLog'} = { 'Description' => 'Admin', 'Group' => ['admin'] };</pre>

```

],
'Loader' => {
  'CSS' => [
    'Core.Agent.Admin.PerformanceLog.css'
  ]
},
},
'NavBarModule' => {
  'Block' => 'System',
  'Description' => 'View performance benchmark results.',
  'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin',
  'Name' => 'Performance Log',
  'Prio' => '550'
},
'NavBarName' => 'Admin',
'Title' => 'Performance Log'
};

```

2.37.20. Frontend::Module###AdminRegistration

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminRegistration'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.Registration.css'] }, 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage system registration.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'System Registration', 'Prio' => '350' }, 'NavBarName' => 'Admin', 'Title' => 'System Registration' }; </pre>

2.37.21. Frontend::Module###AdminServiceCenter

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminServiceCenter'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ServiceCenter.css'] }, }; </pre>

	<pre>'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage OTRS Group services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Service Center', 'Prio' => '351' }, 'NavBarName' => 'Admin', 'Title' => 'Service Center' };</pre>
--	---

2.37.22. Frontend::Module###AdminLog

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminLog'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'View system log messages.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'System Log', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'System Log' };</pre>

2.37.23. Frontend::Module###AdminSelectBox

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSelectBox'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Execute SQL statements.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'SQL Box', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'SQL Box' };</pre>

2.37.24. Frontend::Module###AdminPackageManager

Description:	Frontend module registration for the agent interface.
--------------	---

Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminPackageManager'} = { 'Description' => 'Software Package Manager', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Update and extend your system with software packages.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Package Manager', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Package Manager' };</pre>

2.38. Frontend::Agent

2.38.1. AgentLogo

Description:	The logo shown in the header of the agent interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AgentLogo'} = { 'StyleHeight' => '67px', 'StyleRight' => '38px', 'StyleTop' => '-4px', 'StyleWidth' => '244px', 'URL' => 'skins/Agent/default/img/logo_bg.png' };</pre>

2.38.2. AgentLoginLogo

Description:	The logo shown on top of the login box of the agent interface. The URL to the image must be relative URL to the skin image directory.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'AgentLoginLogo'} = { 'StyleHeight' => '100px', 'URL' => 'skins/Agent/default/img/loginlogo_default.png' };</pre>

2.38.3. LoginURL

Description:	Defines an alternate URL, where the login link refers to.
--------------	---

Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'LoginURL'} = 'http://host.example.com/login.html';</code>

2.38.4. LogoutURL

Description:	Defines an alternate URL, where the logout link refers to.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'LogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';</code>

2.38.5. PreApplicationModule###AgentInfo

Description:	Defines a useful module to load specific user options or to display news.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PreApplicationModule'}->{'AgentInfo'} = 'Kernel::Modules::AgentInfo';</code>

2.38.6. InfoKey

Description:	Defines the key to be checked with Kernel::Modules::AgentInfo module. If this user preferences key is true, the message is accepted by the system.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'InfoKey'} = 'wpt22';</code>

2.38.7. InfoFile

Description:	File that is displayed in the Kernel::Modules::AgentInfo module, if located under Kernel/Output/HTML/Standard/AgentInfo.dtl.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'InfoFile'} = 'AgentInfo';</code>

2.38.8. LostPassword

Description:	Activates lost password feature for agents, in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LostPassword'} = '1';</code>

2.38.9. ShowMotd

Description:	Shows the message of the day on login screen of the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ShowMotd'} = '0';</code>

2.38.10. NotificationSubjectLostPasswordToken

Description:	Defines the subject for notification mails sent to agents, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSubjectLostPasswordToken'} = 'New OTRS password request';</code>

2.38.11. NotificationBodyLostPasswordToken

Description:	Defines the body text for notification mails sent to agents, with token about new requested password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'NotificationBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has requested to change your OTRS password. If you want to do this, click on the link below. You will receive another email containing the password. <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>index.pl? Action=LostPassword;Token=<OTRS_TOKEN> </pre>

	If you did not request a new password, please ignore this email. ';
--	--

2.38.12. NotificationSubjectLostPassword

Description:	Defines the subject for notification mails sent to agents, about new password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSubjectLostPassword'} = 'New OTRS password';</code>

2.38.13. NotificationBodyLostPassword

Description:	Defines the body text for notification mails sent to agents, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'NotificationBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>, Here\'s your new OTRS password. New password: <OTRS_NEWPW> You can log in via the following URL: <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>index.pl '; </pre>

2.38.14. OpenMainMenuOnHover

Description:	If enabled, the first level of the main menu opens on mouse hover (instead of click only).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'OpenMainMenuOnHover'} = '0';</code>

2.38.15. FirstnameLastnameOrder

Description:	Specifies the order in which the firstname and the lastname of agents will be displayed.
Group:	Framework
SubGroup:	Frontend::Agent

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'FirstnameLastnameOrder'} = '0';</code>

2.38.16. Loader::::Skin###000-default

Description:	Default skin for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Loader::<agent>::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'default', 'VisibleName' => 'Default' };</agent></code>

2.38.17. Loader::::Skin###001-slim

Description:	Default skin for the agent interface (slim version).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Loader::<agent>::Skin'}->{'001-slim'} = { 'Description' => "'Slim' skin which tries to save screen space for power users.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'slim', 'VisibleName' => 'Default (Slim)' };</agent></code>

2.38.18. Loader::::Skin###001-ivory

Description:	Balanced white skin by Felix Niklas.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Loader::<agent>::Skin'}->{'001-ivory'} = { 'Description' => 'Balanced white skin by Felix Niklas.', 'HomePage' => 'www.felixniklas.de', 'InternalName' => 'ivory', 'VisibleName' => 'Ivory' };</agent></code>

2.38.19. Loader::::Skin###001-ivory-slim

Description:	Balanced white skin by Felix Niklas (slim version).
Group:	Framework
SubGroup:	Frontend::Agent

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Agent::Skin'}->{'001-ivory-slim'} = { 'Description' => 'Balanced white skin by Felix Niklas (slim version).', 'HomePage' => 'www.felixniklas.de', 'InternalName' => 'ivory-slim', 'VisibleName' => 'Ivory (Slim)' };</pre>

2.38.20.

Loader::Agent::DefaultSelectedSkin::HostBased

Description:	It is possible to configure different skins, for example to distinguish between different agents, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Agent::DefaultSelectedSkin::HostBased'} = { 'host1\\.example\\.com' => 'SomeSkin1', 'host2\\.example\\.com' => 'SomeSkin2' };</pre>

2.38.21. AutoComplete::Agent###Default

Description:	Defines the config options for the autocompletion feature.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AutoComplete::Agent'}->{'Default'} = { 'AutoCompleteActive' => '1', 'ButtonText' => 'Search', 'MaxResultsDisplayed' => '20', 'MinQueryLength' => '2', 'QueryDelay' => '100' };</pre>

2.38.22. AutoComplete::Agent###CustomerSearch

Description:	Defines the config options for the autocompletion feature.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AutoComplete::Agent'}->{'CustomerSearch'} = { 'AutoCompleteActive' => '1',</pre>

	<pre>'ButtonText' => 'Search Customer', 'MaxResultsDisplayed' => '20', 'MinQueryLength' => '2', 'QueryDelay' => '100' };</pre>
--	--

2.38.23. AutoComplete::Agent###UserSearch

Description:	Defines the config options for the autocompletion feature.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AutoComplete::Agent'}->{'UserSearch'} = { 'AutoCompleteActive' => '1', 'ButtonText' => 'Search User', 'MaxResultsDisplayed' => '20', 'MinQueryLength' => '2', 'QueryDelay' => '100' };</pre>

2.38.24. PossibleNextActions

Description:	Defines the list of possible next actions on an error screen.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PossibleNextActions'} = { 'Go to dashboard!' => '\$Env{"CGIHandle"}?Action=AgentDashboard' };</pre>

2.39. Frontend::Agent::Dashboard

2.39.1.

AgentCustomerInformationCenter::Backend###0600-CIC-CustomerCompanyInformation

Description:	Parameters for the dashboard backend of the customer company information of the agent interface . "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AgentCustomerInformationCenter::Backend'}->{'0600-CIC- CustomerCompanyInformation'} = { 'Attributes' => '', 'Block' => 'ContentSmall', 'Default' => '1', 'Description' => 'Customer Company Information',</pre>

	<pre>'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardCustomerCompanyInformation', 'Title' => 'Customer Company Information' };</pre>
--	--

2.39.2. DashboardBackend###0000-ProductNotify

Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0000-ProductNotify'} = { 'Block' => 'ContentLarge', 'CacheTTLLocal' => '1440', 'Default' => '1', 'Description' => 'News about OTRS releases!', 'Group' => 'admin', 'Module' => 'Kernel::Output::HTML::DashboardProductNotify', 'Title' => 'Product News', 'URL' => 'http://www.otrs.com/product.xml' };</pre>

2.39.3. DashboardBackend###0390-UserOutOfOffice

Description:	Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0390-UserOutOfOffice'} = { 'Block' => 'ContentSmall', 'CacheTTLLocal' => '5', 'Default' => '1', 'Description' => '', 'Group' => '', 'IdleMinutes' => '60', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardUserOutOfOffice', 'SortBy' => 'UserFullname', 'Title' => 'Out Of Office' };</pre>

2.39.4. DashboardBackend###0400-UserOnline

Description:	Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to
--------------	--

	restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'DashboardBackend'}->{'0400-UserOnline'} = { 'Block' => 'ContentSmall', 'CacheTTLLocal' => '5', 'Default' => '0', 'Description' => '', 'Filter' => 'Agent', 'Group' => '', 'IdleMinutes' => '60', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardUserOnline', 'ShowEmail' => '0', 'SortBy' => 'UserFullname', 'Title' => 'Online' }; </pre>

2.39.5. DashboardBackend###0410-RSS

Description:	Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'DashboardBackend'}->{'0410-RSS'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '360', 'Default' => '1', 'Description' => '', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardRSS', 'Title' => 'OTRS News', 'URL' => 'http://www.otrs.com/en/rss.xml', 'URL_de' => 'http://www.otrs.com/de/rss.xml', 'URL_es' => 'http://www.otrs.com/es/rss.xml', 'URL_nl' => 'http://www.otrs.com/nl/rss.xml', 'URL_ru' => 'http://www.otrs.com/ru/rss.xml', 'URL_zh' => 'http://www.otrs.com/cn/rss.xml' }; </pre>

2.39.6. DashboardBackend###0200-Image

Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
--------------	---

Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0200-Image'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Description' => 'Some picture description!', 'Group' => '', 'Height' => '140', 'Link' => 'http://otrs.org/', 'LinkTitle' => 'http://otrs.org/', 'Module' => 'Kernel::Output::HTML::DashboardImage', 'Title' => 'A picture', 'URL' => 'http://www.otrs.com/uploads/pics/ jointhecommunity_02.jpg', 'Width' => '198' }; </pre>

2.39.7. DashboardBackend###0210-MOTD

Description:	Shows the message of the day (MOTD) in the agent dashboard. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0210-MOTD'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardMOTD', 'Title' => 'Message of the Day' }; </pre>

2.39.8. DashboardBackend###0300-IFrame

Description:	Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0300-IFrame'} = { 'Align' => 'left', 'Block' => 'ContentLarge', 'Default' => '1', 'Description' => 'Some description!', 'Frameborder' => '1', 'Group' => '', 'Height' => '800', 'Link' => 'http://otrs.org/', 'LinkTitle' => 'OTRS.org/', }; </pre>

	<pre>'Marginheight' => '5', 'Marginwidth' => '5', 'Module' => 'Kernel::Output::HTML::DashboardIFrame', 'Scrolling' => 'auto', 'Title' => 'A Website', 'URL' => 'http://www.otrs.org/', 'Width' => '1024' };</pre>
--	--

2.39.9. AgentCustomerInformationCenter::Backend###0050-CIC-CustomerUserList

Description:	Parameters for the dashboard backend of the customer user list overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AgentCustomerInformationCenter::Backend'}->{'0050-CIC-CustomerUserList'} = { 'Attributes' => '', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All customer users of a CustomerID', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardCustomerUserList', 'Permission' => 'ro', 'Title' => 'Customer Users' };</pre>

2.40. Frontend::Agent::LinkObject

2.40.1. Frontend::AgentLinkObject::WildcardSearch

Description:	Starts a wildcard search of the active object after the link object mask is started.
Group:	Framework
SubGroup:	Frontend::Agent::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::AgentLinkObject::WildcardSearch'} = '0';</pre>

2.41. Frontend::Agent::ModuleMetaHead

2.41.1. Frontend::HeaderMetaModule###100-Refresh

Description:	Defines the module to generate html refresh headers of html sites.
--------------	--

Group:	Framework
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'100-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh' };</pre>

2.42. Frontend::Agent::ModuleNotify

2.42.1. Frontend::NotifyModule###200-UID-Check

Description:	Defines the module to display a notification in the agent interface, if the system is used by the admin user (normally you shouldn't work as admin).
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'200-UID-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationUIDCheck' };</pre>

2.42.2. Frontend::NotifyModule###300-ShowAgentOnline

Description:	Defines the module that shows all the currently logged in agents in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'300-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</pre>

2.42.3. Frontend::NotifyModule###400-ShowCustomerOnline

Description:	Defines the module that shows all the currently logged in customers in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'400-ShowCustomerOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', };</pre>

	<pre>'ShowEmail' => '1';</pre>
--	-----------------------------------

2.42.4. Frontend::NotifyModule###500-OutofOffice-Check

Description:	Defines the module to display a notification in the agent interface, if the agent is logged in while having out-of-office active.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'500-OutofOffice-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationOutofOfficeCheck' };</pre>

2.42.5. Frontend::NotifyModule###900-Generic

Description:	Defines the module that shows a generic notification in the agent interface. Either "Text" - if configured - or the contents of "File" will be displayed.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'900-Generic'} = { 'File' => '<OTRS_CONFIG_Home>/var/notify.txt', 'Link' => 'http://www.otrs.com', 'Module' => 'Kernel::Output::HTML::NotificationGeneric', 'Priority' => 'Warning', 'Text' => 'The OTRS Website' };</pre>

2.43. Frontend::Agent::ModuleRegistration

2.43.1. Frontend::Module###Logout

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'Logout'} = { 'Description' => 'Logout', 'NavBarName' => '', 'Title' => '' };</pre>

2.43.2. Frontend::Module###AgentDashboard

Description:	Frontend module registration for the agent interface.
Group:	Framework

SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentDashboard'} = { 'Description' => 'Agent Dashboard', 'Loader' => { 'CSS' => ['Core.Agent.Dashboard.css', 'Core.AllocationList.css', 'thirdparty/fullcalendar-1.6.1/fullcalendar.css', 'thirdparty/d3js/nv.d3.css'], 'JavaScript' => ['thirdparty/flot-0.8.1/excanvas.js', 'thirdparty/flot-0.8.1/jquery.flot.js', 'thirdparty/fullcalendar-1.6.1/fullcalendar.min.js', 'thirdparty/d3js/d3.v3.min.js', 'thirdparty/d3js/nv.d3.min.js', 'thirdparty/d3js/models/OTRSmultiBarChart.js', 'thirdparty/d3js/models/OTRSstackedAreaChart.js', 'Core.UI.Chart.js', 'Core.UI.AdvancedChart.js', 'Core.UI.DnD.js', 'Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js', 'Core.Agent.Dashboard.js'] }, 'NavBar' => [{ 'AccessKey' => 'd', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentDashboard', 'LinkOption' => '', 'Name' => 'Dashboard', 'NavBar' => 'Dashboard', 'Prio' => '50', 'Type' => 'Menu' }], 'NavBarName' => 'Dashboard', 'Title' => '' }; </pre>

2.43.3. Frontend::Module###AgentCustomerInformationCenter

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentCustomerInformationCenter'} = { 'Description' => 'Customer Information Center', 'Loader' => { 'CSS' => ['Core.AllocationList.css'], 'JavaScript' => ['thirdparty/flot-0.8.1/excanvas.js', 'thirdparty/flot-0.8.1/jquery.flot.js', 'Core.UI.Chart.js', </pre>

```

    'Core.UI.DnD.js',
    'Core.UI.AllocationList.js',
    'Core.Agent.Dashboard.js',
    'Core.Agent.TableFilters.js'
  ]
},
'NavBar' => [
  {
    'AccessKey' => '',
    'Block' => 'ItemArea',
    'Description' => 'Customer Information Center.',
    'Link' => 'Action=AgentCustomerInformationCenter',
    'LinkOption' => 'onclick="window.setTimeout(function()
{Core.Agent.CustomerInformationCenterSearch.OpenSearchDialog();},
0); return false;"',
    'Name' => 'Customer Information Center',
    'NavBar' => 'Customers',
    'Prio' => '50',
    'Type' => ''
  },
  {
    'AccessKey' => 'c',
    'Block' => 'ItemArea',
    'Description' => 'Customer Information Center.',
    'Link' => 'Action=AgentCustomerInformationCenter',
    'LinkOption' => '',
    'Name' => 'Customers',
    'NavBar' => 'Customers',
    'Prio' => '60',
    'Type' => 'Menu'
  }
],
'NavBarName' => 'Customer Information Center',
'Title' => ''
};

```

2.43.4. Frontend::Module###AgentCustomerInformationCenterSearch

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}-> {'AgentCustomerInformationCenterSearch'} = { 'Description' => 'Customer Information Center Search', 'Title' => '' }; </pre>

2.43.5. Frontend::Module###AgentPreferences

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentPreferences'} = { 'Description' => 'Agent Preferences', 'NavBarName' => 'Preferences', 'Title' => '' }; </pre>

```
};
```

2.43.6. Frontend::Module###PictureUpload

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' };</pre>

2.43.7. Frontend::Module###AgentSpelling

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentSpelling'} = { 'Description' => 'Spell checker', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

2.43.8. Frontend::Module###SpellingInline

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

2.43.9. Frontend::Module###AgentBook

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentBook'} = { 'Description' => 'Address book of CustomerUser sources', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Address Book' };</pre>
-----------------	---

2.43.10. Frontend::Module###AgentLinkObject

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentLinkObject'} = { 'Description' => 'Link Object', 'NavBarName' => '', 'Title' => 'Link Object' };</pre>

2.43.11. Frontend::Module###AgentInfo

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentInfo'} = { 'Description' => 'Generic Info module', 'NavBarName' => '', 'Title' => 'Info' };</pre>

2.43.12. Frontend::Module###AgentSearch

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentSearch'} = { 'Description' => 'Global Search Module', 'NavBarName' => '', 'Title' => 'Search' };</pre>

2.43.13. CustomerFrontend::Module###SpellingInline

Description:	Frontend module registration for the customer interface.
Group:	Framework

SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

2.43.14. Frontend::Module###AgentHTMLReference

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentHTMLReference'} = { 'Description' => 'HTML Reference', 'Group' => ['users'], 'GroupRo' => ['users'], 'Loader' => { 'CSS' => ['Core.Agent.HTMLReference.css'] }, 'NavBarName' => '', 'Title' => 'HTML Reference' };</pre>

2.43.15. Frontend::Module###AgentStats

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentStats'} = { 'Description' => 'Stats', 'Group' => ['stats'], 'GroupRo' => ['stats'], 'Loader' => { 'JavaScript' => ['Core.Agent.Stats.js'] }, 'NavBar' => [{ 'AccessKey' => '', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentStats;Subaction=0overview', 'LinkOption' => '' }] };</pre>

```

    'Name' => 'Statistics',
    'NavBar' => 'Stats',
    'Prio' => '8500',
    'Type' => 'Menu'
  },
  {
    'AccessKey' => '',
    'Block' => '',
    'Description' => 'Overview',
    'GroupRo' => [
      'stats'
    ],
    'Link' => 'Action=AgentStats;Subaction=Overview',
    'LinkOption' => '',
    'Name' => 'Overview',
    'NavBar' => 'Stats',
    'Prio' => '100',
    'Type' => ''
  },
  {
    'AccessKey' => '',
    'Block' => '',
    'Description' => 'New',
    'Group' => [
      'stats'
    ],
    'Link' => 'Action=AgentStats;Subaction=Add',
    'LinkOption' => '',
    'Name' => 'New',
    'NavBar' => 'Stats',
    'Prio' => '200',
    'Type' => ''
  },
  {
    'AccessKey' => '',
    'Block' => '',
    'Description' => 'Import',
    'Group' => [
      'stats'
    ],
    'Link' => 'Action=AgentStats;Subaction=Import',
    'LinkOption' => '',
    'Name' => 'Import',
    'NavBar' => 'Stats',
    'Prio' => '300',
    'Type' => ''
  }
],
'NavBarName' => 'Stats',
'Title' => 'Stats'
};

```

2.44. Frontend::Agent::NavBarModule

2.44.1. Frontend::NavBarModule###6-CustomerCompany

Description:	Frontend module registration (disable company link if no company feature is used).
Group:	Framework
SubGroup:	Frontend::Agent::NavBarModule
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::NavBarModule'}->{'6-CustomerCompany'} = { 'Module' => 'Kernel::Output::HTML::NavBarCustomerCompany'</code>

```
};
```

2.45. Frontend::Agent::Preferences

2.45.1. PreferencesTableValue

Description:	Defines the name of the column to store the data in the preferences table.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PreferencesTableValue'} = 'preferences_value';</code>

2.45.2. PreferencesTableUserID

Description:	Defines the name of the column to store the user identifier in the preferences table.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PreferencesTableUserID'} = 'user_id';</code>

2.45.3. PreferencesView

Description:	Sets the display order of the different items in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PreferencesView'} = ['User Profile', 'Email Settings', 'Other Settings'];</code>

2.45.4. PreferencesGroups###Password

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'PreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Agent', 'Column' => 'User Profile',</code>

	<pre>'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMaxLoginFailed' => '0', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', 'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '0500' };</pre>
--	--

2.45.5. PreferencesGroups###SpellDict

Description:	Defines the config parameters of this item, to be shown in the preferences view. Take care to maintain the dictionaries installed in the system in the data section.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'SpellDict'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { 'deutsch' => 'Deutsch', 'english' => 'English' }, 'DataSelected' => 'english', 'Key' => 'Default spelling dictionary', 'Label' => 'Spelling Dictionary', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSpellDict', 'Prio' => '2000' };</pre>

2.45.6. PreferencesGroups###Comment

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Comment'} = { 'Active' => '0', 'Block' => 'Input', 'Column' => 'Other Settings', 'Data' => '\$Env{"UserComment"}', 'Key' => 'Comment', 'Label' => 'Comment', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserComment', 'Prio' => '6000' };</pre>

2.45.7. PreferencesGroups###Language

Description:	Defines the config parameters of this item, to be shown in the preferences view.
--------------	--

Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Language'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend language', 'Label' => 'Language', 'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage', 'Prio' => '1000' };</pre>

2.45.8. PreferencesGroups###Skin

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Skin'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Wear this frontend skin', 'Label' => 'Skin', 'Module' => 'Kernel::Output::HTML::PreferencesSkin', 'PrefKey' => 'UserSkin', 'Prio' => '2000' };</pre>

2.45.9. PreferencesGroups###Theme

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Theme'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend theme', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '3000' };</pre>

2.45.10. PreferencesGroups###OutOfOffice

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework

SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'OutOfOffice'} = { 'Active' => '1', 'Block' => 'OutOfOffice', 'Column' => 'User Profile', 'Key' => '', 'Label' => 'Out Of Office Time', 'Module' => 'Kernel::Output::HTML::PreferencesOutOfOffice', 'PrefKey' => 'UserOutOfOffice', 'Prio' => '4000' };</pre>

2.45.11. PreferencesGroups###TimeZone

Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone', 'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' };</pre>

2.45.12. PreferencesGroups###CSVSeparator

Description:	Gives end users the possibility to override the separator character for CSV files, defined in the translation files.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'CSVSeparator'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { ' ' => ' ', ',' => ',', ';' => ';', '\\t' => 'tab', ' ' => ' ' }, 'DataSelected' => '0', 'Desc' => 'Select the separator character used in CSV files (stats and searches). If you don\\t select a separator here, the default separator for your language will be used.', 'Key' => 'CSV Separator', 'Label' => 'CSV Separator', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserCSVSeparator', 'Prio' => '4000' };</pre>

```
};
```

2.46. Frontend::Agent::SearchRouter

2.46.1. Frontend::SearchDefault

Description:	Search backend default router.
Group:	Framework
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::SearchDefault'} = 'Action=AgentTicketSearch;Subaction=AJAX';</code>

2.47. Frontend::Agent::Stats

2.47.1. Stats::SearchPageShown

Description:	Defines the default maximum number of search results shown on the overview page.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::SearchPageShown'} = '20';</code>

2.47.2. Stats::DefaultSelectedDynamicObject

Description:	Defines the default selection at the drop down menu for dynamic objects (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedDynamicObject'} = 'Ticket';</code>

2.47.3. Stats::DefaultSelectedPermissions

Description:	Defines the default selection at the drop down menu for permissions (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedPermissions'} = ['stats'];</code>

2.47.4. Stats::DefaultSelectedFormat

Description:	Defines the default selection at the drop down menu for stats format (Form: Common Specification). Please insert the format key (see Stats::Format).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::DefaultSelectedFormat'} = ['Print', 'CSV'];</pre>

2.47.5. Stats::SearchLimit

Description:	Defines the search limit for the stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::SearchLimit'} = '500';</pre>

2.47.6. Stats::Format

Description:	Defines all the possible stats output formats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::Format'} = { 'CSV' => 'CSV', 'GD::Graph::area' => 'graph-area', 'GD::Graph::bars' => 'graph-bars', 'GD::Graph::hbars' => 'graph-hbars', 'GD::Graph::lines' => 'graph-lines', 'GD::Graph::linespoints' => 'graph-lines-points', 'GD::Graph::pie' => 'graph-pie', 'GD::Graph::points' => 'graph-points', 'Print' => 'Print' };</pre>

2.47.7. Stats::GraphSize

Description:	Sets the size of the statistic graph.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::GraphSize'} = { '1200x800' => '1200x800', };</pre>

```
'1600x1200' => '1600x1200',
'800x600' => '800x600'
};
```

2.47.8. Stats::TimeType

Description:	Sets the time type which should be shown.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::TimeType'} = 'Extended';</code>

2.47.9. Stats::ExchangeAxis

Description:	Allows agents to exchange the axis of a stat if they generate one.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Stats::ExchangeAxis'} = '0';</code>

2.47.10. Stats::UseAgentElementInStats

Description:	Allows agents to generate individual-related stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Stats::UseAgentElementInStats'} = '0';</code>

2.47.11. Stats::UseInvalidAgentInStats

Description:	Allows invalid agents to generate individual-related stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Stats::UseInvalidAgentInStats'} = '1';</code>

2.47.12. Stats::CustomerIDAsMultiSelect

Description:	Shows all the customer identifiers in a multi-select field (not useful if you have a lot of customer identifiers).
Group:	Framework
SubGroup:	Frontend::Agent::Stats

Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Stats::CustomerIDAsMultiSelect'} = '1';</code>

2.48. Frontend::Customer

2.48.1. CustomerHeadline

Description:	The headline shown in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerHeadline'} = 'Example Company Support';</code>

2.48.2. CustomerLogo

Description:	The logo shown in the header of the customer interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerLogo'} = { 'StyleHeight' => '50px', 'StyleRight' => '25px', 'StyleTop' => '2px', 'StyleWidth' => '135px', 'URL' => 'skins/Customer/default/img/logo.png' };</code>

2.48.3. CustomerPanelUserID

Description:	Defines the user identifier for the customer panel.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelUserID'} = '1';</code>

2.48.4. CustomerGroupSupport

Description:	Activates support for customer groups.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'CustomerGroupSupport'} = '0';</code>
-----------------	---

2.48.5. CustomerGroupAlwaysGroups

Description:	Defines the groups every customer user will be in (if CustomerGroupSupport is enabled and you don't want to manage every user for these groups).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerGroupAlwaysGroups'} = ['users'];</code>

2.48.6. CustomerPanelLoginURL

Description:	Defines an alternate login URL for the customer panel..
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/cgi-bin/login.pl';</code>

2.48.7. CustomerPanelLogoutURL

Description:	Defines an alternate logout URL for the customer panel.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/cgi-bin/login.pl';</code>

2.48.8. Frontend::CustomerUser::Item###1-GoogleMaps

Description:	Defines a customer item, which generates a google maps icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::CustomerUser::Item'}->{'1-GoogleMaps'} = { 'Attributes' => 'UserStreet;UserCity;UserCountry;', 'CSS' => 'Core.Agent.CustomerUser.GoogleMaps.css', 'CSSClass' => 'GoogleMaps', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric',</code>

	<pre>'Required' => 'UserStreet;UserCity;', 'Target' => '_blank', 'Text' => 'Location', 'URL' => 'http://maps.google.com/maps?z=7&q=' };</pre>
--	---

2.48.9. Frontend::CustomerUser::Item###2-Google

Description:	Defines a customer item, which generates a google icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'2-Google'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Google.css', 'CSSClass' => 'Google', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'Google', 'URL' => 'http://google.com/search?q=' };</pre>

2.48.10. Frontend::CustomerUser::Item###2-LinkedIn

Description:	Defines a customer item, which generates a LinkedIn icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'2-LinkedIn'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Linkedin.css', 'CSSClass' => 'LinkedIn', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'LinkedIn', 'URL' => 'http://www.linkedin.com/commonSearch? type=people&keywords=' };</pre>

2.48.11. Frontend::CustomerUser::Item###3-XING

Description:	Defines a customer item, which generates a XING icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'3-XING'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Xing.css', };</pre>

```
'CSSClass' => 'Xing',
'Module' => 'Kernel::Output::HTML::CustomerUserGeneric',
'Required' => 'UserFirstname;UserLastname;',
'Target' => '_blank',
'Text' => 'XING',
'URL' => 'https://www.xing.com/app/search?op=search;keywords='
};
```

2.48.12.

CustomerPanelPreApplicationModule###CustomerAccept

Description:	This module and its PreRun() function will be executed, if defined, for every request. This module is useful to check some user options or to display news about new applications.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelPreApplicationModule'}->{'CustomerAccept'} = 'Kernel::Modules::CustomerAccept';</code>

2.48.13. CustomerPanel::InfoKey

Description:	Defines the key to check with CustomerAccept. If this user preferences key is true, then the message is accepted by the system.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanel::InfoKey'} = 'CustomerAccept1';</code>

2.48.14. CustomerPanel::InfoFile

Description:	Defines the path of the shown info file, that is located under Kernel/Output/HTML/Standard/CustomerAccept.dtl.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanel::InfoFile'} = 'CustomerAccept';</code>

2.48.15. CustomerPanelLostPassword

Description:	Activates lost password feature for customers.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'CustomerPanelLostPassword'} = '1';</code>
-----------------	--

2.48.16. CustomerPanelCreateAccount

Description:	Enables customers to create their own accounts.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelCreateAccount'} = '1';</code>

2.48.17. CustomerPanelSubjectLostPasswordToken

Description:	Defines the subject for notification mails sent to customers, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectLostPasswordToken'} = 'New OTRS password request';</code>

2.48.18. CustomerPanelBodyLostPasswordToken

Description:	Defines the body text for notification mails sent to customers, with token about new requested password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has requested to change your OTRS password. If you want to do this, click on this link. You will receive another email containing the password. <OTRS_CONFIG_HttpType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl? Action=CustomerLostPassword;Token=<OTRS_TOKEN> If you did not request a new password, please ignore this email. '; </pre>

2.48.19. CustomerPanelSubjectLostPassword

Description:	Defines the subject for notification mails sent to customers, about new password.
Group:	Framework

SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectLostPassword'} = 'New OTRS password';</code>

2.48.20. CustomerPanelBodyLostPassword

Description:	Defines the body text for notification mails sent to customers, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>, New password: <OTRS_NEWPW> <OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl '; </pre>

2.48.21. CustomerPanelSubjectNewAccount

Description:	Defines the subject for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectNewAccount'} = 'New OTRS Account!';</code>

2.48.22. CustomerPanelBodyNewAccount

Description:	Defines the body text for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyNewAccount'} = 'Hi <OTRS_USERFIRSTNAME>, You or someone impersonating you has created a new OTRS account for you. Full name: <OTRS_USERFIRSTNAME> <OTRS_USERLASTNAME> User name: <OTRS_USERLOGIN> Password : <OTRS_USERPASSWORD> You can log in via the following URL. We encourage you to change your password via the Preferences button after logging in. </pre>

	<pre><OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl ';</pre>
--	--

2.48.23. Loader::Customer::Skin###000-default

Description:	Default skin for OTRS 3.0 interface.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Customer::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin for OTRS 3.0.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'default', 'VisibleName' => 'Default' };</pre>

2.48.24. Loader::Customer::SelectedSkin

Description:	The customer skin's InternalName which should be used in the customer interface. Please check the available skins in Frontend::Customer::Skins.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::SelectedSkin'} = 'default';</pre>

2.48.25. AutoComplete::Customer###Default

Description:	Defines the config options for the autocompletion feature.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AutoComplete::Customer'}->{'Default'} = { 'AutoCompleteActive' => '1', 'ButtonText' => 'Search', 'MaxResultsDisplayed' => '20', 'MinQueryLength' => '2', 'QueryDelay' => '100' };</pre>

2.49. Frontend::Customer::Auth

2.49.1. Customer::AuthModule

Description:	Defines the module to authenticate customers.
Group:	Framework

SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';</code>

2.49.2. Customer::AuthModule::DB::CryptType

Description:	If "DB" was selected for Customer::AuthModule, the crypt type of passwords must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CryptType'} = 'sha2';</code>

2.49.3. Customer::AuthModule::DB::Table

Description:	If "DB" was selected for Customer::AuthModule, the name of the table where your customer data should be stored must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';</code>

2.49.4. Customer::AuthModule::DB::CustomerKey

Description:	If "DB" was selected for Customer::AuthModule, the name of the column for the CustomerKey in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';</code>

2.49.5. Customer::AuthModule::DB::CustomerPassword

Description:	If "DB" was selected for Customer::AuthModule, the column name for the CustomerPassword in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';</code>

2.49.6. Customer::AuthModule::DB::DSN

Description:	If "DB" was selected for Customer::AuthModule, the DSN for the connection to the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::DSN'} = 'DBI:mysql:database=customerdb;host=customerdbhost';</code>

2.49.7. Customer::AuthModule::DB::User

Description:	If "DB" was selected for Customer::AuthModule, a username to connect to the customer table can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::User'} = 'some_user';</code>

2.49.8. Customer::AuthModule::DB::Password

Description:	If "DB" was selected for Customer::AuthModule, a password to connect to the customer table can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Password'} = 'some_password';</code>

2.49.9. Customer::AuthModule::DB::Type

Description:	If "DB" was selected for Customer::AuthModule, a database driver (normally autodetection is used) can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Type'} = 'mysql';</code>

2.49.10.

Customer::AuthModule::HTTPBasicAuth::Replace

Description:	If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify to strip leading parts of user names (e. g. for domains like example_domain\user to user).
--------------	--

Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::HTTPBasicAuth::Replace'} = 'example_domain\\\\';</code>

2.49.11.

Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp

Description:	If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify (by using a RegExp) to strip parts of REMOTE_USER (e. g. for to remove trailing domains). RegExp-Note, \$1 will be the new Login.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp'} = '^(.+?)@.+?\$';</code>

2.49.12. Customer::AuthModule::LDAP::Host

Description:	If "LDAP" was selected for Customer::AuthModule, the LDAP host can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';</code>

2.49.13. Customer::AuthModule::LDAP::BaseDN

Description:	If "LDAP" was selected for Customer::AuthModule, the BaseDN must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';</code>

2.49.14. Customer::AuthModule::LDAP::UID

Description:	If "LDAP" was selected for Customer::AuthModule, the user identifier must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth

Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';</code>

2.49.15. Customer::AuthModule::LDAP::GroupDN

Description:	If "LDAP" was selected for Customer::Authmodule, you can check if the user is allowed to authenticate because he is in a posixGroup, e.g. user needs to be in a group xyz to use OTRS. Specify the group, who may access the system.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::GroupDN'} = 'cn=otrsallow,ou=posixGroups,dc=example,dc=com';</code>

2.49.16. Customer::AuthModule::LDAP::AccessAttr

Description:	If "LDAP" was selected for Customer::AuthModule, you can specify access attributes here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';</code>

2.49.17. Customer::AuthModule::LDAP::UserAttr

Description:	If "LDAP" was selected for Customer::AuthModule, user attributes can be specified. For LDAP posixGroups use UID, for non LDAP posixGroups use full user DN.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';</code>

2.49.18. Customer::AuthModule::LDAP::SearchUserDN

Description:	If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the username for this special user here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0

Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} = 'cn=binduser,ou=users,dc=example,dc=com';</code>

2.49.19. Customer::AuthModule::LDAP::SearchUserPw

Description:	If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the password for this special user here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = 'some_password';</code>

2.49.20. Customer::AuthModule::LDAP::AlwaysFilter

Description:	If "LDAP" was selected, you can add a filter to each LDAP query, e.g. (mail=*), (objectclass=user) or (!objectclass=computer).
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '(!objectclass=computer)';</code>

2.49.21. Customer::AuthModule::LDAP::UserSuffix

Description:	If "LDAP" was selected for Customer::AuthModule and if you want to add a suffix to every customer login name, specify it here, e. g. you just want to write the username user but in your LDAP directory exists user@domain.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';</code>

2.49.22. Customer::AuthModule::LDAP::Params

Description:	If "LDAP" was selected for Customer::AuthModule and special paramaters are needed for the Net::LDAP perl module, you can specify them here. See "perldoc Net::LDAP" for more information about the parameters.
Group:	Framework
SubGroup:	Frontend::Customer::Auth

Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Params'} = { 'async' => '0', 'port' => '389', 'timeout' => '120', 'version' => '3' };</pre>

2.49.23. Customer::AuthModule::LDAP::Die

Description:	If "LDAP" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Die'} = '1';</pre>

2.49.24. Customer::AuthModule::Radius::Host

Description:	If "Radius" was selected for Customer::AuthModule, the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';</pre>

2.49.25. Customer::AuthModule::Radius::Password

Description:	If "Radius" was selected for Customer::AuthModule, the password to authenticate to the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';</pre>

2.49.26. Customer::AuthModule::Radius::Die

Description:	If "Radius" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::Radius::Die'} = '1';</code>

2.50. Frontend::Customer::ModuleMetaHead

2.50.1. CustomerFrontend::HeaderMetaModule###1-Refresh

Description:	Defines the module to generate html refresh headers of html sites, in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerFrontend::HeaderMetaModule'}->{'1-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh' };</code>

2.51. Frontend::Customer::ModuleNotify

2.51.1. CustomerFrontend::NotifyModule###1-ShowAgentOnline

Description:	Defines the module that shows the currently logged in agents in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</code>

2.51.2. CustomerFrontend::NotifyModule###1-ShowCustomerOnline

Description:	Defines the module that shows the currently logged in customers in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowCustomerOnline'} = { 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', 'ShowEmail' => '1' };</code>

2.52. Frontend::Customer::ModuleRegistration

2.52.1. CustomerFrontend::Module###Logout

Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'Logout'} = { 'Description' => 'Logout of customer panel', 'NavBarName' => '', 'Title' => '' };</pre>

2.52.2. CustomerFrontend::Module###CustomerPreferences

Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerPreferences'} = { 'Description' => 'Customer preferences', 'NavBarName' => '', 'Title' => 'Preferences' };</pre>

2.52.3. CustomerFrontend::Module###CustomerAccept

Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerAccept'} = { 'Description' => 'To accept login information, such as an EULA or license.', 'NavBarName' => '', 'Title' => 'Info' };</pre>

2.52.4. CustomerFrontend::Module###PictureUpload

Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration

Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' };</pre>

2.53. Frontend::Customer::Preferences

2.53.1. PreferencesTable

Description:	Defines the name of the table, where the customer preferences are stored.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PreferencesTable'} = 'user_preferences';</pre>

2.53.2. PreferencesTableKey

Description:	Defines the column to store the keys for the preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PreferencesTableKey'} = 'preferences_key';</pre>

2.53.3. CustomerPreferences

Description:	Defines the parameters for the customer preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerPreferences'} = { 'Module' => 'Kernel::System::CustomerUser::Preferences::DB', 'Params' => { 'Table' => 'customer_preferences', 'TableKey' => 'preferences_key', 'TableUserID' => 'user_id', 'TableValue' => 'preferences_value' } };</pre>

2.53.4. CustomerPreferencesView

Description:	Sets the order of the different items in the customer preferences view.
--------------	---

Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerPreferencesView'} = ['User Profile', 'Other Settings'];</pre>

2.53.5. CustomerPreferencesGroups###Password

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Customer', 'Column' => 'Other Settings', 'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', 'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '1000' };</pre>

2.53.6. CustomerPreferencesGroups###Language

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Language'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Language', 'Label' => 'Interface language', 'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage', 'Prio' => '2000' };</pre>

2.53.7. CustomerPreferencesGroups###Theme

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework

SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Theme'} = { 'Active' => '0', 'Column' => 'User Profile', 'Key' => 'Select your frontend Theme.', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '1000' };</pre>

2.53.8. CustomerPreferencesGroups###TimeZone

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone', 'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' };</pre>

2.53.9. CustomerPreferencesGroups###PGP

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'PGP'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'PGP Key Upload', 'Label' => 'PGP Key', 'Module' => 'Kernel::Output::HTML::PreferencesPGP', 'PrefKey' => 'UserPGPKey', 'Prio' => '10000' };</pre>

2.53.10. CustomerPreferencesGroups###SMIME

Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'SMIME'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'S/MIME Certificate Upload', 'Label' => 'S/MIME Certificate', 'Module' => 'Kernel::Output::HTML::PreferencesSMIME', 'PrefKey' => 'UserSMIMEKey', 'Prio' => '11000' };</pre>

2.54. Frontend::Public

2.54.1. PublicFrontend::CommonParam###Action

Description:	Defines the default value for the action parameter for the public frontend. The action parameter is used in the scripts of the system.
Group:	Framework
SubGroup:	Frontend::Public
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PublicFrontend::CommonParam'}->{'Action'} = 'PublicDefault';</pre>

2.55. Frontend::Public::ModuleRegistration

2.55.1. PublicFrontend::Module###PublicDefault

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}->{'PublicDefault'} = { 'Description' => 'PublicDefault', 'NavBarName' => '', 'Title' => 'PublicDefault' };</pre>

2.55.2. PublicFrontend::Module###PublicRepository

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}->{'PublicRepository'} = { 'Description' => 'PublicRepository', 'NavBarName' => '', 'Title' => 'PublicRepository' };</pre>

2.55.3.

PublicFrontend::Module###PublicSupportDataCollector

Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}->{'PublicSupportDataCollector'} = { 'Description' => 'PublicSupportDataCollector', 'NavBarName' => '', 'Title' => 'PublicSupportDataCollector' };</pre>

3. GenericInterface

3.1. Core::CustomerCompany

3.1.1. CustomerCompany::EventModulePost###1000- GenericInterface

Description:	Performs the configured action for each event (as an Invoker) for each configured Webservice.
Group:	GenericInterface
SubGroup:	Core::CustomerCompany
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerCompany::EventModulePost'}->{'1000- GenericInterface'} = { 'Event' => '', 'Module' => 'Kernel::GenericInterface::Event::Handler', 'Transaction' => '1' };</pre>

3.2. Core::CustomerUser

3.2.1. CustomerUser::EventModulePost###1000- GenericInterface

Description:	Performs the configured action for each event (as an Invoker) for each configured Webservice.
Group:	GenericInterface
SubGroup:	Core::CustomerUser
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerUser::EventModulePost'}->{'1000-GenericInterface'} = { 'Event' => '',</pre>

```
'Module' => 'Kernel::GenericInterface::Event::Handler',
'Transaction' => '1'
};
```

3.3. Core::Package

3.3.1. Package::EventModulePost###1000-GenericInterface

Description:	Performs the configured action for each event (as an Invoker) for each configured Webservice.
Group:	GenericInterface
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Package::EventModulePost'}->{'1000-GenericInterface'} = { 'Event' => '', 'Module' => 'Kernel::GenericInterface::Event::Handler', 'Transaction' => '1' };</pre>

3.4. Core::Ticket

3.4.1. Ticket::EventModulePost###1000-GenericInterface

Description:	Performs the configured action for each event (as an Invoker) for each configured Webservice.
Group:	GenericInterface
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'1000-GenericInterface'} = { 'Event' => '', 'Module' => 'Kernel::GenericInterface::Event::Handler', 'Transaction' => '1' };</pre>

3.5. Frontend::Admin::ModuleRegistration

3.5.1. Frontend::Module###AdminGenericInterfaceDebugger

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGenericInterfaceDebugger'} = {</pre>

```
'Description' => 'Admin',
'Group' => [
  'admin'
],
'Loader' => {
  'CSS' => [
    'Core.Agent.Admin.GenericInterface.css'
  ],
  'CSS_IE7' => 'Core.Agent.Admin.GenericInterface.IE7.css',
  'JavaScript' => [
    'Core.Agent.Admin.GenericInterfaceDebugger.js'
  ]
},
'Title' => 'GenericInterface Debugger GUI'
};
```

3.5.2. Frontend::Module###AdminGenericInterfaceWebservice

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGenericInterfaceWebservice'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceWebservice.js'] }, 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Create and manage web services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Web Services', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'GenericInterface Web Service GUI' };</pre>

3.5.3. Frontend::Module###AdminGenericInterfaceTransportHTTPSOPA

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGenericInterfaceTransportHTTPSOPA'} = { 'Description' => 'Admin',</pre>

	<pre>'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'] }, 'Title' => 'GenericInterface TransporthTTPSOAP GUI' };</pre>
--	---

3.5.4. Frontend::Module###AdminGenericInterfaceWebserviceHistory

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceWebserviceHistory'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceWebserviceHistory.js'] }, 'Title' => 'GenericInterface Webservice History GUI' };</pre>

3.5.5. Frontend::Module###AdminGenericInterfaceOperationDefault

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceOperationDefault'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceOperation.js'] }, 'Title' => 'GenericInterface Operation GUI' };</pre>

3.5.6. Frontend::Module###AdminGenericInterfaceInvokerDefault

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericInterfaceInvokerDefault'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceInvoker.js'] }, 'Title' => 'GenericInterface Invoker GUI' }; </pre>

3.5.7. Frontend::Module###AdminGenericInterfaceMappingSimple

Description:	Frontend module registration for the agent interface.
Group:	GenericInterface
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericInterfaceMappingSimple'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'CSS_IE7' => 'Core.Agent.Admin.GenericInterface.IE7.css', 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceMappingSimple.js'] }, 'Title' => 'GenericInterface Webservice Mapping GUI' }; </pre>

3.6. GenericInterface::Invoker::ModuleRegistration

3.6.1. GenericInterface::Invoker::Module###Test::Test

Description:	GenericInterface module registration for the invoker layer.
--------------	---

Group:	GenericInterface
SubGroup:	GenericInterface::Invoker::ModuleRegistration
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Invoker::Module'}->{'Test::Test'} = { 'ConfigDialog' => 'AdminGenericInterfaceInvokerDefault', 'Controller' => 'Test', 'Name' => 'Test' };</pre>

3.6.2. GenericInterface::Invoker::Module###Test::TestSimple

Description:	GenericInterface module registration for the invoker layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Invoker::ModuleRegistration
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Invoker::Module'}->{'Test::TestSimple'} = { 'ConfigDialog' => 'AdminGenericInterfaceInvokerDefault', 'Controller' => 'Test', 'Name' => 'TestSimple' };</pre>

3.7. GenericInterface::Mapping::ModuleRegistration

3.7.1. GenericInterface::Mapping::Module###Test

Description:	GenericInterface module registration for the mapping layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Mapping::ModuleRegistration
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Mapping::Module'}->{'Test'} = { 'ConfigDialog' => '' };</pre>

3.7.2. GenericInterface::Mapping::Module###Simple

Description:	GenericInterface module registration for the mapping layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Mapping::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Mapping::Module'}->{'Simple'} = { 'ConfigDialog' => 'AdminGenericInterfaceMappingSimple' };</pre>

3.8. GenericInterface::Operation::ModuleRegistration

3.8.1. GenericInterface::Operation::Module###Test::Test

Description:	GenericInterface module registration for the operation layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}->{'Test::Test'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Test', 'Name' => 'Test' };</pre>

3.8.2. GenericInterface::Operation::Module###Session::SessionCreate

Description:	GenericInterface module registration for the operation layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}->{'Session::SessionCreate'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Session', 'Name' => 'SessionCreate' };</pre>

3.8.3. GenericInterface::Operation::Module###Ticket::TicketCreate

Description:	GenericInterface module registration for the operation layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}->{'Ticket::TicketCreate'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Ticket', 'Name' => 'TicketCreate' };</pre>

3.8.4. GenericInterface::Operation::Module###Ticket::TicketUpdate

Description:	GenericInterface module registration for the operation layer.
--------------	---

Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}- >{'Ticket::TicketUpdate'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Ticket', 'Name' => 'TicketUpdate' };</pre>

3.8.5. GenericInterface::Operation::Module###Ticket::TicketGet

Description:	GenericInterface module registration for the operation layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}- >{'Ticket::TicketGet'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Ticket', 'Name' => 'TicketGet' };</pre>

3.8.6. GenericInterface::Operation::Module###Ticket::TicketSearch

Description:	GenericInterface module registration for the operation layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Module'}- >{'Ticket::TicketSearch'} = { 'ConfigDialog' => 'AdminGenericInterfaceOperationDefault', 'Controller' => 'Ticket', 'Name' => 'TicketGet' };</pre>

3.9. GenericInterface::Operation::TicketCreate

3.9.1. GenericInterface::Operation::TicketCreate###ArticleType

Description:	Defines the default type of the article for this operation.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketCreate
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketCreate'}->{'ArticleType'} = 'webrequest';</code>

3.9.2. GenericInterface::Operation::TicketCreate###HistoryType

Description:	Defines the history type for this operation, which gets used for ticket history in the agent interface.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketCreate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketCreate'}->{'HistoryType'} = 'NewTicket';</code>

3.9.3. GenericInterface::Operation::TicketCreate###HistoryComment

Description:	Defines the history comment for this operation, which gets used for ticket history in the agent interface.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketCreate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketCreate'}->{'HistoryComment'} = '%GenericInterface Create';</code>

3.9.4. GenericInterface::Operation::TicketCreate###AutoResponseType

Description:	Defines the default auto response type of the article for this operation.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketCreate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketCreate'}->{'AutoResponseType'} = 'auto reply';</code>

3.10. GenericInterface::Operation::TicketSearch

3.10.1. GenericInterface::Operation::TicketSearch###SearchLimit

Description:	Maximum number of tickets to be displayed in the result of this operation.
Group:	GenericInterface

SubGroup:	GenericInterface::Operation::TicketSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketSearch'}->{'SearchLimit'} = '500';</code>

3.10.2.

GenericInterface::Operation::TicketSearch###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting of the ticket search result of this operation.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketSearch'}->{'SortBy::Default'} = 'Age';</code>

3.10.3.

GenericInterface::Operation::TicketSearch###Order::Default

Description:	Defines the default ticket order in the ticket search result of the this operation. Up: oldest on top. Down: latest on top.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketSearch'}->{'Order::Default'} = 'Down';</code>

3.11.

GenericInterface::Operation::TicketUpdate

3.11.1.

GenericInterface::Operation::TicketUpdate###ArticleType

Description:	Defines the default type of the article for this operation.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketUpdate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketUpdate'}->{'ArticleType'} = 'webrequest';</code>

3.11.2.

GenericInterface::Operation::TicketUpdate###HistoryType

Description:	Defines the history type for this operation, which gets used for ticket history in the agent interface.
--------------	---

Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketUpdate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketUpdate'}->{'HistoryType'} = 'AddNote';</code>

3.11.3.

GenericInterface::Operation::TicketUpdate###HistoryComment

Description:	Defines the history comment for this operation, which gets used for ticket history in the agent interface.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketUpdate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketUpdate'}->{'HistoryComment'} = '%GenericInterface Note';</code>

3.11.4.

GenericInterface::Operation::TicketUpdate###AutoResponseType

Description:	Defines the default auto response type of the article for this operation.
Group:	GenericInterface
SubGroup:	GenericInterface::Operation::TicketUpdate
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Operation::TicketUpdate'}->{'AutoResponseType'} = 'auto follow up';</code>

3.12.

GenericInterface::Transport::ModuleRegistration

3.12.1.

GenericInterface::Transport::Module###HTTP::SOAP

Description:	GenericInterface module registration for the transport layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Transport::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'GenericInterface::Transport::Module'}->{'HTTP::SOAP'} = { 'ConfigDialog' => 'AdminGenericInterfaceTransportHTTPSOAP', 'Name' => 'SOAP', 'Protocol' => 'HTTP' };</code>

3.12.2.

GenericInterface::Transport::Module###HTTP::Test

Description:	GenericInterface module registration for the transport layer.
Group:	GenericInterface
SubGroup:	GenericInterface::Transport::ModuleRegistration
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'GenericInterface::Transport::Module'}->{'HTTP::Test'} = { 'ConfigDialog' => 'AdminGenericInterfaceTransportHTTPTest', 'Name' => 'Test', 'Protocol' => 'HTTP' };</pre>

3.13. GenericInterface::Webservice

3.13.1. GenericInterface::WebserviceConfig::CacheTTL

Description:	Cache time in seconds for the web service config backend.
Group:	GenericInterface
SubGroup:	GenericInterface::Webservice
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'GenericInterface::WebserviceConfig::CacheTTL'} = '86400';</pre>

3.13.2.

GenericInterface::Operation::Common::CachedAuth::AgentCache

Description:	Cache time in seconds for agent authentication in the GenericInterface.
Group:	GenericInterface
SubGroup:	GenericInterface::Webservice
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'GenericInterface::Operation::Common::CachedAuth::AgentCacheTTL'} = '300';</pre>

3.13.3.

GenericInterface::Operation::Common::CachedAuth::CustomerCa

Description:	Cache time in seconds for customer authentication in the GenericInterface.
Group:	GenericInterface
SubGroup:	GenericInterface::Webservice
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'GenericInterface::Operation::Common::CachedAuth::CustomerCacheTTL'} = '300';</code>
-----------------	--

3.13.4. GenericInterface::Webservice::Path::Separator

Description:	Webservice path separator.
Group:	GenericInterface
SubGroup:	GenericInterface::Webservice
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'GenericInterface::Webservice::Path::Separator'} = '>';</code>

4. ProcessManagement

4.1. Core

4.1.1.

Process::DynamicFieldProcessManagementProcessID

Description:	This option defines the dynamic field in which a Process Management process entity id is stored.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::DynamicFieldProcessManagementProcessID'} = 'ProcessManagementProcessID';</code>

4.1.2.

Process::DynamicFieldProcessManagementActivityID

Description:	This option defines the dynamic field in which a Process Management activity entity id is stored.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::DynamicFieldProcessManagementActivityID'} = 'ProcessManagementActivityID';</code>

4.1.3. Process::DefaultQueue

Description:	This option defines the process tickets default queue.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Process::DefaultQueue'} = 'Raw';</code>
-----------------	--

4.1.4. Process::DefaultState

Description:	This option defines the process tickets default state.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::DefaultState'} = 'new';</code>

4.1.5. Process::DefaultLock

Description:	This option defines the process tickets default lock.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::DefaultLock'} = 'unlock';</code>

4.1.6. Process::DefaultPriority

Description:	This option defines the process tickets default priority.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::DefaultPriority'} = '3 normal';</code>

4.1.7. Process::Entity::Prefix

Description:	Default ProcessManagement entity prefixes for entity IDs that are automatically generated.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::Entity::Prefix'} = { 'Activity' => 'A', 'ActivityDialog' => 'AD', 'Process' => 'P', 'Transition' => 'T', 'TransitionAction' => 'TA' };</code>

4.1.8. Process::CacheTTL

Description:	Cache time in seconds for the DB process backend.
Group:	ProcessManagement
SubGroup:	Core

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::CacheTTL'} = '3600';</code>

4.1.9. Process::NavBarOutput::CacheTTL

Description:	Cache time in seconds for the ticket process navigation bar output module.
Group:	ProcessManagement
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Process::NavBarOutput::CacheTTL'} = '900';</code>

4.2. Core::Ticket

4.2.1.

Ticket::EventModulePost###TicketProcessTransitions

Description:	Event module registration. For more performance you can define a trigger event (e. g. Event => TicketCreate).
Group:	ProcessManagement
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'TicketProcessTransitions'} = { 'Event' => '', 'Module' => 'Kernel::System::Ticket::Event::TicketProcessTransitions', 'Transaction' => '1' };</pre>

4.3. Frontend::Admin::ModuleRegistration

4.3.1.

Frontend::Module###AdminProcessManagement

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminProcessManagement'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css', 'Core.AllocationList.css'] } };</pre>

```

    ],
    'JavaScript' => [
      'thirdparty/jsplumb-1.5.5/jsplumb.js',
      'thirdparty/farahey-0.1/farahey.js',
      'thirdparty/jsplumb-labelspacer/label-spacer.js',
      'Core.Agent.Admin.ProcessManagement.js',
      'Core.Agent.Admin.ProcessManagement.Canvas.js',
      'Core.UI.AllocationList.js'
    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Configure Processes.',
    'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin',
    'Name' => 'Process Management',
    'Prio' => '750'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Process Management'
};

```

4.3.2. Frontend::Module###AdminProcessManagementActivity

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminProcessManagementActivity'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.Agent.Admin.ProcessManagement.js', 'Core.UI.AllocationList.js'] }, 'Title' => 'Process Management Activity GUI' }; </pre>

4.3.3. Frontend::Module###AdminProcessManagementActivityDialog

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminProcessManagementActivityDialog'} = { 'Description' => 'Admin', 'Group' => ['admin'] }; </pre>

	<pre>], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.Agent.Admin.ProcessManagement.js', 'Core.UI.AllocationList.js'] }, 'Title' => 'Process Management Activity Dialog GUI' }; </pre>
--	--

4.3.4. Frontend::Module###AdminProcessManagementTransition

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminProcessManagementTransition'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css'], 'JavaScript' => ['Core.Agent.Admin.ProcessManagement.js'] }, 'Title' => 'Process Management Transition GUI' }; </pre>

4.3.5. Frontend::Module###AdminProcessManagementTransitionAction

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminProcessManagementTransitionAction'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css'], 'JavaScript' => ['Core.Agent.Admin.ProcessManagement.js'] }, }; </pre>

```
'Title' => 'Process Management Transition Action GUI'
};
```

4.3.6. Frontend::Module###AdminProcessManagementPath

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminProcessManagementPath'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.ProcessManagement.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.Agent.Admin.ProcessManagement.js', 'Core.UI.AllocationList.js'] }, 'Title' => 'Process Management Path GUI' };</pre>

4.4. Frontend::Agent::ModuleRegistration

4.4.1. Frontend::Module###AgentTicketProcess

Description:	Frontend module registration for the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketProcess'} = { 'Description' => 'Create new process ticket', 'Loader' => { 'CSS' => ['Core.Agent.TicketProcess.css'], 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'p', 'Block' => '', 'Description' => 'Create New process ticket', 'Link' => 'Action=AgentTicketProcess', 'LinkOption' => '', 'Name' => 'New process ticket', 'NavBar' => 'Ticket', 'Prio' => '220', 'Type' => '' }] };</pre>

```

],
'NavBarName' => 'Ticket',
'Title' => 'New process ticket'
};

```

4.5. Frontend::Agent::NavBarModule

4.5.1. Frontend::NavBarOutputModule###1-TicketProcesses

Description:	Frontend module registration (disable ticket processes screen if no process available).
Group:	ProcessManagement
SubGroup:	Frontend::Agent::NavBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NavBarOutputModule'}->{'1-TicketProcesses'} = { 'Module' => 'Kernel::Output::HTML::NavBarOutputModuleAgentTicketProcess' };</pre>

4.6. Frontend::Agent::Ticket::ViewProcess

4.6.1. Ticket::Frontend::AgentTicketProcess###StateType

Description:	Determines the next possible ticket states, for process tickets in the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::Ticket::ViewProcess
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketProcess'}->{'StateType'} = ['new', 'open', 'pending auto', 'pending reminder', 'closed'];</pre>

4.6.2. Ticket::Frontend::CustomerTicketProcess###StateType

Description:	Determines the next possible ticket states, for process tickets in the agent interface.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::Ticket::ViewProcess
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketProcess'}->{'StateType'} = ['new', 'open'];</pre>

];

4.7. Frontend::Agent::Ticket::ViewZoom

4.7.1.

Ticket::Frontend::AgentTicketZoom###ProcessDisplay

Description:	Display settings to override defaults for Process Tickets.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessDisplay'} = { 'NavBarName' => 'Processes', 'WidgetTitle' => 'Process Information' };</pre>

4.7.2.

Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicF

Description:	Dynamic fields groups for process widget. The key is the name of the group, the value contains the fields to be shown. Example: 'Key => My Group', 'Content: Name_X, NameY'.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessWidgetDynamicFieldGroups'} = {};</pre>

4.7.3.

Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicF

Description:	Dynamic fields shown in the process widget in ticket zoom screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	ProcessManagement
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessWidgetDynamicField'} = {};</pre>

4.8. Frontend::Customer::ModuleRegistration

4.8.1.

CustomerFrontend::Module###CustomerTicketProcess

Description:	Frontend module registration for the customer interface.
--------------	--

Group:	ProcessManagement
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::Module'}->{'CustomerTicketProcess'} = { 'Description' => 'Process Ticket', 'Loader' => { 'CSS' => ['Core.Customer.TicketProcess.css'] }, 'NavBar' => [{ 'AccessKey' => 'o', 'Block' => '', 'Description' => 'Create new process ticket', 'Link' => 'Action=CustomerTicketProcess', 'LinkOption' => '', 'Name' => 'New process ticket', 'NavBar' => 'Ticket', 'Prio' => '220', 'Type' => 'Submenu' }], 'NavBarName' => 'Ticket', 'Title' => 'Process ticket' }; </pre>

4.9. Frontend::Customer::NavBarModule

4.9.1. CustomerFrontend::NavBarModule###10-CustomerTicketProcesses

Description:	Frontend module registration (disable ticket processes screen if no process available) for Customer.
Group:	ProcessManagement
SubGroup:	Frontend::Customer::NavBarModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerFrontend::NavBarModule'}->{'10-CustomerTicketProcesses'} = { 'Module' => 'Kernel::Output::HTML::NavBarOutputModuleCustomerTicketProcess' }; </pre>

5. Scheduler

5.1. Core

5.1.1. Scheduler::SleepTime

Description:	Defines scheduler sleep time in seconds after processing all available tasks (floating point number).
Group:	Scheduler
SubGroup:	Core

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Scheduler::SleepTime'} = '1.0';</code>

5.1.2. Scheduler::PIDUpdateTime

Description:	Defines scheduler PID update time in seconds.
Group:	Scheduler
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Scheduler::PIDUpdateTime'} = '600';</code>

5.1.3. Scheduler::RestartAfterSeconds

Description:	Defines the time in seconds after which the Scheduler performs an automatic self-restart.
Group:	Scheduler
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Scheduler::RestartAfterSeconds'} = '86400';</code>

5.1.4. Scheduler::TaskDataLength

Description:	Defines the maximum length (in characters) for a scheduler task data. WARNING: Do not modify this setting unless you are sure of the current Database length for 'task_data' filed from 'scheduler_data_list' table.
Group:	Scheduler
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Scheduler::TaskDataLength'} = '8000';</code>

5.2. Core::Log

5.2.1. Scheduler::LogPath

Description:	Defines the path for scheduler to store its console output (SchedulerOUT.log and SchedulerERR.log).
Group:	Scheduler
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Scheduler::LogPath'} = '<OTRS_CONFIG_Home>/var/log';</code>

5.2.2. Scheduler::Log::DaysToKeep

Description:	Defines the time in days to keep log backup files.
Group:	Scheduler
SubGroup:	Core::Log
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Scheduler::Log::DaysToKeep'} = '10';</code>

5.3. Core::Web

5.3.1. Loader::Agent::CommonJS###000-GenericInterface

Description:	List of JS files to always be loaded for the agent interface.
Group:	Scheduler
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Agent::CommonJS'}->{'000-GenericInterface'} = ['Core.Agent.Admin.Scheduler.js'];</code>

5.4. Frontend::Admin::ModuleRegistration

5.4.1. Frontend::Module###AdminScheduler

Description:	Frontend module registration for the agent interface.
Group:	Scheduler
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::Module'}->{'AdminScheduler'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Title' => 'GenericInterface Webservice Mapping GUI' };</code>

5.5. Frontend::Agent::ModuleNotify

5.5.1. Frontend::NotifyModule###800-Scheduler-Check

Description:	Defines the module to display a notification in the agent interface if the scheduler is not running.
Group:	Scheduler
SubGroup:	Frontend::Agent::ModuleNotify

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'800-Scheduler-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationSchedulerCheck' };</pre>

6. Ticket

6.1. Core

6.1.1. OTRSEscalationEvents::DecayTime

Description:	The duration in minutes after emitting an event, in which the new escalation notify and start events are suppressed.
Group:	Ticket
SubGroup:	Core
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'OTRSEscalationEvents::DecayTime'} = '1440';</pre>

6.2. Core::CustomerCompany

6.2.1. CustomerCompany::EventModulePost###110-UpdateTickets

Description:	Event module that updates tickets after an update of the Customer Company.
Group:	Ticket
SubGroup:	Core::CustomerCompany
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerCompany::EventModulePost'}->{'110-UpdateTickets'} = { 'Event' => 'CustomerCompanyUpdate', 'Module' => 'Kernel::System::CustomerCompany::Event::TicketUpdate', 'Transaction' => '0' };</pre>

6.2.2. CustomerUser::EventModulePost###120-UpdateTickets

Description:	Event module that updates tickets after an update of the Customer User.
Group:	Ticket
SubGroup:	Core::CustomerCompany
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerUser::EventModulePost'}->{'120-UpdateTickets'} = {</pre>

```
'Event' => 'CustomerUserUpdate',
'Module' => 'Kernel::System::CustomerUser::Event::TicketUpdate',
'Transaction' => '0'
};
```

6.3. Core::FulltextSearch

6.3.1. Ticket::SearchIndexModule

Description:	Helps to extend your articles full-text search (From, To, Cc, Subject and Body search). Runtime will do full-text searches on live data (it works fine for up to 50.000 tickets). StaticDB will strip all articles and will build an index after article creation, increasing fulltext searches about 50%. To create an initial index use "bin/otrs.RebuildFulltextIndex.pl".
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndexModule'} = 'Kernel::System::Ticket::ArticleSearchIndex::RuntimeDB';</pre>

6.3.2. Ticket::SearchIndex::Attribute

Description:	Basic fulltext index settings. Execute "bin/otrs.RebuildFulltextIndex.pl" in order to generate a new index.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndex::Attribute'} = { 'WordCountMax' => '1000', 'WordLengthMax' => '30', 'WordLengthMin' => '3' };</pre>

6.3.3. Ticket::SearchIndex::Filters

Description:	Fulltext index regex filters to remove parts of the text.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndex::Filters'} = ['[,\\&\\<\\>\\?\"\\!*\\ \\;\\[\\]\\(\\)\\+\\\$\\^=]', '^[\\':.][\\':.]\$'];</pre>

6.3.4. Ticket::SearchIndex::StopWords

Description:	Stop words for fulltext index. These words will be removed.
Group:	Ticket

SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::SearchIndex::StopWords'} = { 'a' => '1', 'als' => '1', 'and' => '1', 'auf' => '1', 'das' => '1', 'den' => '1', 'der' => '1', 'die' => '1', 'for' => '1', 'im' => '1', 'in' => '1', 'is' => '1', 'of' => '1', 'the' => '1', 'to' => '1', 'und' => '1', 'vom' => '1', 'zu' => '1' };</pre>

6.3.5. Ticket::EventModulePost###98-ArticleSearchIndex

Description:	Builds an article index right after the article's creation.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'98-ArticleSearchIndex'} = { 'Event' => '(ArticleCreate ArticleUpdate)', 'Module' => 'Kernel::System::Ticket::Event::ArticleSearchIndex' };</pre>

6.4. Core::LinkObject

6.4.1. LinkObject::PossibleLink###0200

Description:	Links 2 tickets with a "Normal" type link.
Group:	Ticket
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::PossibleLink'}->{'0200'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'Normal' };</pre>

6.4.2. LinkObject::PossibleLink###0201

Description:	Links 2 tickets with a "ParentChild" type link.
Group:	Ticket

SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::PossibleLink'}->{'0201'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'ParentChild' };</pre>

6.5. Core::PostMaster

6.5.1. PostmasterMaxEmails

Description:	Maximal auto email responses to own email-address a day (Loop-Protection).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostmasterMaxEmails'} = '40';</pre>

6.5.2. PostMasterMaxEmailSize

Description:	Maximal size in KBytes for mails that can be fetched via POP3/POP3S/IMAP/IMAPS (KBytes).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMasterMaxEmailSize'} = '16384';</pre>

6.5.3. PostMasterReconnectMessage

Description:	The "bin/PostMasterMailAccount.pl" will reconnect to POP3/POP3S/IMAP/IMAPS host after the specified count of messages.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMasterReconnectMessage'} = '20';</pre>

6.5.4. LoopProtectionModule

Description:	Default loop protection module.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'LoopProtectionModule'} = 'Kernel::System::PostMaster::LoopProtection::DB';</code>
-----------------	--

6.5.5. LoopProtectionLog

Description:	Path for the log file (it only applies if "FS" was selected for LoopProtectionModule and it is mandatory).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LoopProtectionLog'} = '<OTRS_CONFIG_Home>/var/log/LoopProtection';</code>

6.5.6. PostmasterAutoHTML2Text

Description:	Converts HTML mails into text messages.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterAutoHTML2Text'} = '1';</code>

6.5.7. PostmasterFollowUpSearchInReferences

Description:	Executes follow up checks on In-Reply-To or References headers for mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInReferences'} = '0';</code>

6.5.8. PostmasterFollowUpSearchInBody

Description:	Executes follow up mail body checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInBody'} = '0';</code>

6.5.9. PostmasterFollowUpSearchInAttachment

Description:	Executes follow up mail attachments checks in mails that don't have a ticket number in the subject.
Group:	Ticket

SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInAttachment'} = '0';</code>

6.5.10. PostmasterFollowUpSearchInRaw

Description:	Executes follow up plain/raw mail checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInRaw'} = '0';</code>

6.5.11. PostmasterUserID

Description:	Specifies user id of the postmaster data base.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterUserID'} = '1';</code>

6.5.12. PostmasterDefaultQueue

Description:	Defines the postmaster default queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultQueue'} = 'Raw';</code>

6.5.13. PostmasterDefaultPriority

Description:	Defines the default priority of new tickets.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultPriority'} = '3 normal';</code>

6.5.14. PostmasterDefaultState

Description:	Defines the default state of new tickets.
Group:	Ticket

SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultState'} = 'new';</code>

6.5.15. PostmasterFollowUpState

Description:	Defines the state of a ticket if it gets a follow-up.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpState'} = 'open';</code>

6.5.16. PostmasterFollowUpStateClosed

Description:	Defines the state of a ticket if it gets a follow-up and the ticket was already closed.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostmasterFollowUpStateClosed'} = 'open';</code>

6.5.17. PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner

Description:	Sends agent follow-up notification only to the owner, if a ticket is unlocked (the default is to send the notification to all agents).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner'} = '0';</code>

6.5.18. PostmasterX-Header

Description:	Defines all the X-headers that should be scanned.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterX-Header'} = ['From', 'To', 'Cc', 'Reply-To', 'ReplyTo',</code>

```

'Subject',
'Message-ID',
'Message-Id',
'Resent-To',
'Resent-From',
'Precedence',
'Mailing-List',
'List-Id',
'List-Archive',
'Errors-To',
'References',
'In-Reply-To',
'Auto-Submitted',
'X-Loop',
'X-Spam-Flag',
'X-Spam-Level',
'X-Spam-Score',
'X-Spam-Status',
'X-No-Loop',
'X-Priority',
'Importance',
'X-Mailer',
'User-Agent',
'Organization',
'X-Original-To',
'Delivered-To',
'Envelope-To',
'Return-Path',
'X-OTRS-Owner',
'X-OTRS-OwnerID',
'X-OTRS-Responsible',
'X-OTRS-ResponsibleID',
'X-OTRS-Loop',
'X-OTRS-Priority',
'X-OTRS-Queue',
'X-OTRS-Lock',
'X-OTRS-Ignore',
'X-OTRS-State',
'X-OTRS-State-PendingTime',
'X-OTRS-Type',
'X-OTRS-Service',
'X-OTRS-SLA',
'X-OTRS-CustomerNo',
'X-OTRS-CustomerUser',
'X-OTRS-SenderType',
'X-OTRS-ArticleType',
'X-OTRS-FollowUp-Priority',
'X-OTRS-FollowUp-Queue',
'X-OTRS-FollowUp-Lock',
'X-OTRS-FollowUp-State',
'X-OTRS-FollowUp-State-PendingTime',
'X-OTRS-FollowUp-Type',
'X-OTRS-FollowUp-Service',
'X-OTRS-FollowUp-SLA',
'X-OTRS-FollowUp-SenderType',
'X-OTRS-FollowUp-ArticleType'
];

```

6.5.19. PostMaster::PreFilterModule###1-Match

Description:	Module to filter and manipulate incoming messages. Block/ignore all spam email with From: noreply@ address.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = {</code>

```
'Match' => {
  'From' => 'noreply@'
},
'Module' => 'Kernel::System::PostMaster::Filter::Match',
'Set' => {
  'X-OTRS-Ignore' => 'yes'
},
'StopAfterMatch' => '0'
};
```

6.5.20. PostMaster::PreFilterModule###2-Match

Description:	Module to filter and manipulate incoming messages. Get a 4 digit number to ticket free text, use regex in Match e. g. From => '(.+?)@.+?', and use () as [***] in Set =>.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'2-Match'} = { 'Match' => { 'Subject' => 'SomeNumber:(\\d\\d\\d\\d)' }, 'Module' => 'Kernel::System::PostMaster::Filter::Match', 'Set' => { 'X-OTRS-DynamicField-TicketFreeKey1' => 'SomeNumber', 'X-OTRS-DynamicField-TicketFreeText1' => '[***]' }, 'StopAfterMatch' => '0' };</pre>

6.5.21. PostMaster::PreFilterModule###3-NewTicketReject

Description:	Blocks all the incoming emails that do not have a valid ticket number in subject with From: @example.com address.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'3-NewTicketReject'} = { 'Match' => { 'From' => '@example.com' }, 'Module' => 'Kernel::System::PostMaster::Filter::NewTicketReject', 'Set' => { 'X-OTRS-Ignore' => 'yes' }, 'StopAfterMatch' => '0' };</pre>

6.5.22. PostMaster::PreFilterModule::NewTicketReject::Sender

Description:	Defines the sender for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster

Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostMaster::PreFilterModule::NewTicketReject::Sender'} = 'noreply@example.com';</code>

6.5.23.

PostMaster::PreFilterModule::NewTicketReject::Subject

Description:	Defines the subject for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostMaster::PreFilterModule::NewTicketReject::Subject'} = 'Email Rejected';</code>

6.5.24.

PostMaster::PreFilterModule::NewTicketReject::Body

Description:	Defines the body text for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostMaster::PreFilterModule::NewTicketReject::Body'} = 'Dear Customer, Unfortunately we could not detect a valid ticket number in your subject, so this email can\'t be processed. Please create a new ticket via the customer panel. Thanks for your help! Your Helpdesk Team ';</code>

6.5.25. PostMaster::PreFilterModule###4-CMD

Description:	CMD example setup. Ignores emails where external CMD returns some output on STDOUT (email will be piped into STDIN of some.bin).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostMaster::PreFilterModule'}->{'4-CMD'} = { 'CMD' => '/usr/bin/some.bin', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Ignore' => 'yes' } };</code>

6.5.26. PostMaster::PreFilterModule###5-SpamAssassin

Description:	Spam Assassin example setup. Ignores emails that are marked with SpamAssassin.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = { 'CMD' => '/usr/bin/spamassassin grep -i "X-Spam-Status: yes"', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Ignore' => 'yes' } };</pre>

6.5.27. PostMaster::PreFilterModule###6-SpamAssassin

Description:	Spam Assassin example setup. Moves marked mails to spam queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'6-SpamAssassin'} = { 'CMD' => '/usr/bin/spamassassin grep -i "X-Spam-Status: yes"', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Queue' => 'spam' } };</pre>

6.5.28. PostMaster::PreFilterModule###000-MatchDBSource

Description:	Module to use database filter storage.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-MatchDBSource'} = { 'Module' => 'Kernel::System::PostMaster::Filter::MatchDBSource' };</pre>

6.5.29. PostMaster::PostFilterModule###000-FollowUpArticleTypeCheck

Description:	Module to check if arrived emails should be marked as email-internal (because of original forwarded internal email). ArticleType and SenderType define the values for the arrived email/article.
--------------	--

Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PostFilterModule'}->{'000-FollowUpArticleTypeCheck'} = { 'ArticleType' => 'email-internal', 'Module' => 'Kernel::System::PostMaster::Filter::FollowUpArticleTypeCheck', 'SenderType' => 'customer' };</pre>

6.5.30. PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition1

Description:	Recognize if a ticket is a follow up to an existing ticket using an external ticket number.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition1'} = { 'ArticleType' => 'note-report', 'DynamicFieldName' => 'Name_X', 'FromAddressRegExp' => '\\s*@example.com', 'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition', 'Name' => 'Some Description', 'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*', 'SearchInBody' => '1', 'SearchInSubject' => '1', 'SenderType' => 'system', 'TicketStateTypes' => 'new;open' };</pre>

6.5.31. PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition2

Description:	Recognize if a ticket is a follow up to an existing ticket using an external ticket number.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition2'} = { 'ArticleType' => 'note-report', 'DynamicFieldName' => 'Name_X', 'FromAddressRegExp' => '\\s*@example.com', 'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition', 'Name' => 'Some Description', 'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*', 'SearchInBody' => '1', 'SearchInSubject' => '1', 'SenderType' => 'system', };</pre>


```
'TicketStateTypes' => 'new;open'
};
```

6.5.32. PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition3

Description:	Recognize if a ticket is a follow up to an existing ticket using an external ticket number.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition3'} = { 'ArticleType' => 'note-report', 'DynamicFieldName' => 'Name_X', 'FromAddressRegExp' => '\\s*@example.com', 'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition', 'Name' => 'Some Description', 'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*', 'SearchInBody' => '1', 'SearchInSubject' => '1', 'SenderType' => 'system', 'TicketStateTypes' => 'new;open' };</pre>

6.5.33. PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition4

Description:	Recognize if a ticket is a follow up to an existing ticket using an external ticket number.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition4'} = { 'ArticleType' => 'note-report', 'DynamicFieldName' => 'Name_X', 'FromAddressRegExp' => '\\s*@example.com', 'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition', 'Name' => 'Some Description', 'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*', 'SearchInBody' => '1', 'SearchInSubject' => '1', 'SenderType' => 'system', 'TicketStateTypes' => 'new;open' };</pre>

6.5.34. SendNoAutoResponseRegExp

Description:	If this regex matches, no message will be send by the autoresponder.
Group:	Ticket
SubGroup:	Core::PostMaster

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendNoAutoResponseRegExp'} = '(MAILER-DAEMON postmaster abuse)@.+?\.\.?';</code>

6.5.35. AutoResponseForWebTickets

Description:	If this option is set to 'Yes', tickets created via the web interface, via Customers or Agents, will receive an autoresponse if configured. If this option is set to 'No', no autoresponses will be sent.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AutoResponseForWebTickets'} = '1';</code>

6.6. Core::Stats

6.6.1. Stats::DynamicObjectRegistration###Ticket

Description:	Module to generate ticket statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Stats::DynamicObjectRegistration'}->{'Ticket'} = { 'Module' => 'Kernel::System::Stats::Dynamic::Ticket' };</code>

6.6.2. Stats::DynamicObjectRegistration###TicketList

Description:	Determines if the statistics module may generate ticket lists.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Stats::DynamicObjectRegistration'}->{'TicketList'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketList' };</code>

6.6.3. Stats::DynamicObjectRegistration###TicketAccountedTime

Description:	Module to generate accounted time ticket statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1

Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'TicketAccountedTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketAccountedTime' };</pre>

6.6.4.

Stats::DynamicObjectRegistration###TicketSolutionResponseTi

Description:	Module to generate ticket solution and response time statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}- >{'TicketSolutionResponseTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketSolutionResponseTime' };</pre>

6.7. Core::Ticket

6.7.1. Ticket::Hook

Description:	The identifier for a ticket, e.g. Ticket#, Call#, MyTicket#. The default is Ticket#.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Hook'} = 'Ticket#';</pre>

6.7.2. Ticket::HookDivider

Description:	The divider between TicketHook and ticket number. E.g ' '.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::HookDivider'} = ' ';</pre>

6.7.3. Ticket::SubjectSize

Description:	Max size of the subjects in an email reply.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::SubjectSize'} = '100';</code>
-----------------	--

6.7.4. Ticket::SubjectRe

Description:	The text at the beginning of the subject in an email reply, e.g. RE, AW, or AS.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectRe'} = 'Re';</code>

6.7.5. Ticket::SubjectFwd

Description:	The text at the beginning of the subject when an email is forwarded, e.g. FW, Fwd, or WG.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectFwd'} = 'Fwd';</code>

6.7.6. Ticket::SubjectFormat

Description:	The format of the subject. 'Left' means '[TicketHook#:12345] Some Subject', 'Right' means 'Some Subject [TicketHook#:12345]', 'None' means 'Some Subject' and no ticket number. In the last case you should enable PostmasterFollowupSearchInRaw or PostmasterFollowUpSearchInReferences to recognize followups based on email headers and/or body.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectFormat'} = 'Left';</code>

6.7.7. Ticket::CustomQueue

Description:	Name of custom queue. The custom queue is a queue selection of your preferred queues and can be selected in the preferences settings.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomQueue'} = 'My Queues';</code>

6.7.8. Ticket::NewArticleIgnoreSystemSender

Description:	Ignore article with system sender type for new article feature (e. g. auto responses or email notifications).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NewArticleIgnoreSystemSender'} = '0';</code>

6.7.9. Ticket::ChangeOwnerToEveryone

Description:	Changes the owner of tickets to everyone (useful for ASP). Normally only agent with rw permissions in the queue of the ticket will be shown.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ChangeOwnerToEveryone'} = '0';</code>

6.7.10. Ticket::Responsible

Description:	Enables ticket responsible feature, to keep track of a specific ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Responsible'} = '0';</code>

6.7.11. Ticket::ResponsibleAutoSet

Description:	Automatically sets the owner of a ticket as the responsible for it (if ticket responsible feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::ResponsibleAutoSet'} = '1';</code>

6.7.12. Ticket::Type

Description:	Allows defining new types for ticket (if ticket type feature is enabled).
Group:	Ticket

SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Type'} = '0';</code>

6.7.13. Ticket::Service

Description:	Allows defining services and SLAs for tickets (e. g. email, desktop, network, ...), and escalation attributes for SLAs (if ticket service/SLA feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Service'} = '0';</code>

6.7.14. Ticket::Service::Default::UnknownCustomer

Description:	Allows default services to be selected also for non existing customers.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Service::Default::UnknownCustomer'} = '0';</code>

6.7.15. Ticket::ArchiveSystem

Description:	Activates the ticket archive system to have a faster system by moving some tickets out of the daily scope. To search for these tickets, the archive flag has to be enabled in the ticket search.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ArchiveSystem'} = '0';</code>

6.7.16. Ticket::ArchiveSystem::RemoveSeenFlags

Description:	Controls if the ticket and article seen flags are removed when a ticket is archived.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ArchiveSystem::RemoveSeenFlags'} = '1';</code>

6.7.17. Ticket::ArchiveSystem::RemoveTicketWatchers

Description:	Removes the ticket watcher information when a ticket is archived.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ArchiveSystem::RemoveTicketWatchers'} = '1';</code>

6.7.18. Ticket::CustomerArchiveSystem

Description:	Activates the ticket archive system search in the customer interface.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerArchiveSystem'} = '0';</code>

6.7.19. Ticket::NumberGenerator

Description:	Selects the ticket number generator module. "AutoIncrement" increments the ticket number, the SystemID and the counter are used with SystemID.counter format (e.g. 1010138, 1010139). With "Date" the ticket numbers will be generated by the current date, the SystemID and the counter. The format looks like Year.Month.Day.SystemID.counter (e.g. 200206231010138, 200206231010139). With "DateChecksum" the counter will be appended as checksum to the string of date and SystemID. The checksum will be rotated on a daily basis. The format looks like Year.Month.Day.SystemID.Counter.CheckSum (e.g. 2002070110101520, 2002070110101535). "Random" generates randomized ticket numbers in the format "SystemID.Random" (e.g. 100057866352, 103745394596).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator'} = 'Kernel::System::Ticket::Number::DateChecksum';</code>

6.7.20. Ticket::NumberGenerator::CheckSystemID

Description:	Checks the SystemID in ticket number detection for follow-ups (use "No" if SystemID has been changed after using the system).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::CheckSystemID'} = '1';</code>
-----------------	---

6.7.21. Ticket::NumberGenerator::MinCounterSize

Description:	Sets the minimal ticket counter size (if "AutoIncrement" was selected as TicketNumberGenerator). Default is 5, this means the counter starts from 10000.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::MinCounterSize'} = '5';</code>

6.7.22.

Ticket::NumberGenerator::Date::UseFormattedCounter

Description:	Enables the minimal ticket counter size (if "Date" was selected as TicketNumberGenerator).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::Date::UseFormattedCounter'} = '0';</code>

6.7.23. Ticket::CounterLog

Description:	Log file for the ticket counter.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CounterLog'} = '<OTRS_CONFIG_Home>/var/log/TicketCounter.log';</code>

6.7.24. Ticket::IndexModule

Description:	IndexAccelerator: to choose your backend TicketViewAccelerator module. "RuntimeDB" generates each queue view on the fly from ticket table (no performance problems up to approx. 60.000 tickets in total and 6.000 open tickets in the system). "StaticDB" is the most powerful module, it uses an extra ticket-index table that works like a view (recommended if more than 80.000 and 6.000 open tickets are stored in the system). Use the script "bin/otrs.RebuildTicketIndex.pl" for initial index update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::IndexModule'} = 'Kernel::System::Ticket::IndexAccelerator::RuntimeDB';</code>

6.7.25. Ticket::StorageModule

Description:	Saves the attachments of articles. "DB" stores all data in the database (not recommended for storing big attachments). "FS" stores the data on the filesystem; this is faster but the webserver should run under the OTRS user. You can switch between the modules even on a system that is already in production without any loss of data.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::StorageModule'} = 'Kernel::System::Ticket::ArticleStorageDB';</code>

6.7.26. ArticleDir

Description:	Specifies the directory to store the data in, if "FS" was selected for TicketStorageModule.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ArticleDir'} = '<OTRS_CONFIG_Home>/var/article';</code>

6.7.27. Ticket::EventModulePost###100-ArchiveRestore

Description:	Restores a ticket from the archive (only if the event is a state change, from closed to any open available state).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'100-ArchiveRestore'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::ArchiveRestore' };</code>

6.7.28. Ticket::EventModulePost###110-AcceleratorUpdate

Description:	Updates the ticket index accelerator.
Group:	Ticket
SubGroup:	Core::Ticket

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'110-AcceleratorUpdate'} = { 'Event' => 'TicketStateUpdate TicketQueueUpdate TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketAcceleratorUpdate' };</pre>

6.7.29. Ticket::EventModulePost###120-ForceOwnerResetOnMove

Description:	Resets and unlocks the owner of a ticket if it was moved to another queue.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'120-ForceOwnerResetOnMove'} = { 'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceOwnerReset' };</pre>

6.7.30. Ticket::EventModulePost###130-ForceStateChangeOnLock

Description:	Forces to choose a different ticket state (from current) after lock action. Define the current state as key, and the next state after lock action as content.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'130-ForceStateChangeOnLock'} = { 'Event' => 'TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceState', 'new' => 'open' };</pre>

6.7.31. Ticket::EventModulePost###140-ResponsibleAutoSet

Description:	Automatically sets the responsible of a ticket (if it is not set yet) after the first owner update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'140-ResponsibleAutoSet'} = { 'Event' => 'TicketOwnerUpdate', 'Module' => 'Kernel::System::Ticket::Event::ResponsibleAutoSet' };</pre>

```
};
```

6.7.32. Ticket::EventModulePost###150-TicketPendingTimeReset

Description:	Sets the PendingTime of a ticket to 0 if the state is changed to a non-pending state.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'150-TicketPendingTimeReset'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketPendingTimeReset' };</pre>

6.7.33. Ticket::EventModulePost###500-NotificationEvent

Description:	Sends the notifications which are configured in the admin interface under "Notfication (Event)".
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'500-NotificationEvent'} = { 'Event' => '', 'Module' => 'Kernel::System::Ticket::Event::NotificationEvent', 'Transaction' => '1' };</pre>

6.7.34. Ticket::EventModulePost###900-EscalationIndex

Description:	Updates the ticket escalation index after a ticket attribute got updated.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'900-EscalationIndex'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate TicketCreate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TicketEscalationIndex' };</pre>

6.7.35. Ticket::EventModulePost###900-EscalationStopEvents

Description:	Ticket event module that triggers the escalation stop events.
--------------	---

Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'900-EscalationStopEvents'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TriggerEscalationStopEvents' };</pre>

6.7.36. Ticket::EventModulePost###910-ForceUnlockOnMove

Description:	Forces to unlock tickets after being moved to another queue.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'910-ForceUnlockOnMove'} = { 'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceUnlock' };</pre>

6.7.37. Ticket::EventModulePost###920-TicketArticleNewMessageUpdate

Description:	Update Ticket "Seen" flag if every article got seen or a new Article got created.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'920- TicketArticleNewMessageUpdate'} = { 'Event' => 'ArticleCreate ArticleFlagSet', 'Module' => 'Kernel::System::Ticket::Event::TicketNewMessageUpdate' };</pre>

6.7.38. DynamicFieldFromCustomerUser::Mapping

Description:	Define a mapping between variables of the customer user data (keys) and dynamic fields of a ticket (values). The purpose is to store customer user data in ticket dynamic fields. The dynamic fields must be present in the system and should be enabled for AgentTicketFreeText, so that they can be set/updated manually by the agent. They mustn't be enabled for AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer. If they were, they would have precedence over the automatically set values. To use this mapping, you have to also activate the next setting below.
Group:	Ticket

SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'DynamicFieldFromCustomerUser::Mapping'} = { 'UserFirstname' => 'CustomerFirstname' };</pre>

6.7.39. Ticket::EventModulePost###930-DynamicFieldFromCustomerUser

Description:	This event module stores attributes from CustomerUser as DynamicFields tickets. Please see the setting above for how to configure the mapping.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'930-DynamicFieldFromCustomerUser'} = { 'Event' => '(TicketCreate TicketCustomerUpdate)', 'Module' => 'Kernel::System::Ticket::Event::DynamicFieldFromCustomerUser' };</pre>

6.7.40. Ticket::CustomModule###001-CustomModule

Description:	Overloads (redefines) existing functions in Kernel::System::Ticket. Used to easily add customizations.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::CustomModule'}->{'001-CustomModule'} = 'Kernel::System::Ticket::Custom';</pre>

6.7.41. Ticket::ViewableSenderTypes

Description:	Defines the default viewable sender types of a ticket (default: customer).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableSenderTypes'} = ['\customer\''];</pre>

6.7.42. Ticket::ViewableLocks

Description:	Defines the viewable locks of a ticket. Default: unlock, tmp_lock.
--------------	--

Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableLocks'} = ['\unlock\'', '\tmp_lock\'];</pre>

6.7.43. Ticket::ViewableStateType

Description:	Defines the valid state types for a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableStateType'} = ['new', 'open', 'pending reminder', 'pending auto'];</pre>

6.7.44. Ticket::UnlockStateType

Description:	Defines the valid states for unlocked tickets. To unlock tickets the script "bin/otrs.UnlockTickets.pl" can be used.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::UnlockStateType'} = ['new', 'open'];</pre>

6.7.45. Ticket::PendingNotificationOnlyToOwner

Description:	Sends reminder notifications of unlocked ticket after reaching the reminder date (only sent to ticket owner).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::PendingNotificationOnlyToOwner'} = '0';</pre>

6.7.46. Ticket::PendingNotificationNotToResponsible

Description:	Disables sending reminder notifications to the responsible agent of a ticket (Ticket::Responsible needs to be activated).
Group:	Ticket

SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingNotificationNotToResponsible'} = '0';</code>

6.7.47. Ticket::PendingReminderStateType

Description:	Defines the state type of the reminder for pending tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingReminderStateType'} = ['pending reminder'];</code>

6.7.48. Ticket::PendingAutoStateType

Description:	Determines the possible states for pending tickets that changed state after reaching time limit.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingAutoStateType'} = ['pending auto'];</code>

6.7.49. Ticket::StateAfterPending

Description:	Defines which states should be set automatically (Content), after the pending time of state (Key) has been reached.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::StateAfterPending'} = { 'pending auto close+' => 'closed successful', 'pending auto close-' => 'closed unsuccessful' };</code>

6.7.50. System::Permission

Description:	Standard available permissions for agents within the application. If more permissions are needed, they can be entered here. Permissions must be defined to be effective. Some other good permissions have also been provided built-in: note, close, pending, customer, freetext, move, compose, responsible, forward, and bounce. Make sure that "rw" is always the last registered permission.
--------------	---

Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'System::Permission'} = ['ro', 'move_into', 'create', 'note', 'owner', 'priority', 'rw'];</pre>

6.7.51. Ticket::Permission###1-OwnerCheck

Description:	Module to check the owner of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'1-OwnerCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::OwnerCheck', 'Required' => '0' };</pre>

6.7.52. Ticket::Permission###2-ResponsibleCheck

Description:	Module to check the agent responsible of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'2-ResponsibleCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::ResponsibleCheck', 'Required' => '0' };</pre>

6.7.53. Ticket::Permission###3-GroupCheck

Description:	Module to check if a user is in a special group. Access is granted, if the user is in the specified group and has ro and rw permissions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'3-GroupCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::GroupCheck', 'Required' => '0' };</pre>

6.7.54. Ticket::Permission###4-WatcherCheck

Description:	Module to check the watcher agents of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'4-WatcherCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::WatcherCheck', 'Required' => '0' };</pre>

6.7.55. CustomerTicket::Permission###1-GroupCheck

Description:	Module to check the group permissions for the access to customer tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'1-GroupCheck'} = { 'Granted' => '0', 'Module' => 'Kernel::System::Ticket::CustomerPermission::GroupCheck', 'Required' => '1' };</pre>

6.7.56. CustomerTicket::Permission###2-CustomerUserIDCheck

Description:	Grants access, if the customer ID of the ticket matches the customer user's ID and the customer user has group permissions on the queue the ticket is in.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'2-CustomerUserIDCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerUserIDCheck', 'Required' => '0' };</pre>

6.7.57. CustomerTicket::Permission###3-CustomerIDCheck

Description:	Module to check customer permissions.
Group:	Ticket
SubGroup:	Core::Ticket

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'3-CustomerIDCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerIDCheck', 'Required' => '0' };</pre>

6.7.58. Ticket::DefineEmailFrom

Description:	Defines how the From field from the emails (sent from answers and email tickets) should look like.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefineEmailFrom'} = 'SystemAddressName';</pre>

6.7.59. Ticket::DefineEmailFromSeparator

Description:	Defines the separator between the agents real name and the given queue email address.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefineEmailFromSeparator'} = 'via';</pre>

6.7.60. CustomerNotifyJustToRealCustomer

Description:	Sends customer notifications just to the mapped customer. Normally, if no customer is mapped, the latest customer sender gets the notification.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerNotifyJustToRealCustomer'} = '0';</pre>

6.7.61. AgentSelfNotifyOnAction

Description:	Specifies if an agent should receive email notification of his own actions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'AgentSelfNotifyOnAction'} = '0';</code>
-----------------	--

6.7.62. Ticket::EventModulePost###900-GenericAgent

Description:	Event module registration. For more performance you can define a trigger event (e. g. Event => TicketCreate).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'900-GenericAgent'} = { 'Event' => '', 'Module' => 'Kernel::System::Ticket::Event::GenericAgent', 'Transaction' => '1' };</code>

6.7.63. StandardTemplate::Types

Description:	Defines the list of types for templates.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'StandardTemplate::Types'} = { 'Answer' => 'Answer', 'Create' => 'Create', 'Forward' => 'Forward', 'PhoneCall' => 'Phone call' };</code>

6.8. Core::TicketACL

6.8.1. Ticket::Acl::Module###1-Ticket::Acl::Module

Description:	ACL module that allows closing parent tickets only if all its children are already closed ("State" shows which states are not available for the parent ticket until all child tickets are closed).
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Acl::Module'}->{'1-Ticket::Acl::Module'} = { 'Module' => 'Kernel::System::Ticket::Acl::CloseParentAfterClosedChilds', 'State' => ['closed successful', 'closed unsuccessful'] };</code>

6.8.2. TicketACL::Default::Action

Description:	Default ACL values for ticket actions.
--------------	--

Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TicketACL::Default::Action'} = {};</code>

6.8.3. ACLKeysLevel1Match

Description:	Defines which items are available in first level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'ACLKeysLevel1Match'} = { 'Properties' => 'Properties', 'PropertiesDatabase' => 'PropertiesDatabase' };</code>

6.8.4. ACLKeysLevel1Change

Description:	Defines which items are available in first level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'ACLKeysLevel1Change'} = { 'Possible' => 'Possible', 'PossibleNot' => 'PossibleNot' };</code>

6.8.5. ACLKeysLevel2::Possible

Description:	Defines which items are available in second level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'ACLKeysLevel2::Possible'} = { 'Action' => 'Action', 'ActivityDialog' => 'ActivityDialog', 'Process' => 'Process', 'Ticket' => 'Ticket' };</code>

6.8.6. ACLKeysLevel2::PossibleNot

Description:	Defines which items are available in second level of the ACL structure.
--------------	---

Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'ACLKeysLevel2::PossibleNot'} = { 'ActivityDialog' => 'ActivityDialog', 'Process' => 'Process', 'Ticket' => 'Ticket' };</pre>

6.8.7. ACLKeysLevel2::Properties

Description:	Defines which items are available in second level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'ACLKeysLevel2::Properties'} = { 'CustomerUser' => 'CustomerUser', 'DynamicField' => 'DynamicField', 'Frontend' => 'Frontend', 'Owner' => 'Owner', 'Priority' => 'Priority', 'Process' => 'Process', 'Queue' => 'Queue', 'Responsible' => 'Responsible', 'SLA' => 'SLA', 'Service' => 'Service', 'State' => 'State', 'Ticket' => 'Ticket', 'Type' => 'Type', 'User' => 'User' };</pre>

6.8.8. ACLKeysLevel2::PropertiesDatabase

Description:	Defines which items are available in second level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'ACLKeysLevel2::PropertiesDatabase'} = { 'CustomerUser' => 'CustomerUser', 'DynamicField' => 'DynamicField', 'Owner' => 'Owner', 'Priority' => 'Priority', 'Process' => 'Process', 'Queue' => 'Queue', 'Responsible' => 'Responsible', 'SLA' => 'SLA', 'Service' => 'Service', 'State' => 'State', 'Ticket' => 'Ticket', 'Type' => 'Type', 'User' => 'User' };</pre>

6.8.9. ACLKeysLevel3::Actions###100-Default

Description:	Defines which items are available for 'Action' in third level of the ACL structure.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'ACLKeysLevel3::Actions'}->{'100-Default'} = ['AgentTicketBounce', 'AgentTicketClose', 'AgentTicketCompose', 'AgentTicketCustomer', 'AgentTicketForward', 'AgentTicketFreeText', 'AgentTicketHistory', 'AgentTicketLink', 'AgentTicketLock', 'AgentTicketMerge', 'AgentTicketMove', 'AgentTicketNote', 'AgentTicketOwner', 'AgentTicketPending', 'AgentTicketPhone', 'AgentTicketPhoneInbound', 'AgentTicketPhoneOutbound', 'AgentTicketPrint', 'AgentTicketPriority', 'AgentTicketResponsible', 'AgentTicketWatcher', 'AgentTicketZoom', 'AgentLinkObject', 'CustomerTicketMessage', 'CustomerTicketPrint', 'CustomerTicketProcess', 'CustomerTicketZoom']; </pre>

6.8.10. ACL::CacheTTL

Description:	Cache time in seconds for the DB ACL backend.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	1
Config-Setting:	<pre> \$self->{'ACL::CacheTTL'} = '3600'; </pre>

6.9. Core::TicketBulkAction

6.9.1. Ticket::Frontend::BulkFeature

Description:	Enables ticket bulk action feature for the agent frontend to work on more than one ticket at a time.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkFeature'} = '1';</code>

6.9.2. Ticket::Frontend::BulkFeatureGroup

Description:	Enables ticket bulk action feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkFeatureGroup'} = ['admin', 'users'];</code>

6.10. Core::TicketDynamicFieldDefault

6.10.1.

Ticket::EventModulePost###TicketDynamicFieldDefault

Description:	Event module registration. For more performance you can define a trigger event (e. g. Event => TicketCreate). This is only possible if all Ticket dynamic fields need the same event.
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'TicketDynamicFieldDefault'} = { 'Module' => 'Kernel::System::Ticket::Event::TicketDynamicFieldDefault', 'Transaction' => '1' };</code>

6.10.2.

Ticket::TicketDynamicFieldDefault###Element1

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element1'} = { 'Event' => 'TicketCreate', 'Name' => 'Field1', 'Value' => 'Default' };</code>

6.10.3.

Ticket::TicketDynamicFieldDefault###Element2

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element2'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.4.

Ticket::TicketDynamicFieldDefault###Element3

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element3'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.5.

Ticket::TicketDynamicFieldDefault###Element4

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element4'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>


```
};
```

6.10.6. Ticket::TicketDynamicFieldDefault###Element5

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element5'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.7. Ticket::TicketDynamicFieldDefault###Element6

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element6'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.8. Ticket::TicketDynamicFieldDefault###Element7

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element7'} = { 'Event' => '',</pre>

```
'Name' => '',
'Value' => ''
};
```

6.10.9.

Ticket::TicketDynamicFieldDefault###Element8

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element8'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.10.

Ticket::TicketDynamicFieldDefault###Element9

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element9'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.11.

Ticket::TicketDynamicFieldDefault###Element10

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0

Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element10'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>
-----------------	--

6.10.12.

Ticket::TicketDynamicFieldDefault###Element11

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element11'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.13.

Ticket::TicketDynamicFieldDefault###Element12

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element12'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.14.

Ticket::TicketDynamicFieldDefault###Element13

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0

Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element13'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.15.

Ticket::TicketDynamicFieldDefault###Element14

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element14'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.16.

Ticket::TicketDynamicFieldDefault###Element15

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element15'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.10.17.

Ticket::TicketDynamicFieldDefault###Element16

Description:	Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault

Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element16'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

6.11. Core::TicketWatcher

6.11.1. Ticket::Watcher

Description:	Enables or disables the ticket watcher feature, to keep track of tickets without being the owner nor the responsible.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Watcher'} = '0';</pre>

6.11.2. Ticket::WatcherGroup

Description:	Enables ticket watcher feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::WatcherGroup'} = ['admin', 'users'];</pre>

6.12. Frontend::Admin

6.12.1. Events###Ticket

Description:	List of all ticket events to be displayed in the GUI.
Group:	Ticket
SubGroup:	Frontend::Admin
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Events'}->{'Ticket'} = ['TicketCreate', 'TicketDelete', 'TicketTitleUpdate', 'TicketUnlockTimeoutUpdate', 'TicketQueueUpdate', 'TicketTypeUpdate', 'TicketServiceUpdate', 'TicketSLAUpdate', 'TicketCustomerUpdate', 'TicketPendingTimeUpdate',];</pre>

	<pre>'TicketLockUpdate', 'TicketArchiveFlagUpdate', 'TicketStateUpdate', 'TicketOwnerUpdate', 'TicketResponsibleUpdate', 'TicketPriorityUpdate', 'HistoryAdd', 'HistoryDelete', 'TicketAccountTime', 'TicketMerge', 'TicketSubscribe', 'TicketUnsubscribe', 'TicketFlagSet', 'TicketFlagDelete', 'TicketSlaveLinkAdd', 'TicketSlaveLinkDelete', 'TicketMasterLinkDelete', 'EscalationResponseTimeNotifyBefore', 'EscalationUpdateTimeNotifyBefore', 'EscalationSolutionTimeNotifyBefore', 'EscalationResponseTimeStart', 'EscalationUpdateTimeStart', 'EscalationSolutionTimeStart', 'EscalationResponseTimeStop', 'EscalationUpdateTimeStop', 'EscalationSolutionTimeStop'];</pre>
--	--

6.12.2. Events###Article

Description:	List of all article events to be displayed in the GUI.
Group:	Ticket
SubGroup:	Frontend::Admin
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Events'}->{'Article'} = ['ArticleCreate', 'ArticleUpdate', 'ArticleSend', 'ArticleBounce', 'ArticleAgentNotification', 'ArticleCustomerNotification', 'ArticleAutoResponse', 'ArticleFlagSet', 'ArticleFlagDelete', 'ArticleAgentNotification', 'ArticleCustomerNotification'];</pre>

6.13. Frontend::Admin::ModuleRegistration

6.13.1. Frontend::Module###AdminACL

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminACL'} = { 'Description' => 'Admin', 'Group' => [</pre>

```

    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ACL.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ACL.js'
    ]
  },
  'NavBarModule' => {
    'Block' => 'Ticket',
    'Description' => 'Configure and manage ACLs.',
    'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin',
    'Name' => 'Access Control Lists (ACL)',
    'Prio' => '750'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Access Control Lists (ACL)'
};

```

6.13.2. Frontend::Module###AdminQueue

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminQueue'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Queues', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Queues' }; </pre>

6.13.3. Frontend::Module###AdminTemplate

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminTemplate'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Templates', 'Prio' => '200' } }; </pre>

```

},
'NavBarName' => 'Admin',
'Title' => 'Templates'
};

```

6.13.4. Frontend::Module###AdminQueueTemplates

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminQueueTemplates'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link templates to queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Templates <-> Queues', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Templates <-> Queues' }; </pre>

6.13.5. Frontend::Module###AdminAutoResponse

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage responses that are automatically sent.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses' }; </pre>

6.13.6.

Frontend::Module###AdminQueueAutoResponse

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration

Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminQueueAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link queues to auto responses.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses <-> Queues', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses <-> Queues' }; </pre>

6.13.7. Frontend::Module###AdminAttachment

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminAttachment'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage attachments.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Attachments' }; </pre>

6.13.8. Frontend::Module###AdminTemplateAttachment

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminTemplateAttachment'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link attachments to templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments <-> Templates', 'Prio' => '700' } }; </pre>

```

},
'NavBarName' => 'Admin',
'Title' => 'Attachments <-> Templates'
};

```

6.13.9. Frontend::Module###AdminSalutation

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminSalutation'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage salutations.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Salutations', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Salutations' }; </pre>

6.13.10. Frontend::Module###AdminSignature

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminSignature'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage signatures.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Signatures', 'Prio' => '900' }, 'NavBarName' => 'Admin', 'Title' => 'Signatures' }; </pre>

6.13.11. Frontend::Module###AdminSystemAddress

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSystemAddress'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Set sender email addresses for this system.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Email Addresses', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Email Addresses' }; </pre>
-----------------	---

6.13.12. Frontend::Module###AdminNotification

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminNotification'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Manage notifications that are sent to agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agent Notifications', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Agent Notifications' }; </pre>

6.13.13. Frontend::Module###AdminNotificationEvent

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminNotificationEvent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage event based notifications.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Notifications (Event)', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Notifications (Event)' }; </pre>

6.13.14. Frontend::Module###AdminService

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Services', 'Prio' => '900' }, 'NavBarName' => 'Admin', 'Title' => 'Services' };</pre>

6.13.15. Frontend::Module###AdminSLA

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSLA'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage Service Level Agreements (SLAs).', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Service Level Agreements', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Service Level Agreements' };</pre>

6.13.16. Frontend::Module###AdminType

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminType'} = { 'Description' => 'Admin', 'Group' => ['admin'] };</pre>

```

],
'NavBarModule' => {
  'Block' => 'Ticket',
  'Description' => 'Create and manage ticket types.',
  'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin',
  'Name' => 'Types',
  'Prio' => '700'
},
'NavBarName' => 'Admin',
'Title' => 'Types'
};

```

6.13.17. Frontend::Module###AdminState

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminState'} = { 'Description' => 'Admin', 'Group' => ['admin'], }, 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket states.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'States', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'States' }; </pre>

6.13.18. Frontend::Module###AdminPriority

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminPriority'} = { 'Description' => 'Admin', 'Group' => ['admin'], }, 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket priorities.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Priorities', 'Prio' => '850' }, 'NavBarName' => 'Admin', 'Title' => 'Priorities' }; </pre>

6.13.19. Frontend::Module###AdminGenericAgent

Description:	Frontend module registration for the agent interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericAgent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'JavaScript' => ['Core.Agent.Admin.GenericAgent.js'] }, 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage tasks triggered by event or time based execution.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'GenericAgent', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'GenericAgent' }; </pre>

6.14. Frontend::Agent

6.14.1. Ticket::Frontend::PendingDiffTime

Description:	Time in seconds that gets added to the actual time if setting a pending-state (default: 86400 = 1 day).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::PendingDiffTime'} = '86400';</code>

6.14.2. Ticket::Frontend::MaxQueueLevel

Description:	Define the max depth of queues.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::MaxQueueLevel'} = '5';</code>

6.14.3. Ticket::Frontend::ListType

Description:	Shows existing parent/child queue lists in the system in the form of a tree or a list.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ListType'} = 'tree';</code>

6.14.4. Ticket::Frontend::TextAreaEmail

Description:	Permitted width for compose email windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaEmail'} = '82';</code>

6.14.5. Ticket::Frontend::TextAreaNote

Description:	Permitted width for compose note windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaNote'} = '78';</code>

6.14.6. Ticket::Frontend::InformAgentMaxSize

Description:	Max size (in rows) of the informed agents box in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::InformAgentMaxSize'} = '3';</code>

6.14.7. Ticket::Frontend::InvolvedAgentMaxSize

Description:	Max size (in rows) of the involved agents box in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::InvolvedAgentMaxSize'} = '3';</code>

6.14.8. Ticket::Frontend::CustomerInfoCompose

Description:	Shows the customer user information (phone and email) in the compose screen.
Group:	Ticket

SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoCompose'} = '1';</code>

6.14.9. Ticket::Frontend::CustomerInfoComposeMaxSize

Description:	Max size (in characters) of the customer information table (phone and email) in the compose screen.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoComposeMaxSize'} = '22';</code>

6.14.10. Ticket::Frontend::CustomerInfoZoom

Description:	Shows the customer user's info in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoom'} = '1';</code>

6.14.11. Ticket::Frontend::CustomerInfoZoomMaxSize

Description:	Maximum size (in characters) of the customer information table in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoomMaxSize'} = '22';</code>

6.14.12. Ticket::Frontend::DynamicFieldsZoomMaxSizeSidebar

Description:	Maximum length (in characters) of the dynamic field in the sidebar of the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::DynamicFieldsZoomMaxSizeSidebar'} = '18';</code>

6.14.13. Ticket::Frontend::DynamicFieldsZoomMaxSizeArticle

Description:	Maximum length (in characters) of the dynamic field in the article of the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::DynamicFieldsZoomMaxSizeArticle'} = '160';</code>

6.14.14. Ticket::Frontend::AccountTime

Description:	Activates time accounting.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AccountTime'} = '1';</code>

6.14.15. Ticket::Frontend::TimeUnits

Description:	Sets the preferred time units (e.g. work units, hours, minutes).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TimeUnits'} = '(work units)';</code>

6.14.16. Ticket::Frontend::NeedAccountedTime

Description:	Defines if time accounting is mandatory in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NeedAccountedTime'} = '0';</code>

6.14.17. Ticket::Frontend::BulkAccountedTime

Description:	Defines if time accounting must be set to all tickets in bulk action.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkAccountedTime'} = '1';</code>
-----------------	--

6.14.18. Ticket::Frontend::NeedSpellCheck

Description:	Defines if composed messages have to be spell checked in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NeedSpellCheck'} = '0';</code>

6.14.19. Ticket::Frontend::NewOwnerSelection

Description:	Shows an owner selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewOwnerSelection'} = '1';</code>

6.14.20. Ticket::Frontend::NewResponsibleSelection

Description:	Show a responsible selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewResponsibleSelection'} = '1';</code>

6.14.21. Ticket::Frontend::NewQueueSelectionType

Description:	Defines the receipt target of the phone ticket and the sender of the email ticket ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewQueueSelectionType'} = 'Queue';</code>

6.14.22. Ticket::Frontend::NewQueueSelectionString

Description:	Determines the strings that will be shown as receipt (To:) of the phone ticket and as sender (From:) of the email ticket in the agent
--------------	---

	interface. For Queue as NewQueueSelectionType "<Queue>" shows the names of the queues and for SystemAddress "<Realname> <<Email>>" shows the name and email of the recipient.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewQueueSelectionString'} = '<Queue>';</code>

6.14.23. Ticket::Frontend::NewQueueOwnSelection

Description:	Determines which options will be valid of the recipient (phone ticket) and the sender (email ticket) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewQueueOwnSelection'} = { '1' => 'First Queue!' '2' => 'Second Queue!' };</code>

6.14.24. Ticket::Frontend::ShowCustomerTickets

Description:	Shows customer history tickets in AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ShowCustomerTickets'} = '1';</code>

6.14.25. NewTicketInNewWindow::Enabled

Description:	If enabled, TicketPhone and TicketEmail will be open in new windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NewTicketInNewWindow::Enabled'} = '0';</code>

6.14.26. CustomerDBLink

Description:	Defines an external link to the database of the customer (e.g. 'http://yourhost/customer.php?CID=\$Data{"CustomerID"}' or "").
Group:	Ticket

SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLink'} = '\$Env{"CGIHandle"}?Action=AgentCustomerInformationCenter;CustomerID=\$LQData{"CustomerID"}';</code>

6.14.27. CustomerDBLinkTarget

Description:	Defines the target attribute in the link to external customer database. E.g. 'target="cdb"'.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLinkTarget'} = '';</code>

6.14.28. CustomerDBLinkClass

Description:	Defines the target attribute in the link to external customer database. E.g. 'AsPopup PopupType_TicketAction'.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLinkClass'} = '';</code>

6.14.29. Frontend::CommonObject###QueueObject

Description:	Path of the file that stores all the settings for the QueueObject object for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue';</code>

6.14.30. Frontend::CommonObject###TicketObject

Description:	Path of the file that stores all the settings for the TicketObject for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonObject'}->{'TicketObject'} = 'Kernel::System::Ticket';</code>

6.14.31. Frontend::CommonParam###Action

Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'Action'} = 'AgentDashboard';</code>

6.14.32. Frontend::CommonParam###QueueID

Description:	Default queue ID used by the system in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'QueueID'} = '0';</code>

6.14.33. Frontend::CommonParam###TicketID

Description:	Default ticket ID used by the system in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'TicketID'} = '';</code>

6.14.34. DefaultOverviewColumns

Description:	General ticket data shown in the ticket overviews (fall-back). Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note that TicketNumber can not be disabled, because it is necessary.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'DefaultOverviewColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '2', 'Owner' => '2', 'PendingTime' => '1', 'Priority' => '1',</pre>

```
'Queue' => '2',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

6.15. Frontend::Agent::Dashboard

6.15.1. DashboardBackend###0100-TicketPendingReminder

Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0100-TicketPendingReminder'} = { 'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending reminder;SortBy=PendingTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }, 'Description' => 'All tickets with a reminder set where the reminder date has been reached', 'Filter' => 'Locked', 'Group' => '', 'Limit' => '10',</pre>

	<pre>'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'UntilTime', 'Title' => 'Reminder Tickets' };</pre>
--	--

6.15.2. DashboardBackend###0110-TicketEscalation

Description:	Parameters for the dashboard backend of the ticket escalation overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0110-TicketEscalation'} = { 'Attributes' => 'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }, 'Description' => 'All escalated tickets', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'EscalationTime', 'Title' => 'Escalated Tickets' };</pre>

6.15.3. DashboardBackend###0120-TicketNew

Description:	Parameters for the dashboard backend of the new tickets overview of the agent interface. "Limit" is the number of entries shown by
--------------	--

	default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0120-TicketNew'} = { 'Attributes' => 'StateType=new;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }, 'Description' => 'All new tickets, these tickets have not been worked on yet', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'New Tickets' }; </pre>

6.15.4. DashboardBackend###0130-TicketOpen

Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket

SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'DashboardBackend'}->{'0130-TicketOpen'} = { 'Attributes' => 'StateType=open;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }, 'Description' => 'All open tickets, these tickets have already been worked on, but need a response', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'Open Tickets / Need to be answered' }; </pre>

6.15.5. DashboardBackend###0250-TicketStats

Description:	Parameters for the dashboard backend of the ticket stats of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'DashboardBackend'}->{'0250-TicketStats'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '30', 'Changed' => '1', 'Closed' => '1', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardTicketStatsGeneric', 'Permission' => 'rw', 'Title' => '7 Day Stats' }; </pre>

```
};
```

6.15.6. DashboardBackend###0260-TicketCalendar

Description:	Parameters for the dashboard backend of the ticket calendar of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0260-TicketCalendar'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '2', 'Default' => '1', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardCalendar', 'OwnerOnly' => '', 'Permission' => 'rw', 'Title' => 'Upcoming Events' };</pre>

6.15.7. DashboardBackend###0270-TicketQueueOverview

Description:	Parameters for the dashboard backend of the queue overview widget of the agent interface. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "QueuePermissionGroup" is not mandatory, queues are only listed if they belong to this permission group if you enable it. "States" is a list of states, the key is the sort order of the state in the widget. "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0270-TicketQueueOverview'} = { 'Block' => 'ContentLarge', 'CacheTTLocal' => '0.5', 'Default' => '1', 'Description' => 'Provides a matrix overview of the tickets per state per queue.', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardTicketQueueOverview', 'Permission' => 'rw', 'QueuePermissionGroup' => 'users', 'Sort' => 'SortBy=Age;OrderBy=Up', 'States' => { '1' => 'new', '4' => 'open', '6' => 'pending reminder' }, };</pre>

	'Title' => 'Ticket Queue Overview' };
--	--

6.15.8. DashboardBackend###0280-DashboardEventsTicketCalendar

Description:	Parameters for the dashboard backend of the ticket stats of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0280-DashboardEventsTicketCalendar'} = { 'Block' => 'ContentLarge', 'CacheTTL' => '0', 'Default' => '0', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardEventsTicketCalendar', 'Title' => 'Events Ticket Calendar' };</pre>

6.15.9. AgentCustomerInformationCenter::Backend###0100-CIC-TicketPendingReminder

Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AgentCustomerInformationCenter::Backend'}->{'0100-CIC- TicketPendingReminder'} = { 'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending reminder;SortBy=PendingTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1',</pre>

```

'CustomerUserID' => '1',
'EscalationResponseTime' => '1',
'EscalationSolutionTime' => '1',
'EscalationTime' => '1',
'EscalationUpdateTime' => '1',
'Lock' => '1',
'Owner' => '1',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '1',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '1',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
},
'Description' => 'All tickets with a reminder set where the
reminder date has been reached',
'Filter' => 'Locked',
'Group' => '',
'Limit' => '10',
'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric',
'Permission' => 'ro',
'Time' => 'UntilTime',
'Title' => 'Reminder Tickets'
};

```

6.15.10. AgentCustomerInformationCenter::Backend###0110-CIC-TicketEscalation

Description:	Parameters for the dashboard backend of the ticket escalation overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'AgentCustomerInformationCenter::Backend'}->{'0110-CIC-TicketEscalation'} = { 'Attributes' => 'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', </pre>

```

'Owner' => '1',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '1',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '1',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
},
'Description' => 'All escalated tickets',
'Filter' => 'All',
'Group' => '',
'Limit' => '10',
'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric',
'Permission' => 'ro',
'Time' => 'EscalationTime',
'Title' => 'Escalated Tickets'
};

```

6.15.11. AgentCustomerInformationCenter::Backend###0120-CIC-TicketNew

Description:	Parameters for the dashboard backend of the new tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'AgentCustomerInformationCenter::Backend'}->{'0120-CIC-TicketNew'} = { 'Attributes' => 'StateType=new;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', </pre>

```

    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All new tickets, these tickets have not been
worked on yet',
  'Filter' => 'All',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric',
  'Permission' => 'ro',
  'Time' => 'Age',
  'Title' => 'New Tickets'
};

```

6.15.12. AgentCustomerInformationCenter::Backend###0130-CIC-TicketOpen

Description:	Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'AgentCustomerInformationCenter::Backend'}->{'0130-CIC-TicketOpen'} = { 'Attributes' => 'StateType=open;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'DefaultColumns' => { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '1', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '1', 'Owner' => '1', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '1', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }, 'Description' => 'All open tickets, these tickets have already been worked on, but need a response', </pre>

	<pre>'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'ro', 'Time' => 'Age', 'Title' => 'Open Tickets / Need to be answered' };</pre>
--	--

6.15.13. AgentCustomerInformationCenter::Backend###0500-CIC-CustomerIDStatus

Description:	Parameters for the dashboard backend of the customer id status widget of the agent interface . "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'AgentCustomerInformationCenter::Backend'}->{'0500-CIC-CustomerIDStatus'} = { 'Attributes' => '', 'Block' => 'ContentSmall', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'Company Status', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardCustomerIDStatus', 'Permission' => 'ro', 'Title' => 'Company Status' };</pre>

6.16. Frontend::Agent::Dashboard::EventsTicketCalendar

6.16.1. DashboardEventsTicketCalendar###CalendarWidth

Description:	Defines the calendar width in percent. Default is 95%.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar'}->{'CalendarWidth'} = '95';</pre>

6.16.2. DashboardEventsTicketCalendar###Queues

Description:	Defines queues that's tickets are used for displaying as calendar events.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar'}->{'Queues'} = ['Raw'];</pre>

6.16.3.

DashboardEventsTicketCalendar::DynamicFieldStartTime

Description:	Define dynamic field name for start time. This field has to be manually added to the system as Ticket: "Date / Time" and must be activated in ticket creation screens and/or in any other ticket action screens.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar::DynamicFieldStartTime'} = 'TicketCalendarStartTime';</pre>

6.16.4.

DashboardEventsTicketCalendar::DynamicFieldEndTime

Description:	Define dynamic field name for end time. This field has to be manually added to the system as Ticket: "Date / Time" and must be activated in ticket creation screens and/or in any other ticket action screens.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar::DynamicFieldEndTime'} = 'TicketCalendarEndTime';</pre>

6.16.5.

DashboardEventsTicketCalendar::DynamicFieldsForEvents

Description:	Defines the dynamic fields that are used for displaying on calendar events.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar::DynamicFieldsForEvents'} = ['TicketCalendarStartTime', 'TicketCalendarEndTime'];</pre>

6.16.6.

DashboardEventsTicketCalendar::TicketFieldsForEvents

Description:	Defines the ticket fields that are going to be displayed calendar events. The "Key" defines the field or ticket attribute and the "Content" defines the display name.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::EventsTicketCalendar
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'DashboardEventsTicketCalendar::TicketFieldsForEvents'} = { 'CustomerID' => 'Customer ID', 'CustomerUserID' => 'Customer user', 'Priority' => 'Priority', 'Queue' => 'Queue', 'SLA' => 'SLA', 'Service' => 'Service', 'State' => 'State', 'Title' => 'Title', 'Type' => 'Type' };</pre>

6.17.

Frontend::Agent::Dashboard::TicketFilters

6.17.1. OnlyValuesOnTicket

Description:	Defines if the list for filters should be retrieve just from current tickets in system. Just for clarification, Customers list will always came from system's tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard::TicketFilters
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'OnlyValuesOnTicket'} = '1';</pre>

6.18. Frontend::Agent::ModuleMetaHead

6.18.1. Frontend::HeaderMetaModule###2-TicketSearch

Description:	Module to generate html OpenSearch profile for short ticket search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'2-TicketSearch'} = { 'Action' => 'AgentTicketSearch', 'Module' => 'Kernel::Output::HTML::HeaderMetaTicketSearch'</pre>

```
};
```

6.19. Frontend::Agent::ModuleNotify

6.19.1. Frontend::NotifyModule###5-Ticket::TicketEscalation

Description:	Module to show notifications and escalations (ShownMax: max. shown escalations, EscalationInMinutes: Show ticket which will escalation in, CacheTime: Cache of calculated escalations in seconds).
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'5-Ticket::TicketEscalation'} = { 'CacheTime' => '40', 'EscalationInMinutes' => '120', 'Module' => 'Kernel::Output::HTML::NotificationAgentTicketEscalation', 'ShownMax' => '25' };</pre>

6.20. Frontend::Agent::ModuleRegistration

6.20.1. Frontend::Module###AgentTicketQueue

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketQueue'} = { 'Description' => 'Overview of all open Tickets', 'Loader' => { 'CSS' => ['Core.AgentTicketQueue.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBar' => [{ 'AccessKey' => 'o', 'Block' => '', 'Description' => 'Overview of all open Tickets', 'Link' => 'Action=AgentTicketQueue', 'LinkOption' => '', 'Name' => 'Queue view', 'NavBar' => 'Ticket', 'Prio' => '100', 'Type' => '' }, { 'AccessKey' => 't',</pre>

```

'Block' => 'ItemArea',
'Description' => '',
'Link' => 'Action=AgentTicketQueue',
'LinkOption' => '',
'Name' => 'Tickets',
'NavBar' => 'Ticket',
'Prio' => '200',
'Type' => 'Menu'
}
],
'NavBarName' => 'Ticket',
'Title' => 'QueueView'
};

```

6.20.2. Frontend::Module###AgentTicketPhone

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketPhone'} = { 'Description' => 'Create new phone ticket', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'n', 'Block' => '', 'Description' => 'Create new phone ticket (inbound)', 'Link' => 'Action=AgentTicketPhone', 'LinkOption' => '', 'Name' => 'New phone ticket', 'NavBar' => 'Ticket', 'Prio' => '200', 'Type' => '' }] }, 'NavBarName' => 'Ticket', 'Title' => 'New phone ticket' }; </pre>

6.20.3. Frontend::Module###AgentTicketPhoneOutbound

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketPhoneOutbound'} = { 'Description' => 'Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', }; </pre>

	<pre>'Title' => 'Phone-Ticket' };</pre>
--	--

6.20.4. Frontend::Module###AgentTicketPhoneInbound

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPhoneInbound'} = { 'Description' => 'Incoming Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Phone-Ticket' };</pre>

6.20.5. Frontend::Module###AgentTicketEmail

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketEmail'} = { 'Description' => 'Create new email ticket', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'm', 'Block' => '', 'Description' => 'Create new email ticket and send this out (outbound)', 'Link' => 'Action=AgentTicketEmail', 'LinkOption' => '', 'Name' => 'New email ticket', 'NavBar' => 'Ticket', 'Prio' => '210', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'New email ticket' };</pre>

6.20.6. Frontend::Module###AgentTicketSearch

Description:	Frontend module registration for the agent interface.
Group:	Ticket

SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketSearch'} = { 'Description' => 'Search Ticket', 'Loader' => { 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBar' => [{ 'AccessKey' => 's', 'Block' => '', 'Description' => 'Search Tickets', 'Link' => 'Action=AgentTicketSearch', 'LinkOption' => 'onclick="window.setTimeout(function() (Core.Agent.Search.OpenSearchDialog(\'AgentTicketSearch\'));}, 0); return false;"', 'Name' => 'Search', 'NavBar' => 'Ticket', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Search' }; </pre>

6.20.7. Frontend::Module###AgentTicketLockedView

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketLockedView'} = { 'Description' => 'Locked Tickets', 'Loader' => { 'CSS' => ['Core.AgentTicketQueue.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Locked Tickets' }; </pre>

6.20.8. Frontend::Module###AgentTicketResponsibleView

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1

Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketResponsibleView'} = { 'Description' => 'Responsible Tickets', 'Loader' => { 'CSS' => ['Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Responsible Tickets' }; </pre>

6.20.9. Frontend::Module###AgentTicketWatchView

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketWatchView'} = { 'Description' => 'Watched Tickets', 'Loader' => { 'CSS' => ['Core.AgentTicketQueue.css', 'Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Watched Tickets' }; </pre>

6.20.10. Frontend::Module###AgentCustomerSearch

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentCustomerSearch'} = { 'Description' => 'AgentCustomerSearch', 'NavBarName' => 'Ticket', 'Title' => 'AgentCustomerSearch' }; </pre>

6.20.11. Frontend::Module###AgentUserSearch

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1

Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentUserSearch'} = { 'Description' => 'AgentUserSearch', 'NavBarName' => 'Ticket', 'Title' => 'AgentUserSearch' };</pre>

6.20.12. Frontend::Module###AgentTicketStatusView

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketStatusView'} = { 'Description' => 'Overview of all open tickets', 'Loader' => { 'CSS' => ['Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, 'NavBar' => [{ 'AccessKey' => 'v', 'Block' => '', 'Description' => 'Overview of all open Tickets.', 'Link' => 'Action=AgentTicketStatusView', 'LinkOption' => '', 'Name' => 'Status view', 'NavBar' => 'Ticket', 'Prio' => '110', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Status view' };</pre>

6.20.13. Frontend::Module###AgentTicketEscalationView

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketEscalationView'} = { 'Description' => 'Overview of all escalated tickets', 'Loader' => { 'CSS' => ['Core.AllocationList.css'], 'JavaScript' => ['Core.UI.AllocationList.js', 'Core.Agent.TableFilters.js'] }, };</pre>

	<pre>'NavBar' => [{ 'AccessKey' => 'e', 'Block' => '', 'Description' => 'Overview Escalated Tickets', 'Link' => 'Action=AgentTicketEscalationView', 'LinkOption' => '', 'Name' => 'Escalation view', 'NavBar' => 'Ticket', 'Prio' => '120', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Escalation view' };</pre>
--	--

6.20.14. Frontend::Module###AgentZoom

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentZoom'} = { 'Description' => 'compat module for AgentZoom to AgentTicketZoom', 'NavBarName' => 'Ticket', 'Title' => '' };</pre>

6.20.15. Frontend::Module###AgentTicketZoom

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketZoom'} = { 'Description' => 'Ticket Zoom', 'Loader' => { 'CSS' => ['Core.Agent.TicketProcess.css'], 'JavaScript' => ['thirdparty/jquery-tablesorter-2.0.5/jquery.tablesorter.js', 'Core.UI.Table.Sort.js', 'Core.Agent.TicketZoom.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' };</pre>

6.20.16. Frontend::Module###AgentTicketAttachment

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1

Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketAttachment'} = { 'Description' => 'To download attachments', 'NavBarName' => 'Ticket', 'Title' => '' };</pre>

6.20.17. Frontend::Module###AgentTicketPlain

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPlain'} = { 'Description' => 'Ticket plain view of an email', 'NavBarName' => 'Ticket', 'Title' => 'Plain' };</pre>

6.20.18. Frontend::Module###AgentTicketNote

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketNote'} = { 'Description' => 'Ticket Note', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Note' };</pre>

6.20.19. Frontend::Module###AgentTicketMerge

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketMerge'} = { 'Description' => 'Ticket Merge', 'NavBarName' => 'Ticket', 'Title' => 'Merge' };</pre>

6.20.20. Frontend::Module###AgentTicketPending

Description:	Frontend module registration for the agent interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPending'} = { 'Description' => 'Ticket Pending', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Pending' };</pre>

6.20.21. Frontend::Module###AgentTicketWatcher

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketWatcher'} = { 'Description' => 'A TicketWatcher Module', 'NavBarName' => 'Ticket-Watcher', 'Title' => 'Ticket-Watcher' };</pre>

6.20.22. Frontend::Module###AgentTicketPriority

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPriority'} = { 'Description' => 'Ticket Priority', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Priority' };</pre>

6.20.23. Frontend::Module###AgentTicketLock

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketLock'} = { 'Description' => 'Ticket Lock',</pre>

	<pre>'NavBarName' => 'Ticket', 'Title' => 'Lock' };</pre>
--	---

6.20.24. Frontend::Module###AgentTicketMove

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketMove'} = { 'Description' => 'Ticket Move', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Move' };</pre>

6.20.25. Frontend::Module###AgentTicketHistory

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketHistory'} = { 'Description' => 'Ticket History', 'NavBarName' => 'Ticket', 'Title' => 'History' };</pre>

6.20.26. Frontend::Module###AgentTicketOwner

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketOwner'} = { 'Description' => 'Ticket Owner', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Owner' };</pre>

6.20.27. Frontend::Module###AgentTicketResponsible

Description:	Frontend module registration for the agent interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketResponsible'} = { 'Description' => 'Ticket Responsible', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Responsible' }; </pre>

6.20.28. Frontend::Module###AgentTicketCompose

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketCompose'} = { 'Description' => 'Ticket Compose email Answer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Compose' }; </pre>

6.20.29. Frontend::Module###AgentTicketBounce

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketBounce'} = { 'Description' => 'Ticket Compose Bounce Email', 'NavBarName' => 'Ticket', 'Title' => 'Bounce' }; </pre>

6.20.30. Frontend::Module###AgentTicketForward

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketForward'} = { 'Description' => 'Ticket Forward Email', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Forward' };</pre>
-----------------	---

6.20.31. Frontend::Module###AgentTicketCustomer

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketCustomer'} = { 'Description' => 'Ticket Customer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Customer' };</pre>

6.20.32. Frontend::Module###AgentTicketClose

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketClose'} = { 'Description' => 'Ticket Close', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Close' };</pre>

6.20.33. Frontend::Module###AgentTicketFreeText

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketFreeText'} = {</pre>

	<pre>'Description' => 'Ticket FreeText', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Free Fields' };</pre>
--	---

6.20.34. Frontend::Module###AgentTicketPrint

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPrint'} = { 'Description' => 'Ticket Print', 'NavBarName' => 'Ticket', 'Title' => 'Print' };</pre>

6.20.35. Frontend::Module###AgentTicketBulk

Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketBulk'} = { 'Description' => 'Ticket bulk module', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Bulk-Action' };</pre>

6.21. Frontend::Agent::Preferences

6.21.1. PreferencesGroups###NewTicketNotify

Description:	Parameters for the NewTicketNotify object in the preferences view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'NewTicketNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => {</pre>

	<pre> '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if there is a new ticket in "My Queues".', 'Key' => 'Send new ticket notifications', 'Label' => 'New ticket notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendNewTicketNotification', 'Prio' => '1000' }; </pre>
--	--

6.21.2. PreferencesGroups###FollowUpNotify

Description:	Parameters for the FollowUpNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'FollowUpNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a customer sends a follow up and I\'m the owner of the ticket or the ticket is unlocked and is in one of my subscribed queues.', 'Key' => 'Send ticket follow up notifications', 'Label' => 'Ticket follow up notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendFollowUpNotification', 'Prio' => '2000' }; </pre>

6.21.3. PreferencesGroups###LockTimeoutNotify

Description:	Parameters for the LockTimeoutNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'LockTimeoutNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is unlocked by the system.', 'Key' => 'Send ticket lock timeout notifications', 'Label' => 'Ticket lock timeout notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendLockTimeoutNotification', 'Prio' => '3000' }; </pre>

```
};
```

6.21.4. PreferencesGroups###MoveNotify

Description:	Parameters for the MoveNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'MoveNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is moved into one of "My Queues".', 'Key' => 'Send ticket move notifications', 'Label' => 'Ticket move notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendMoveNotification', 'Prio' => '4000' };</pre>

6.21.5. PreferencesGroups###WatcherNotify

Description:	Parameters for the WatcherNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'WatcherNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me the same notifications for my watched tickets that the ticket owners will get.', 'Key' => 'Send ticket watch notifications', 'Label' => 'Ticket watch notification', 'Module' => 'Kernel::Output::HTML::PreferencesTicketWatcher', 'PrefKey' => 'UserSendWatcherNotification', 'Prio' => '5000' };</pre>

6.21.6. PreferencesGroups###CustomQueue

Description:	Parameters for the CustomQueue object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'CustomQueue'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Desc' => 'Your queue selection of your favorite queues. You also get notified about those queues via email if enabled.', 'Key' => 'My Queues', 'Label' => 'My Queues', 'Module' => 'Kernel::Output::HTML::PreferencesCustomQueue', 'Permission' => 'ro', 'Prio' => '1000' };</pre>

6.21.7. PreferencesGroups###RefreshTime

Description:	Parameters for the RefreshTime object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'RefreshTime'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { '0' => 'off', '10' => '10 minutes', '15' => '15 minutes', '2' => ' 2 minutes', '5' => ' 5 minutes', '7' => ' 7 minutes' }, 'DataSelected' => '0', 'Desc' => 'If enabled, the different overviews (Dashboard, LockedView, QueueView) will automatically refresh after the specified time.', 'Key' => 'Refresh Overviews after', 'Label' => 'Overview Refresh Time', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '2000' };</pre>

6.21.8. PreferencesGroups###TicketOverviewSmallPageShown

Description:	Parameters for the pages (in which the tickets are shown) of the small ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TicketOverviewSmallPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', </pre>

	<pre>'20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '25', 'Key' => 'Ticket limit per page for Ticket Overview "Small"', 'Label' => 'Ticket Overview "Small" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewSmallPageShown', 'Prio' => '8000' };</pre>
--	---

6.21.9. PreferencesGroups###TicketOverviewFilterSettings

Description:	Parameters for .
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TicketOverviewFilterSettings'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Key' => 'Column ticket filters for Ticket Overviews type "Small".', 'Label' => 'Enabled filters.', 'Module' => 'Kernel::Output::HTML::PreferencesColumnFilters', 'PrefKey' => 'UserFilterColumnsEnabled', 'Prio' => '8100' };</pre>

6.21.10. PreferencesGroups###TicketOverviewMediumPageShown

Description:	Parameters for the pages (in which the tickets are shown) of the medium ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TicketOverviewMediumPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '20', 'Key' => 'Ticket limit per page for Ticket Overview "Medium"', 'Label' => 'Ticket Overview "Medium" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewMediumPageShown', 'Prio' => '8100' };</pre>

6.21.11. PreferencesGroups###TicketOverviewPreviewPageShown

Description:	Parameters for the pages (in which the tickets are shown) of the ticket preview overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TicketOverviewPreviewPageShown'} = { 'Active' => '0', 'Column' => 'Other Settings', 'Data' => { '10' => '10', '15' => '15', '20' => '20', '25' => '25', '30' => '30', '35' => '35' }, 'DataSelected' => '15', 'Key' => 'Ticket limit per page for Ticket Overview "Preview"', 'Label' => 'Ticket Overview "Preview" Limit', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewPreviewPageShown', 'Prio' => '8200' };</pre>

6.21.12. PreferencesGroups###CreateNextMask

Description:	Parameters for the CreateNextMask object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'CreateNextMask'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { '' => 'CreateTicket', 'AgentTicketZoom' => 'TicketZoom' }, 'DataSelected' => '', 'Key' => 'Show this screen after I created a new ticket', 'Label' => 'Screen after new ticket', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserCreateNextMask', 'Prio' => '3000' };</pre>

6.22. Frontend::Agent::SearchRouter

6.22.1.

Frontend::Search###AgentCustomerInformationCenter

Description:	Search backend router.
--------------	------------------------

Group:	Ticket
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Search'}->{'AgentCustomerInformationCenter'} = { '^AgentCustomerInformationCenter' => 'Action=AgentCustomerInformationCenterSearch' };</pre>

6.22.2. Frontend::Search###Ticket

Description:	Search backend router.
Group:	Ticket
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Search'}->{'Ticket'} = { '^AgentTicket' => 'Action=AgentTicketSearch;Subaction=AJAX' };</pre>

6.23. Frontend::Agent::Ticket::ArticleAttachmentModule

6.23.1. Ticket::Frontend::ArticleAttachmentModule###1-Download

Description:	Shows a link to download article attachments in the zoom view of the article in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'1-Download'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentDownload' };</pre>

6.23.2. Ticket::Frontend::ArticleAttachmentModule###2-HTML-Viewer

Description:	Shows a link to access article attachments via a html online viewer in the zoom view of the article in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1

Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'2-HTML-Viewer'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentHTMLViewer' };</pre>

6.24.

Frontend::Agent::Ticket::ArticleComposeModule

6.24.1. Ticket::Frontend::ArticleComposeModule###1-SignEmail

Description:	Module to compose signed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'1-SignEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeSign' };</pre>

6.24.2. Ticket::Frontend::ArticleComposeModule###2-CryptEmail

Description:	Module to crypt composed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'2-CryptEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeCrypt' };</pre>

6.25.

Frontend::Agent::Ticket::ArticleViewModule

6.25.1. Ticket::Frontend::ArticleViewModule###1-PGP

Description:	Agent interface article notification module to check PGP.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' };</pre>

6.25.2. Ticket::Frontend::ArticleViewModule###1-SMIME

Description:	Agent interface module to check incoming emails in the Ticket-Zoom-View if the S/MIME-key is available and true.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' };</pre>

6.26. Frontend::Agent::Ticket::ArticleViewModulePre

6.26.1. Ticket::Frontend::ArticlePreViewModule###1-PGP

Description:	Agent interface article notification module to check PGP.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' };</pre>

6.26.2. Ticket::Frontend::ArticlePreViewModule###1-SMIME

Description:	Agent interface article notification module to check S/MIME.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' };</pre>

6.27. Frontend::Agent::Ticket::MenuModule

6.27.1. Ticket::Frontend::MenuModule###000-Back

Description:	Shows a link in the menu to go back in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule

Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'000-Back'} = { 'Action' => '', 'Description' => 'Back', 'Link' => '\$Env{"LastScreenOverview"};TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Back', 'PopupType' => '', 'Target' => '' };</pre>

6.27.2. Ticket::Frontend::MenuModule###100-Lock

Description:	Shows a link in the menu to lock/unlock tickets in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'Target' => '' };</pre>

6.27.3. Ticket::Frontend::MenuModule###200-History

Description:	Shows a link in the menu to access the history of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'200-History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Show the ticket history', 'Link' => 'Action=AgentTicketHistory;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' };</pre>

6.27.4. Ticket::Frontend::MenuModule###210-Print

Description:	Shows a link in the menu to print a ticket or an article in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'210-Print'} = {</pre>

	<pre>'Action' => 'AgentTicketPrint', 'Description' => 'Print this ticket', 'Link' => 'Action=AgentTicketPrint;TicketID=\$QData{"TicketID"}', 'LinkParam' => 'target="print"', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Print', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>
--	--

6.27.5. Ticket::Frontend::MenuModule###300-Priority

Description:	Shows a link in the menu to see the priority of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'300-Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the ticket priority', 'Link' => 'Action=AgentTicketPriority;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.27.6. Ticket::Frontend::MenuModule###310-FreeText

Description:	Shows a link in the menu to add a free text field in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'310-FreeText'} = { 'Action' => 'AgentTicketFreeText', 'Description' => 'Change the free fields for this ticket', 'Link' => 'Action=AgentTicketFreeText;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Free Fields', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.27.7. Ticket::Frontend::MenuModule###320-Link

Description:	Shows a link in the menu that allows linking a ticket with another object in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'320-Link'} = { 'Action' => 'AgentLinkObject', 'Description' => 'Link this ticket to other objects', 'Link' => 'Action=AgentLinkObject;SourceObject=Ticket;SourceKey= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Link', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>
-----------------	--

6.27.8. Ticket::Frontend::MenuModule###400-Owner

Description:	Shows a link in the menu to see the owner of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'400-Owner'} = { 'Action' => 'AgentTicketOwner', 'Description' => 'Change the owner for this ticket', 'Link' => 'Action=AgentTicketOwner;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Owner', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.27.9. Ticket::Frontend::MenuModule###410-Responsible

Description:	Shows a link in the menu to see the responsible agent of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'410-Responsible'} = { 'Action' => 'AgentTicketResponsible', 'Description' => 'Change the responsible person for this ticket', 'Link' => 'Action=AgentTicketResponsible;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuResponsible', 'Name' => 'Responsible', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.27.10. Ticket::Frontend::MenuModule###420-Customer

Description:	Shows a link in the menu to see the customer who requested the ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule

Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'420-Customer'} = { 'Action' => 'AgentTicketCustomer', 'Description' => 'Change the customer for this ticket', 'Link' => 'Action=AgentTicketCustomer;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Customer', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.11. Ticket::Frontend::MenuModule###420-Note

Description:	Shows a link in the menu to add a note in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'420-Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.12. Ticket::Frontend::MenuModule###425-Phone Call Outbound

Description:	Shows a link in the menu to add a note in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'425-Phone Call Outbound'} = { 'Action' => 'AgentTicketPhoneOutbound', 'Description' => 'Phone Call Outbound', 'Link' => 'Action=AgentTicketPhoneOutbound;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Phone Call Outbound', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.13. Ticket::Frontend::MenuModule###426-Phone Call Inbound

Description:	Shows a link in the menu to add a note in the ticket zoom view of the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'426-Phone Call Inbound'} = { 'Action' => 'AgentTicketPhoneInbound', 'Description' => 'Phone Call Inbound', 'Link' => 'Action=AgentTicketPhoneInbound;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Phone Call Inbound', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.14. Ticket::Frontend::MenuModule###430-Merge

Description:	Shows a link in the menu that allows merging tickets in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'430-Merge'} = { 'Action' => 'AgentTicketMerge', 'Description' => 'Merge into a different ticket', 'Link' => 'Action=AgentTicketMerge;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Merge', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.15. Ticket::Frontend::MenuModule###440-Pending

Description:	Shows a link in the menu to set a ticket as pending in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'440-Pending'} = { 'Action' => 'AgentTicketPending', 'Description' => 'Set this ticket to pending', 'Link' => 'Action=AgentTicketPending;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Pending', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

6.27.16. Ticket::Frontend::MenuModule###448-Watch

Description:	Shows a link in the menu for subscribing / unsubscribing from a ticket in the ticket zoom view of the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'448-Watch'} = { 'Action' => 'AgentTicketWatcher', 'Module' => 'Kernel::Output::HTML::TicketMenuTicketWatcher', 'Name' => 'Watch', 'Target' => '' };</pre>

6.27.17. Ticket::Frontend::MenuModule###450-Close

Description:	Shows a link in the menu to close a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'450-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.27.18. Ticket::Frontend::MenuModule###460-Delete

Description:	Shows a link in the menu to delete a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'460-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' };</pre>

6.27.19. Ticket::Frontend::MenuModule###470-Spam

Description:	Shows a link to set a ticket as spam in the ticket zoom view of the agent interface. Additional access control to show or not
--------------	---

	show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'470-Spam'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID= \$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' }; </pre>

6.28. Frontend::Agent::Ticket::MenuModulePre

6.28.1. Ticket::Frontend::PreMenuModule###100-Lock

Description:	Shows a link in the menu to lock / unlock a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'PopupType' => '', 'Target' => '' }; </pre>

6.28.2. Ticket::Frontend::PreMenuModule###200-Zoom

Description:	Shows a link in the menu to zoom a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'200-Zoom'} = { 'Action' => 'AgentTicketZoom', 'Description' => 'Look into a ticket!', 'Link' => 'Action=AgentTicketZoom;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Zoom', 'PopupType' => '', 'Target' => '' }; </pre>

6.28.3. Ticket::Frontend::PreMenuModule###210-History

Description:	Shows a link in the menu to see the history of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'210-History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Show the ticket history', 'Link' => 'Action=AgentTicketHistory;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' };</pre>

6.28.4. Ticket::Frontend::PreMenuModule###300-Priority

Description:	Shows a link in the menu to set the priority of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'300-Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the priority for this ticket', 'Link' => 'Action=AgentTicketPriority;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.28.5. Ticket::Frontend::PreMenuModule###420-Note

Description:	Shows a link in the menu to add a note to a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'420-Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.28.6. Ticket::Frontend::PreMenuModule###440-Close

Description:	Shows a link in the menu to close a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'440-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

6.28.7. Ticket::Frontend::PreMenuModule###445-Move

Description:	Shows a link in the menu to move a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'445-Move'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Change queue!', 'Module' => 'Kernel::Output::HTML::TicketMenuMove', 'Name' => 'Move' };</pre>

6.28.8. Ticket::Frontend::PreMenuModule###450-Delete

Description:	Shows a link in the menu to delete a ticket in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'450-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' };</pre>

```
};
```

6.28.9. Ticket::Frontend::PreMenuModule###460-Spam

Description:	Shows a link in the menu to set a ticket as spam in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'460-Spam'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID= \$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' };</pre>

6.29. Frontend::Agent::Ticket::OverviewMenuModule

6.29.1. Ticket::Frontend::OverviewMenuModule###001-Sort

Description:	Shows a select of ticket attributes to order the queue view ticket list. The possible selections can be configured via 'TicketOverviewMenuSort###SortAttributes'.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::OverviewMenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewMenuModule'}->{'001-Sort'} = { 'Module' => 'Kernel::Output::HTML::TicketOverviewMenuSort' };</pre>

6.29.2. TicketOverviewMenuSort###SortAttributes

Description:	Defines from which ticket attributes the agent can select the result order.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::OverviewMenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'TicketOverviewMenuSort'}->{'SortAttributes'} = { 'Age' => '1', 'Title' => '1' };</pre>

6.30. Frontend::Agent::Ticket::ViewBounce

6.30.1.

Ticket::Frontend::AgentTicketBounce###Permission

Description:	Required permissions to use the ticket bounce screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'Permission'} = 'bounce';</code>

6.30.2.

Ticket::Frontend::AgentTicketBounce###RequiredLock

Description:	Defines if a ticket lock is required in the ticket bounce screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'RequiredLock'} = '1';</code>

6.30.3.

Ticket::Frontend::AgentTicketBounce###StateDefault

Description:	Defines the default next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateDefault'} = 'closed successful';</code>

6.30.4.

Ticket::Frontend::AgentTicketBounce###StateType

Description:	Defines the next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1

Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateType'} = ['open', 'closed'];</pre>
-----------------	---

6.30.5. Ticket::Frontend::BounceText

Description:	Defines the default ticket bounced notification for customer/sender in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::BounceText'} = 'Your email with ticket number "<OTRS_TICKET>" is bounced to "<OTRS_BOUNCE_TO>". Contact this address for further information.'</pre>

6.31. Frontend::Agent::Ticket::ViewBulk

6.31.1.

Ticket::Frontend::AgentTicketBulk###RequiredLock

Description:	Automatically lock and set owner to current Agent after selecting for an Bulk Action.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'RequiredLock'} = '1';</pre>

6.31.2.

Ticket::Frontend::AgentTicketBulk###TicketType

Description:	Sets the ticket type in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'TicketType'} = '1';</pre>

6.31.3. Ticket::Frontend::AgentTicketBulk###Owner

Description:	Sets the ticket owner in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Owner'} = '1';</code>
-----------------	--

6.31.4. Ticket::Frontend::AgentTicketBulk###Responsible

Description:	Sets the responsible agent of the ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Responsible'} = '1';</code>

6.31.5. Ticket::Frontend::AgentTicketBulk###State

Description:	If a note is added by an agent, sets the state of a ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'State'} = '1';</code>

6.31.6. Ticket::Frontend::AgentTicketBulk###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

6.31.7. Ticket::Frontend::AgentTicketBulk###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateDefault'} = 'open';</code>
-----------------	--

6.31.8. Ticket::Frontend::AgentTicketBulk###Priority

Description:	Shows the ticket priority options in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Priority'} = '1';</code>

6.31.9.

Ticket::Frontend::AgentTicketBulk###PriorityDefault

Description:	Defines the default ticket priority in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'PriorityDefault'} = '3 normal';</code>

6.31.10.

Ticket::Frontend::AgentTicketBulk###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.31.11.

Ticket::Frontend::AgentTicketBulk###ArticleTypes

Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0'</code>

```
};
```

6.32. Frontend::Agent::Ticket::ViewClose

6.32.1.

Ticket::Frontend::AgentTicketClose###Permission

Description:	Required permissions to use the close ticket screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Permission'} = 'close';</code>

6.32.2.

Ticket::Frontend::AgentTicketClose###RequiredLock

Description:	Defines if a ticket lock is required in the close ticket screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RequiredLock'} = '1';</code>

6.32.3.

Ticket::Frontend::AgentTicketClose###TicketType

Description:	Sets the ticket type in the close ticket screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'TicketType'} = '0';</code>

6.32.4. Ticket::Frontend::AgentTicketClose###Service

Description:	Sets the service in the close ticket screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Service'} = '0';</code>
-----------------	---

6.32.5.

Ticket::Frontend::AgentTicketClose###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ServiceMandatory'} = '0';</code>

6.32.6.

Ticket::Frontend::AgentTicketClose###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'SLAMandatory'} = '0';</code>

6.32.7. Ticket::Frontend::AgentTicketClose###Queue

Description:	Sets the queue in the ticket close screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Queue'} = '0';</code>

6.32.8. Ticket::Frontend::AgentTicketClose###Owner

Description:	Sets the ticket owner in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Owner'} = '0';</code>

6.32.9.

Ticket::Frontend::AgentTicketClose###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'OwnerMandatory'} = '0';</code>

6.32.10. Ticket::Frontend::AgentTicketClose###Responsible

Description:	Sets the responsible agent of the ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Responsible'} = '0';</code>

6.32.11. Ticket::Frontend::AgentTicketClose###State

Description:	If a note is added by an agent, sets the state of a ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'State'} = '1';</code>

6.32.12. Ticket::Frontend::AgentTicketClose###StateType

Description:	Defines the next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateType'} = ['closed'];</code>

6.32.13. Ticket::Frontend::AgentTicketClose###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateDefault'} = 'closed successful';</code>

6.32.14. Ticket::Frontend::AgentTicketClose###Note

Description:	Allows adding notes in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Note'} = '1';</code>

6.32.15.

Ticket::Frontend::AgentTicketClose###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'NoteMandatory'} = '1';</code>

6.32.16.

Ticket::Frontend::AgentTicketClose###Subject

Description:	Sets the default subject for notes added in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Subject'} = '\$Text{"Close"}';</code>

6.32.17. Ticket::Frontend::AgentTicketClose###Body

Description:	Sets the default body text for notes added in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Body'} = '';</code>

6.32.18.

Ticket::Frontend::AgentTicketClose###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InvolvedAgent'} = '0';</code>

6.32.19.

Ticket::Frontend::AgentTicketClose###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InformAgent'} = '0';</code>

6.32.20.

Ticket::Frontend::AgentTicketClose###ArticleTypeDefault

Description:	Defines the default type of the note in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.32.21.

Ticket::Frontend::AgentTicketClose###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>
-----------------	--

6.32.22.

Ticket::Frontend::AgentTicketClose###Priority

Description:	Shows the ticket priority options in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Priority'} = '0';</pre>

6.32.23.

Ticket::Frontend::AgentTicketClose###PriorityDefault

Description:	Defines the default ticket priority in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'PriorityDefault'} = '3 normal';</pre>

6.32.24. Ticket::Frontend::AgentTicketClose###Title

Description:	Shows the title fields in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Title'} = '0';</pre>

6.32.25.

Ticket::Frontend::AgentTicketClose###HistoryType

Description:	Defines the history type for the close ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryType'} = 'AddNote';</code>
-----------------	---

6.32.26.

Ticket::Frontend::AgentTicketClose###HistoryComment

Description:	Defines the history comment for the close ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryComment'} = '%Close';</code>

6.32.27.

Ticket::Frontend::AgentTicketClose###DynamicField

Description:	Dynamic fields shown in the ticket close screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'DynamicField'} = {};</code>

6.32.28.

Ticket::Frontend::AgentTicketClose###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RichTextWidth'} = '620';</code>

6.32.29.

Ticket::Frontend::AgentTicketClose###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RichTextHeight'} = '100';</code>

6.33. Frontend::Agent::Ticket::ViewCompose

6.33.1.

Ticket::Frontend::AgentTicketCompose###Permission

Description:	Required permissions to use the ticket compose screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'Permission'} = 'compose';</code>

6.33.2.

Ticket::Frontend::AgentTicketCompose###RequiredLock

Description:	Defines if a ticket lock is required in the ticket compose screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'RequiredLock'} = '1';</code>

6.33.3.

Ticket::Frontend::AgentTicketCompose###StateDefault

Description:	Defines the default next state of a ticket if it is composed / answered in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateDefault'} = 'open';</code>

6.33.4.

Ticket::Frontend::AgentTicketCompose###StateType

Description:	Defines the next possible states after composing / answering a ticket in the ticket compose screen of the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateType'} = ['open', 'closed', 'pending auto', 'pending reminder'];</pre>

6.33.5. Ticket::Frontend::AgentTicketCompose###ArticleTypes

Description:	Specifies the different article types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'ArticleTypes'} = ['email-external', 'email-internal'];</pre>

6.33.6. Ticket::Frontend::AgentTicketCompose###DefaultArticleType

Description:	Specifies the default article type for the ticket compose screen in the agent interface if the article type cannot be automatically detected.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DefaultArticleType'} = 'email-external';</pre>

6.33.7. Ticket::Frontend::ResponseFormat

Description:	Defines the format of responses in the ticket compose screen of the agent interface (\$QData{"OrigFrom"} is From 1:1, \$QData{"OrigFromName"} is only realname of From).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ResponseFormat'} = \$QData{"Salutation"} \$QData{"StdResponse"} \$QData{"Signature"}</pre>

	<pre>\$TimeShort{"\$QData{"Created"}} - \$QData{"OrigFromName"} \$Text{"wrote"}: \$QData{"Body"} ';</pre>
--	---

6.33.8. Ticket::Frontend::Quote

Description:	Defines the used character for plaintext email quotes in the ticket compose screen of the agent interface. If this is empty or inactive, original emails will not be quoted but appended to the response.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Quote'} = '>';</code>

6.33.9. Ticket::Frontend::ResponseQuoteMaxLines

Description:	Defines the maximum number of quoted lines to be added to responses.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ResponseQuoteMaxLines'} = '99';</code>

6.33.10.

Ticket::Frontend::ComposeAddCustomerAddress

Description:	Adds customers email addresses to recipients in the ticket compose screen of the agent interface. The customers email address won't be added if the article type is email-internal.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeAddCustomerAddress'} = '1';</code>

6.33.11.

Ticket::Frontend::ComposeReplaceSenderAddress

Description:	Replaces the original sender with current customer's email address on compose answer in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeReplaceSenderAddress'} = '0';</code>

6.33.12.

Ticket::Frontend::ComposeExcludeCcRecipients

Description:	Uses Cc recipients in reply Cc list on compose an email answer in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeExcludeCcRecipients'} = '0';</code>

6.33.13.

Ticket::Frontend::AgentTicketCompose###DynamicField

Description:	Dynamic fields shown in the ticket compose screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DynamicField'} = {};</code>

6.34. Frontend::Agent::Ticket::ViewCustomer

6.34.1.

Ticket::Frontend::AgentTicketCustomer###Permission

Description:	Required permissions to change the customer of a ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'Permission'} = 'customer';</code>

6.34.2.

Ticket::Frontend::AgentTicketCustomer###RequiredLock

Description:	Defines if a ticket lock is required to change the customer of a ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'RequiredLock'} = '0';</code>

6.35. Frontend::Agent::Ticket::ViewEmailNew

6.35.1. Ticket::Frontend::AgentTicketEmail###Priority

Description:	Sets the default priority for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Priority'} = '3 normal';</code>

6.35.2.

Ticket::Frontend::AgentTicketEmail###ArticleType

Description:	Sets the default article type for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ArticleType'} = 'email-external';</code>

6.35.3.

Ticket::Frontend::AgentTicketEmail###SenderType

Description:	Sets the default sender type for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SenderType'} = 'agent';</code>

6.35.4. Ticket::Frontend::AgentTicketEmail###Subject

Description:	Sets the default subject for new email tickets (e.g. 'email Outbound') in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Subject'} = '';</code>

6.35.5. Ticket::Frontend::AgentTicketEmail###Body

Description:	Sets the default text for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Body'} = '';</code>

6.35.6. Ticket::Frontend::AgentTicketEmail###StateDefault

Description:	Sets the default next ticket state, after the creation of an email ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateDefault'} = 'open';</code>

6.35.7. Ticket::Frontend::AgentTicketEmail###StateType

Description:	Determines the next possible ticket states, after the creation of a new email ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

6.35.8. Ticket::Frontend::AgentTicketEmail###HistoryType

Description:	Defines the history type for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryType'} = 'EmailAgent';</code>

6.35.9. Ticket::Frontend::AgentTicketEmail###HistoryComment

Description:	Defines the history comment for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryComment'} = '';</code>

6.35.10. Ticket::Frontend::AgentTicketEmail###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ServiceMandatory'} = '0';</code>

6.35.11. Ticket::Frontend::AgentTicketEmail###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SLAMandatory'} = '0';</code>

6.35.12. Ticket::Frontend::AgentTicketEmail###DynamicField

Description:	Dynamic fields shown in the ticket email screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'DynamicField'} = {};</code>
-----------------	---

6.35.13.

Ticket::Frontend::AgentTicketEmail###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'RichTextWidth'} = '620';</code>

6.35.14.

Ticket::Frontend::AgentTicketEmail###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'RichTextHeight'} = '320';</code>

6.36. Frontend::Agent::Ticket::ViewEscalation

6.36.1.

Ticket::Frontend::AgentTicketEscalationView###TicketPermission

Description:	Defines the required permission to show a ticket in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'TicketPermission'} = 'rw';</code>

6.36.2.

Ticket::Frontend::AgentTicketEscalationView###ViewableTicket

Description:	Shows all open tickets (even if they are locked) in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'ViewableTicketsPage'} = '50';</code>

6.36.3.

Ticket::Frontend::AgentTicketEscalationView###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'SortBy::Default'} = 'EscalationTime';</code>

6.36.4.

Ticket::Frontend::AgentTicketEscalationView###Order::Default

Description:	Defines the default ticket order (after priority sort) in the escalation view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'Order::Default'} = 'Up';</code>

6.36.5.

Ticket::Frontend::AgentTicketEscalationView###DefaultColumns

Description:	Columns that can be filtered in the escalation view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '2', 'EscalationUpdateTime' => '1',</code>

```
'Lock' => '2',
'Owner' => '2',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '2',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

6.37. Frontend::Agent::Ticket::ViewForward

6.37.1.

Ticket::Frontend::AgentTicketForward###Permission

Description:	Required permissions to use the ticket forward screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'Permission'} = 'forward';</code>

6.37.2.

Ticket::Frontend::AgentTicketForward###RequiredLock

Description:	Defines if a ticket lock is required in the ticket forward screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RequiredLock'} = '1';</code>

6.37.3.

Ticket::Frontend::AgentTicketForward###StateDefault

Description:	Defines the default next state of a ticket after being forwarded, in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateDefault'} = 'closed successful';</code>

6.37.4.

Ticket::Frontend::AgentTicketForward###StateType

Description:	Defines the next possible states after forwarding a ticket in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

6.37.5.

Ticket::Frontend::AgentTicketForward###ArticleTypeDefault

Description:	Defines the default type of forwarded message in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypeDefault'} = 'email-external';</pre>

6.37.6.

Ticket::Frontend::AgentTicketForward###ArticleTypes

Description:	Specifies the different article types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypes'} = ['email-external', 'email-internal'];</pre>

6.37.7.

Ticket::Frontend::AgentTicketForward###DynamicField

Description:	Dynamic fields shown in the ticket forward screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'DynamicField'} = {};</code>

6.37.8.

Ticket::Frontend::AgentTicketForward###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RichTextWidth'} = '620';</code>

6.37.9.

Ticket::Frontend::AgentTicketForward###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RichTextHeight'} = '100';</code>

6.38. Frontend::Agent::Ticket::ViewFreeText

6.38.1.

Ticket::Frontend::AgentTicketFreeText###Permission

Description:	Required permissions to use the ticket free text screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Permission'} = 'rw';</code>

6.38.2.

Ticket::Frontend::AgentTicketFreeText###RequiredLock

Description:	Defines if a ticket lock is required in the ticket free text screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RequiredLock'} = '0';</code>

6.38.3. Ticket::Frontend::AgentTicketFreeText###TicketType

Description:	Sets the ticket type in the ticket free text screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'TicketType'} = '1';</code>

6.38.4. Ticket::Frontend::AgentTicketFreeText###Service

Description:	Sets the service in the ticket free text screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Service'} = '1';</code>

6.38.5. Ticket::Frontend::AgentTicketFreeText###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ServiceMandatory'} = '0';</code>

6.38.6. Ticket::Frontend::AgentTicketFreeText###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText

Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'SLAMandatory'} = '0';</code>

6.38.7.

Ticket::Frontend::AgentTicketFreeText###Queue

Description:	Sets the queue in the ticket free text screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Queue'} = '0';</code>

6.38.8.

Ticket::Frontend::AgentTicketFreeText###Owner

Description:	Sets the ticket owner in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Owner'} = '0';</code>

6.38.9.

Ticket::Frontend::AgentTicketFreeText###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'OwnerMandatory'} = '0';</code>

6.38.10.

Ticket::Frontend::AgentTicketFreeText###Responsible

Description:	Sets the responsible agent of the ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Responsible'} = '0';</code>
-----------------	--

6.38.11.

Ticket::Frontend::AgentTicketFreeText###State

Description:	If a note is added by an agent, sets the state of a ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'State'} = '0';</code>

6.38.12.

Ticket::Frontend::AgentTicketFreeText###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

6.38.13.

Ticket::Frontend::AgentTicketFreeText###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateDefault'} = 'open';</code>

6.38.14.

Ticket::Frontend::AgentTicketFreeText###Note

Description:	Allows adding notes in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText

Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Note'} = '0';</code>

6.38.15.

Ticket::Frontend::AgentTicketFreeText###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'NoteMandatory'} = '0';</code>

6.38.16.

Ticket::Frontend::AgentTicketFreeText###Subject

Description:	Defines the default subject of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Subject'} = '\$Text{"Note"}';</code>

6.38.17.

Ticket::Frontend::AgentTicketFreeText###Body

Description:	Defines the default body of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Body'} = '';</code>

6.38.18.

Ticket::Frontend::AgentTicketFreeText###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InvolvedAgent'} = '0';</code>
-----------------	--

6.38.19.

Ticket::Frontend::AgentTicketFreeText###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InformAgent'} = '0';</code>

6.38.20.

Ticket::Frontend::AgentTicketFreeText###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.38.21.

Ticket::Frontend::AgentTicketFreeText###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</code>

6.38.22.

Ticket::Frontend::AgentTicketFreeText###Priority

Description:	Shows the ticket priority options in the ticket free text screen of the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Priority'} = '0';</code>

6.38.23. Ticket::Frontend::AgentTicketFreeText###PriorityDefault

Description:	Defines the default ticket priority in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'PriorityDefault'} = '3 normal';</code>

6.38.24. Ticket::Frontend::AgentTicketFreeText###Title

Description:	Shows the title fields in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Title'} = '1';</code>

6.38.25. Ticket::Frontend::AgentTicketFreeText###HistoryType

Description:	Defines the history type for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryType'} = 'AddNote';</code>

6.38.26. Ticket::Frontend::AgentTicketFreeText###HistoryComment

Description:	Defines the history comment for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryComment'} = '%FreeText';</code>

6.38.27.

Ticket::Frontend::AgentTicketFreeText###DynamicField

Description:	Dynamic fields shown in the ticket free text screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'DynamicField'} = {};</code>

6.38.28.

Ticket::Frontend::AgentTicketFreeText###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RichTextWidth'} = '620';</code>

6.38.29.

Ticket::Frontend::AgentTicketFreeText###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RichTextHeight'} = '100';</code>

6.39. Frontend::Agent::Ticket::ViewHistory

6.39.1. Ticket::Frontend::HistoryOrder

Description:	Shows the ticket history (reverse ordered) in the agent interface.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewHistory
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HistoryOrder'} = 'normal';</code>

6.40. Frontend::Agent::Ticket::ViewLocked

6.40.1.

Ticket::Frontend::AgentTicketLockedView###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in the locked ticket view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewLocked
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'SortBy::Default'} = 'Age';</code>

6.40.2.

Ticket::Frontend::AgentTicketLockedView###Order::Default

Description:	Defines the default ticket order in the ticket locked view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewLocked
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'Order::Default'} = 'Up';</code>

6.40.3.

Ticket::Frontend::AgentTicketLockedView###DefaultColumns

Description:	Columns that can be filtered in the locked view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewLocked
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1',</code>

```
'EscalationSolutionTime' => '1',
'EscalationTime' => '1',
'EscalationUpdateTime' => '1',
'Lock' => '2',
'Owner' => '2',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '2',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

6.41. Frontend::Agent::Ticket::ViewMerge

6.41.1.

Ticket::Frontend::AgentTicketMerge###Permission

Description:	Required permissions to use the ticket merge screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'Permission'} = 'rw';</code>

6.41.2.

Ticket::Frontend::AgentTicketMerge###RequiredLock

Description:	Defines if a ticket lock is required in the ticket merge screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RequiredLock'} = '1';</code>

6.41.3. Ticket::Frontend::MergeText

Description:	When tickets are merged, the customer can be informed per email by setting the check box "Inform Sender". In this text area, you can define a pre-formatted text which can later be modified by the agents.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::MergeText'} = 'Your email with ticket number "<OTRS_TICKET>" is merged to "<OTRS_MERGE_TO_TICKET>";</code>

6.41.4. Ticket::Frontend::AutomaticMergeSubject

Description:	When tickets are merged, a note will be added automatically to the ticket which is no longer active. Here you can define the subject of this note (this subject cannot be changed by the agent).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AutomaticMergeSubject'} = 'Ticket Merged';</code>

6.41.5. Ticket::Frontend::AutomaticMergeText

Description:	When tickets are merged, a note will be added automatically to the ticket which is no longer active. Here you can define the body of this note (this text cannot be changed by the agent).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AutomaticMergeText'} = 'Merged Ticket <OTRS_TICKET> to <OTRS_MERGE_TO_TICKET>.';</code>

6.41.6. Ticket::Frontend::AgentTicketMerge###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RichTextWidth'} = '620';</code>

6.41.7. Ticket::Frontend::AgentTicketMerge###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RichTextHeight'} = '100';</code>
-----------------	--

6.42. Frontend::Agent::Ticket::ViewMove

6.42.1. Ticket::Frontend::MoveType

Description:	Determines if the list of possible queues to move to ticket into should be displayed in a dropdown list or in a new window in the agent interface. If "New Window" is set you can add a move note to the ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::MoveType'} = 'form';</code>

6.42.2. Ticket::Frontend::AgentTicketMove###State

Description:	Allows to set a new ticket state in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'State'} = '1';</code>

6.42.3.

Ticket::Frontend::AgentTicketMove###StateType

Description:	Defines the next state of a ticket after being moved to another queue, in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'StateType'} = ['open', 'closed'];</code>

6.42.4. Ticket::Frontend::AgentTicketMove###Priority

Description:	Shows the ticket priority options in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Priority'} = '0';</code>
-----------------	---

6.42.5. Ticket::Frontend::AgentTicketMove###RequiredLock

Description:	Automatically lock and set owner to current Agent after opening the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RequiredLock'} = '1';</code>

6.42.6. Ticket::Frontend::AgentTicketMove###NextScreen

Description:	Determines the next screen after the ticket is moved. LastScreenOverview will return the last overview screen (e.g. search results, queueview, dashboard). TicketZoom will return to the TicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'NextScreen'} = 'TicketZoom';</code>

6.42.7. Ticket::Frontend::AgentTicketMove###Subject

Description:	Sets the default subject for notes added in the ticket move screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Subject'} = '\$Text{"Change Queue"}';</code>

6.42.8. Ticket::Frontend::AgentTicketMove###Body

Description:	Sets the default body text for notes added in the ticket move screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Body'} = '';</code>

6.42.9.

Ticket::Frontend::AgentTicketMove###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RichTextWidth'} = '620';</code>

6.42.10.

Ticket::Frontend::AgentTicketMove###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RichTextHeight'} = '100';</code>

6.42.11.

Ticket::Frontend::AgentTicketMove###DynamicField

Description:	Dynamic fields shown in the ticket move screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'DynamicField'} = {};</code>

6.43. Frontend::Agent::Ticket::ViewNote

6.43.1.

Ticket::Frontend::AgentTicketNote###Permission

Description:	Required permissions to use the ticket note screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Permission'} = 'note';</code>
-----------------	--

6.43.2.

Ticket::Frontend::AgentTicketNote###RequiredLock

Description:	Defines if a ticket lock is required in the ticket note screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RequiredLock'} = '0';</code>

6.43.3.

Ticket::Frontend::AgentTicketNote###TicketType

Description:	Sets the ticket type in the ticket note screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'TicketType'} = '0';</code>

6.43.4. Ticket::Frontend::AgentTicketNote###Service

Description:	Sets the service in the ticket note screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Service'} = '0';</code>

6.43.5.

Ticket::Frontend::AgentTicketNote###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ServiceMandatory'} = '0';</code>

6.43.6. Ticket::Frontend::AgentTicketNote###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'SLAMandatory'} = '0';</code>

6.43.7. Ticket::Frontend::AgentTicketNote###Queue

Description:	Sets the queue in the ticket note screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Queue'} = '0';</code>

6.43.8. Ticket::Frontend::AgentTicketNote###Owner

Description:	Sets the ticket owner in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Owner'} = '0';</code>

6.43.9. Ticket::Frontend::AgentTicketNote###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'OwnerMandatory'} = '0';</code>

6.43.10. Ticket::Frontend::AgentTicketNote###Responsible

Description:	Sets the responsible agent of the ticket in the ticket note screen of the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Responsible'} = '0';</code>

6.43.11. Ticket::Frontend::AgentTicketNote###State

Description:	If a note is added by an agent, sets the state of a ticket in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'State'} = '0';</code>

6.43.12. Ticket::Frontend::AgentTicketNote###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

6.43.13. Ticket::Frontend::AgentTicketNote###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateDefault'} = 'open';</code>

6.43.14. Ticket::Frontend::AgentTicketNote###Note

Description:	Allows adding notes in the ticket note screen of the agent interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Note'} = '1';</code>

6.43.15.

Ticket::Frontend::AgentTicketNote###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'NoteMandatory'} = '1';</code>

6.43.16. Ticket::Frontend::AgentTicketNote###Subject

Description:	Sets the default subject for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Subject'} = '\$Text{"Note"}';</code>

6.43.17. Ticket::Frontend::AgentTicketNote###Body

Description:	Sets the default body text for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Body'} = '';</code>

6.43.18.

Ticket::Frontend::AgentTicketNote###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InvolvedAgent'} = '0';</code>
-----------------	--

6.43.19.

Ticket::Frontend::AgentTicketNote###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InformAgent'} = '0';</code>

6.43.20.

Ticket::Frontend::AgentTicketNote###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.43.21.

Ticket::Frontend::AgentTicketNote###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</code>

6.43.22. Ticket::Frontend::AgentTicketNote###Priority

Description:	Shows the ticket priority options in the ticket note screen of the agent interface.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Priority'} = '0';</code>

6.43.23. Ticket::Frontend::AgentTicketNote###PriorityDefault

Description:	Defines the default ticket priority in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'PriorityDefault'} = '3 normal';</code>

6.43.24. Ticket::Frontend::AgentTicketNote###Title

Description:	Shows the title fields in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Title'} = '0';</code>

6.43.25. Ticket::Frontend::AgentTicketNote###HistoryType

Description:	Defines the history type for the ticket note screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryType'} = 'AddNote';</code>

6.43.26. Ticket::Frontend::AgentTicketNote###HistoryComment

Description:	Defines the history comment for the ticket note screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryComment'} = '%Note';</code>

6.43.27.

Ticket::Frontend::AgentTicketNote###DynamicField

Description:	Dynamic fields shown in the ticket note screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'DynamicField'} = {};</code>

6.43.28.

Ticket::Frontend::AgentTicketNote###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RichTextWidth'} = '620';</code>

6.43.29.

Ticket::Frontend::AgentTicketNote###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RichTextHeight'} = '100';</code>

6.44. Frontend::Agent::Ticket::ViewOwner

6.44.1.

Ticket::Frontend::AgentTicketOwner###Permission

Description:	Required permissions to use the ticket owner screen of a zoomed ticket in the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Permission'} = 'owner';</code>

6.44.2.

Ticket::Frontend::AgentTicketOwner###RequiredLock

Description:	Defines if a ticket lock is required in the ticket owner screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RequiredLock'} = '0';</code>

6.44.3.

Ticket::Frontend::AgentTicketOwner###TicketType

Description:	Sets the ticket type in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'TicketType'} = '0';</code>

6.44.4.

Ticket::Frontend::AgentTicketOwner###Service

Description:	Sets the service in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Service'} = '0';</code>

6.44.5.

Ticket::Frontend::AgentTicketOwner###ServiceMandatory

Description:	Sets if service must be selected by the agent.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ServiceMandatory'} = '0';</code>

6.44.6. Ticket::Frontend::AgentTicketOwner###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'SLAMandatory'} = '0';</code>

6.44.7. Ticket::Frontend::AgentTicketOwner###Queue

Description:	Sets the queue in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Queue'} = '0';</code>

6.44.8. Ticket::Frontend::AgentTicketOwner###Owner

Description:	Sets the ticket owner in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Owner'} = '1';</code>

6.44.9. Ticket::Frontend::AgentTicketOwner###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'OwnerMandatory'} = '1';</code>
-----------------	--

6.44.10. Ticket::Frontend::AgentTicketOwner###Responsible

Description:	Sets the responsible agent of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Responsible'} = '0';</code>

6.44.11. Ticket::Frontend::AgentTicketOwner###State

Description:	If a note is added by an agent, sets the state of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'State'} = '0';</code>

6.44.12. Ticket::Frontend::AgentTicketOwner###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</code>

6.44.13. Ticket::Frontend::AgentTicketOwner###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateDefault'} = 'open';</code>

6.44.14. Ticket::Frontend::AgentTicketOwner###Note

Description:	Allows adding notes in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Note'} = '1';</code>

6.44.15.

Ticket::Frontend::AgentTicketOwner###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'NoteMandatory'} = '1';</code>

6.44.16.

Ticket::Frontend::AgentTicketOwner###Subject

Description:	Sets the default subject for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Subject'} = '\$Text{"Owner Update"}!';</code>

6.44.17. Ticket::Frontend::AgentTicketOwner###Body

Description:	Sets the default body text for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Body'} = '';</code>

6.44.18.

Ticket::Frontend::AgentTicketOwner###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InvolvedAgent'} = '0';</code>

6.44.19.

Ticket::Frontend::AgentTicketOwner###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InformAgent'} = '0';</code>

6.44.20.

Ticket::Frontend::AgentTicketOwner###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.44.21.

Ticket::Frontend::AgentTicketOwner###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>
-----------------	--

6.44.22.

Ticket::Frontend::AgentTicketOwner###Priority

Description:	Shows the ticket priority options in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Priority'} = '0';</pre>

6.44.23.

Ticket::Frontend::AgentTicketOwner###PriorityDefault

Description:	Defines the default ticket priority in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'PriorityDefault'} = '3 normal';</pre>

6.44.24. Ticket::Frontend::AgentTicketOwner###Title

Description:	Shows the title fields in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Title'} = '0';</pre>

6.44.25.

Ticket::Frontend::AgentTicketOwner###HistoryType

Description:	Defines the history type for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryType'} = 'AddNote';</code>
-----------------	---

6.44.26.

Ticket::Frontend::AgentTicketOwner###HistoryComment

Description:	Defines the history comment for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryComment'} = '%0owner';</code>

6.44.27.

Ticket::Frontend::AgentTicketOwner###DynamicField

Description:	Dynamic fields shown in the ticket owner screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'DynamicField'} = {};</code>

6.44.28.

Ticket::Frontend::AgentTicketOwner###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RichTextWidth'} = '620';</code>

6.44.29.

Ticket::Frontend::AgentTicketOwner###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RichTextHeight'} = '100';</code>

6.45. Frontend::Agent::Ticket::ViewPending

6.45.1.

Ticket::Frontend::AgentTicketPending###Permission

Description:	Required permissions to use the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Permission'} = 'pending';</code>

6.45.2.

Ticket::Frontend::AgentTicketPending###RequiredLock

Description:	Defines if a ticket lock is required in the ticket pending screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RequiredLock'} = '1';</code>

6.45.3.

Ticket::Frontend::AgentTicketPending###TicketType

Description:	Sets the ticket type in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'TicketType'} = '0';</code>

6.45.4.

Ticket::Frontend::AgentTicketPending###Service

Description:	Sets the service in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Service'} = '0';</code>

6.45.5. Ticket::Frontend::AgentTicketPending###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ServiceMandatory'} = '0';</code>

6.45.6. Ticket::Frontend::AgentTicketPending###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'SLAMandatory'} = '0';</code>

6.45.7. Ticket::Frontend::AgentTicketPending###Queue

Description:	Sets the queue in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Queue'} = '0';</code>

6.45.8. Ticket::Frontend::AgentTicketPending###Owner

Description:	Sets the ticket owner in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending

Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Owner'} = '0';</code>

6.45.9.

Ticket::Frontend::AgentTicketPending###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'OwnerMandatory'} = '0';</code>

6.45.10.

Ticket::Frontend::AgentTicketPending###Responsible

Description:	Sets the responsible agent of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Responsible'} = '0';</code>

6.45.11.

Ticket::Frontend::AgentTicketPending###State

Description:	If a note is added by an agent, sets the state of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'State'} = '1';</code>

6.45.12.

Ticket::Frontend::AgentTicketPending###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1

Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateType'} = ['pending reminder', 'pending auto'];</pre>
-----------------	--

6.45.13.

Ticket::Frontend::AgentTicketPending###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateDefault'} = 'pending reminder';</pre>

6.45.14.

Ticket::Frontend::AgentTicketPending###Note

Description:	Allows adding notes in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Note'} = '1';</pre>

6.45.15.

Ticket::Frontend::AgentTicketPending###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'NoteMandatory'} = '1';</pre>

6.45.16.

Ticket::Frontend::AgentTicketPending###Subject

Description:	Sets the default subject for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Subject'} = '\$Text{"Pending"}!';</code>
-----------------	--

6.45.17.

Ticket::Frontend::AgentTicketPending###Body

Description:	Sets the default body text for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Body'} = '';</code>

6.45.18.

Ticket::Frontend::AgentTicketPending###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InvolvedAgent'} = '0';</code>

6.45.19.

Ticket::Frontend::AgentTicketPending###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InformAgent'} = '0';</code>

6.45.20.

Ticket::Frontend::AgentTicketPending###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.45.21. Ticket::Frontend::AgentTicketPending###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</code>

6.45.22. Ticket::Frontend::AgentTicketPending###Priority

Description:	Shows the ticket priority options in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Priority'} = '0';</code>

6.45.23. Ticket::Frontend::AgentTicketPending###PriorityDefault

Description:	Defines the default ticket priority in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'PriorityDefault'} = '3 normal';</code>

6.45.24. Ticket::Frontend::AgentTicketPending###Title

Description:	Shows the title fields in the ticket pending screen of a zoomed ticket in the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Title'} = '0';</code>

6.45.25.

Ticket::Frontend::AgentTicketPending###HistoryType

Description:	Defines the history type for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryType'} = 'AddNote';</code>

6.45.26.

Ticket::Frontend::AgentTicketPending###HistoryComment

Description:	Defines the history comment for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryComment'} = '%Pending';</code>

6.45.27.

Ticket::Frontend::AgentTicketPending###DynamicField

Description:	Dynamic fields shown in the ticket pending screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'DynamicField'} = {};</code>

6.45.28.

Ticket::Frontend::AgentTicketPending###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RichTextWidth'} = '620';</code>

6.45.29.

Ticket::Frontend::AgentTicketPending###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RichTextHeight'} = '100';</code>

6.46.

Frontend::Agent::Ticket::ViewPhoneInbound

6.46.1.

Ticket::Frontend::AgentTicketPhoneInbound###Permission

Description:	Required permissions to use the ticket phone inbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Permission'} = 'phone';</code>

6.46.2.

Ticket::Frontend::AgentTicketPhoneInbound###RequiredLock

Description:	Defines if a ticket lock is required in the ticket phone inbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RequiredLock'} = '0';</code>

6.46.3.

Ticket::Frontend::AgentTicketPhoneInbound###ArticleType

Description:	Defines the default type of the note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'ArticleType'} = 'phone';</code>

6.46.4.

Ticket::Frontend::AgentTicketPhoneInbound###SenderType

Description:	Defines the default sender type for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'SenderType'} = 'customer';</code>

6.46.5.

Ticket::Frontend::AgentTicketPhoneInbound###Subject

Description:	Defines the default subject for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Subject'} = '\$Text{"Phone call"}!';</code>

6.46.6.

Ticket::Frontend::AgentTicketPhoneInbound###Body

Description:	Defines the default note body text for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Body'} = '';</code>

6.46.7.

Ticket::Frontend::AgentTicketPhoneInbound###State

Description:	Defines the default ticket next state after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'State'} = 'open';</code>

6.46.8.

Ticket::Frontend::AgentTicketPhoneInbound###StateType

Description:	Next possible ticket states after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

6.46.9.

Ticket::Frontend::AgentTicketPhoneInbound###HistoryType

Description:	Defines the history type for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryType'} = 'PhoneCallCustomer';</code>

6.46.10.

Ticket::Frontend::AgentTicketPhoneInbound###HistoryComment

Description:	Defines the history comment for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryComment'} = '';</code>
-----------------	--

6.46.11.

Ticket::Frontend::AgentTicketPhoneInbound###DynamicField

Description:	Dynamic fields shown in the ticket phone inbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'DynamicField'} = {};</code>

6.46.12.

Ticket::Frontend::AgentTicketPhoneInbound###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RichTextWidth'} = '475';</code>

6.46.13.

Ticket::Frontend::AgentTicketPhoneInbound###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RichTextHeight'} = '200';</code>

6.47. Frontend::Agent::Ticket::ViewPhoneNew

6.47.1.

Ticket::Frontend::AgentTicketPhone###Priority

Description:	Sets the default priority for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Priority'} = '3 normal';</code>

6.47.2.

Ticket::Frontend::AgentTicketPhone###ArticleType

Description:	Sets the default article type for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ArticleType'} = 'phone';</code>

6.47.3.

Ticket::Frontend::AgentTicketPhone###SenderType

Description:	Sets the default sender type for new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SenderType'} = 'customer';</code>

6.47.4.

Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom

Description:	Controls if more than one from entry can be set in the new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom'} = '1';</code>

6.47.5.

Ticket::Frontend::AgentTicketPhone###Subject

Description:	Sets the default subject for new phone tickets (e.g. 'Phone call') in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Subject'} = '';</code>

6.47.6. Ticket::Frontend::AgentTicketPhone###Body

Description:	Sets the default note text for new telephone tickets. E.g 'New ticket via call' in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Body'} = '';</code>

6.47.7.

Ticket::Frontend::AgentTicketPhone###StateDefault

Description:	Sets the default next state for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateDefault'} = 'open';</code>

6.47.8.

Ticket::Frontend::AgentTicketPhone###StateType

Description:	Determines the next possible ticket states, after the creation of a new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

6.47.9.

Ticket::Frontend::AgentTicketPhone###HistoryType

Description:	Defines the history type for the phone ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryType'} = 'PhoneCallCustomer';</code>
-----------------	---

6.47.10.

Ticket::Frontend::AgentTicketPhone###HistoryComment

Description:	Defines the history comment for the phone ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryComment'} = '';</code>

6.47.11.

Ticket::Frontend::AgentTicketPhone###SplitLinkType

Description:	Sets the default link type of splitted tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SplitLinkType'} = { 'Direction' => 'Target', 'LinkType' => 'ParentChild' };</code>

6.47.12.

Ticket::Frontend::AgentTicketPhone###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ServiceMandatory'} = '0';</code>

6.47.13.

Ticket::Frontend::AgentTicketPhone###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SLAMandatory'} = '0';</code>

6.47.14.

Ticket::Frontend::AgentTicketPhone###DynamicField

Description:	Dynamic fields shown in the ticket phone screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'DynamicField'} = {};</code>

6.47.15.

Ticket::Frontend::AgentTicketPhone###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'RichTextWidth'} = '620';</code>

6.47.16.

Ticket::Frontend::AgentTicketPhone###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'RichTextHeight'} = '320';</code>

6.48.

Frontend::Agent::Ticket::ViewPhoneOutbound

6.48.1.

Ticket::Frontend::AgentTicketPhoneOutbound###Permission

Description:	Required permissions to use the ticket phone outbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Permission'} = 'phone';</code>

6.48.2.

Ticket::Frontend::AgentTicketPhoneOutbound###RequiredLock

Description:	Defines if a ticket lock is required in the ticket phone outbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RequiredLock'} = '1';</code>

6.48.3.

Ticket::Frontend::AgentTicketPhoneOutbound###ArticleType

Description:	Defines the default type of the note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'ArticleType'} = 'phone';</code>

6.48.4.

Ticket::Frontend::AgentTicketPhoneOutbound###SenderType

Description:	Defines the default sender type for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'SenderType'} = 'agent';</code>

6.48.5.

Ticket::Frontend::AgentTicketPhoneOutbound###Subject

Description:	Defines the default subject for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound

Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Subject'} = '\$Text{"Phone call"}!';</code>

6.48.6.

Ticket::Frontend::AgentTicketPhoneOutbound###Body

Description:	Defines the default note body text for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Body'} = '';</code>

6.48.7.

Ticket::Frontend::AgentTicketPhoneOutbound###State

Description:	Defines the default ticket next state after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'State'} = 'closed successful';</code>

6.48.8.

Ticket::Frontend::AgentTicketPhoneOutbound###StateType

Description:	Next possible ticket states after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

6.48.9.

Ticket::Frontend::AgentTicketPhoneOutbound###HistoryType

Description:	Defines the history type for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryType'} = 'PhoneCallAgent';</code>

6.48.10.

Ticket::Frontend::AgentTicketPhoneOutbound###HistoryComment

Description:	Defines the history comment for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryComment'} = '';</code>

6.48.11.

Ticket::Frontend::AgentTicketPhoneOutbound###DynamicField

Description:	Dynamic fields shown in the ticket phone outbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'DynamicField'} = {};</code>

6.48.12.

Ticket::Frontend::AgentTicketPhoneOutbound###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RichTextWidth'} = '475';</code>

6.48.13.

Ticket::Frontend::AgentTicketPhoneOutbound###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RichTextHeight'} = '200';</code>

6.49. Frontend::Agent::Ticket::ViewPrint

6.49.1.

Ticket::Frontend::AgentTicketPrint###DynamicField

Description:	Dynamic fields shown in the ticket print screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPrint'}->{'DynamicField'} = {};</code>

6.50. Frontend::Agent::Ticket::ViewPriority

6.50.1.

Ticket::Frontend::AgentTicketPriority###Permission

Description:	Required permissions to use the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Permission'} = 'priority';</code>

6.50.2.

Ticket::Frontend::AgentTicketPriority###RequiredLock

Description:	Defines if a ticket lock is required in the ticket priority screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RequiredLock'} = '1';</code>
-----------------	---

6.50.3.

Ticket::Frontend::AgentTicketPriority###TicketType

Description:	Sets the ticket type in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'TicketType'} = '0';</code>

6.50.4.

Ticket::Frontend::AgentTicketPriority###Service

Description:	Sets the service in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Service'} = '0';</code>

6.50.5.

Ticket::Frontend::AgentTicketPriority###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ServiceMandatory'} = '0';</code>

6.50.6.

Ticket::Frontend::AgentTicketPriority###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'SLAMandatory'} = '0';</code>
-----------------	---

6.50.7.

Ticket::Frontend::AgentTicketPriority###Queue

Description:	Sets the queue in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Queue'} = '0';</code>

6.50.8.

Ticket::Frontend::AgentTicketPriority###Owner

Description:	Sets the ticket owner in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Owner'} = '0';</code>

6.50.9.

Ticket::Frontend::AgentTicketPriority###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'OwnerMandatory'} = '0';</code>

6.50.10.

Ticket::Frontend::AgentTicketPriority###Responsible

Description:	Sets the responsible agent of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Responsible'} = '0';</code>

6.50.11.

Ticket::Frontend::AgentTicketPriority###State

Description:	If a note is added by an agent, sets the state of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'State'} = '0';</code>

6.50.12.

Ticket::Frontend::AgentTicketPriority###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</code>

6.50.13.

Ticket::Frontend::AgentTicketPriority###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateDefault'} = 'open';</code>

6.50.14. Ticket::Frontend::AgentTicketPriority###Note

Description:	Allows adding notes in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Note'} = '1';</code>
-----------------	---

6.50.15. Ticket::Frontend::AgentTicketPriority###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'NoteMandatory'} = '1';</code>

6.50.16. Ticket::Frontend::AgentTicketPriority###Subject

Description:	Sets the default subject for notes added in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Subject'} = '\$Text{"Priority Update"}!';</code>

6.50.17. Ticket::Frontend::AgentTicketPriority###Body

Description:	Sets the default body text for notes added in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Body'} = '';</code>

6.50.18. Ticket::Frontend::AgentTicketPriority###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InvolvedAgent'} = '0';</code>

6.50.19. Ticket::Frontend::AgentTicketPriority###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InformAgent'} = '0';</code>

6.50.20. Ticket::Frontend::AgentTicketPriority###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.50.21. Ticket::Frontend::AgentTicketPriority###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</code>

6.50.22. Ticket::Frontend::AgentTicketPriority###Priority

Description:	Shows the ticket priority options in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority

Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Priority'} = '1';</code>

6.50.23.

Ticket::Frontend::AgentTicketPriority###PriorityDefault

Description:	Defines the default ticket priority in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'PriorityDefault'} = '3 normal';</code>

6.50.24. Ticket::Frontend::AgentTicketPriority###Title

Description:	Shows the title fields in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Title'} = '0';</code>

6.50.25.

Ticket::Frontend::AgentTicketPriority###HistoryType

Description:	Defines the history type for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryType'} = 'AddNote';</code>

6.50.26.

Ticket::Frontend::AgentTicketPriority###HistoryComment

Description:	Defines the history comment for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryComment'} = '%Priority';</code>

6.50.27.

Ticket::Frontend::AgentTicketPriority###DynamicField

Description:	Dynamic fields shown in the ticket priority screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'DynamicField'} = {};</code>

6.50.28.

Ticket::Frontend::AgentTicketPriority###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RichTextWidth'} = '620';</code>

6.50.29.

Ticket::Frontend::AgentTicketPriority###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RichTextHeight'} = '100';</code>

6.51. Frontend::Agent::Ticket::ViewQueue

6.51.1.

Ticket::Frontend::AgentTicketQueue###StripEmptyLines

Description:	Strips empty lines on the ticket preview in the queue view.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'StripEmptyLines'} = '0';</code>

6.51.2.

Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets

Description:	Shows all both ro and rw queues in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'ViewAllPossibleTickets'} = '0';</code>

6.51.3.

Ticket::Frontend::AgentTicketQueue###HighlightAge1

Description:	Sets the age in minutes (first level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge1'} = '1440';</code>

6.51.4.

Ticket::Frontend::AgentTicketQueue###HighlightAge2

Description:	Sets the age in minutes (second level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge2'} = '2880';</code>

6.51.5. Ticket::Frontend::AgentTicketQueue###Blink

Description:	Activates a blinking mechanism of the queue that contains the oldest ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Blink'} = '1';</code>

6.51.6.

Ticket::Frontend::AgentTicketQueue###QueueSort

Description:	Sorts the tickets (ascendingly or descendingly) when a single queue is selected in the queue view and after the tickets are sorted by priority. Values: 0 = ascending (oldest on top, default), 1 = descending (youngest on top). Use the QueueID for the key and 0 or 1 for value.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'QueueSort'} = { '3' => '0', '7' => '1' };</code>

6.51.7.

Ticket::Frontend::AgentTicketQueue###SortBy::Default

Description:	Defines the default sort criteria for all queues displayed in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'SortBy::Default'} = 'Age';</code>

6.51.8.

Ticket::Frontend::AgentTicketQueue###PreSort::ByPriority

Description:	Defines if a pre-sorting by priority should be done in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'PreSort::ByPriority'} = '1';</code>

6.51.9.

Ticket::Frontend::AgentTicketQueue###Order::Default

Description:	Defines the default sort order for all queues in the queue view, after priority sort.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Order::Default'} = 'Up';</code>

6.51.10.

Ticket::Frontend::AgentTicketQueue###DefaultColumns

Description:	Columns that can be filtered in the queue view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '2', 'Owner' => '2', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '2', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '2', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' };</code>

6.52.

Frontend::Agent::Ticket::ViewResponsible

6.52.1.

Ticket::Frontend::AgentTicketResponsibleView###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in the responsible view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'SortBy::Default'} = 'Age';</code>
-----------------	---

6.52.2.

Ticket::Frontend::AgentTicketResponsibleView###Order::Default

Description:	Defines the default ticket order in the responsible view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'Order::Default'} = 'Up';</code>

6.52.3.

Ticket::Frontend::AgentTicketResponsible###Permission

Description:	Required permissions to use the ticket responsible screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Permission'} = 'responsible';</code>

6.52.4.

Ticket::Frontend::AgentTicketResponsible###RequiredLock

Description:	Defines if a ticket lock is required in the ticket responsible screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RequiredLock'} = '0';</code>

6.52.5.

Ticket::Frontend::AgentTicketResponsible###TicketType

Description:	Sets the ticket type in the ticket responsible screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'TicketType'} = '0';</code>

6.52.6. Ticket::Frontend::AgentTicketResponsible###Service

Description:	Sets the service in the ticket responsible screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Service'} = '0';</code>

6.52.7. Ticket::Frontend::AgentTicketResponsible###ServiceMandatory

Description:	Sets if service must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ServiceMandatory'} = '0';</code>

6.52.8. Ticket::Frontend::AgentTicketResponsible###SLAMandatory

Description:	Sets if SLA must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'SLAMandatory'} = '0';</code>

6.52.9. Ticket::Frontend::AgentTicketResponsible###Queue

Description:	Sets the queue in the ticket responsible screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Queue'} = '0';</code>
-----------------	---

6.52.10.

Ticket::Frontend::AgentTicketResponsible###Owner

Description:	Sets the ticket owner in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Owner'} = '0';</code>

6.52.11.

Ticket::Frontend::AgentTicketResponsible###OwnerMandatory

Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'OwnerMandatory'} = '0';</code>

6.52.12.

Ticket::Frontend::AgentTicketResponsible###Responsible

Description:	Sets the responsible agent of the ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Responsible'} = '1';</code>

6.52.13.

Ticket::Frontend::AgentTicketResponsible###State

Description:	If a note is added by an agent, sets the state of a ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'State'} = '0';</code>

6.52.14.

Ticket::Frontend::AgentTicketResponsible###StateType

Description:	Defines the next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</pre>

6.52.15.

Ticket::Frontend::AgentTicketResponsible###StateDefault

Description:	Defines the default next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateDefault'} = 'open';</pre>

6.52.16.

Ticket::Frontend::AgentTicketResponsible###Note

Description:	Allows adding notes in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Note'} = '1';</pre>

6.52.17.

Ticket::Frontend::AgentTicketResponsible###NoteMandatory

Description:	Sets if note must be filled in by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'NoteMandatory'} = '1';</code>
-----------------	---

6.52.18.

Ticket::Frontend::AgentTicketResponsible###Subject

Description:	Sets the default subject for notes added in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Subject'} = '\$Text{"Responsible Update"}!';</code>

6.52.19.

Ticket::Frontend::AgentTicketResponsible###Body

Description:	Sets the default body text for notes added in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Body'} = '';</code>

6.52.20.

Ticket::Frontend::AgentTicketResponsible###InvolvedAgent

Description:	Shows a list of all the involved agents on this ticket, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'InvolvedAgent'} = '0';</code>

6.52.21.

Ticket::Frontend::AgentTicketResponsible###InformAgent

Description:	Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'InformAgent'} = '0';</code>

6.52.22.

Ticket::Frontend::AgentTicketResponsible###ArticleTypeDefault

Description:	Defines the default type of the note in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ArticleTypeDefault'} = 'note-internal';</code>

6.52.23.

Ticket::Frontend::AgentTicketResponsible###ArticleTypes

Description:	Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</code>

6.52.24.

Ticket::Frontend::AgentTicketResponsible###Priority

Description:	Shows the ticket priority options in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Priority'} = '0';</code>

6.52.25.

Ticket::Frontend::AgentTicketResponsible###PriorityDefault

Description:	Defines the default ticket priority in the ticket responsible screen of the agent interface.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'PriorityDefault'} = '3 normal';</code>

6.52.26.

Ticket::Frontend::AgentTicketResponsible###Title

Description:	Shows the title fields in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Title'} = '1';</code>

6.52.27.

Ticket::Frontend::AgentTicketResponsible###HistoryType

Description:	Defines the history type for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryType'} = 'AddNote';</code>

6.52.28.

Ticket::Frontend::AgentTicketResponsible###HistoryComment

Description:	Defines the history comment for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryComment'} = '%%Responsible';</code>

6.52.29.

Ticket::Frontend::AgentTicketResponsible###DynamicField

Description:	Dynamic fields shown in the ticket responsible screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket

SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'DynamicField'} = {};</code>

6.52.30.

Ticket::Frontend::AgentTicketResponsible###RichTextWidth

Description:	Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RichTextWidth'} = '620';</code>

6.52.31.

Ticket::Frontend::AgentTicketResponsible###RichTextHeight

Description:	Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RichTextHeight'} = '100';</code>

6.52.32.

Ticket::Frontend::AgentTicketResponsibleView###DefaultColumns

Description:	Columns that can be filtered in the responsible view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1',</code>

```
'EscalationUpdateTime' => '1',
'Lock' => '2',
'Owner' => '2',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '2',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

6.53. Frontend::Agent::Ticket::ViewSearch

6.53.1.

Ticket::Frontend::AgentTicketSearch###ExtendedSearchCondition

Description:	Allows extended search conditions in ticket search of the agent interface. With this feature you can search e. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ExtendedSearchCondition'} = '1';</code>

6.53.2.

Ticket::Frontend::AgentTicketSearch###SearchLimit

Description:	Maximum number of tickets to be displayed in the result of a search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchLimit'} = '2000';</code>

6.53.3.

Ticket::Frontend::AgentTicketSearch###SearchPageShown

Description:	Number of tickets to be displayed in each page of a search result in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchPageShown'} = '40';</code>

6.53.4.

Ticket::Frontend::AgentTicketSearch###SearchViewableTicketLines

Description:	Number of lines (per ticket) that are shown by the search utility in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchViewableTicketLines'} = '10';</code>

6.53.5.

Ticket::Frontend::AgentTicketSearch###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting of the ticket search result of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SortBy::Default'} = 'Age';</code>

6.53.6.

Ticket::Frontend::AgentTicketSearch###Order::Default

Description:	Defines the default ticket order in the ticket search result of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Order::Default'} = 'Down';</code>

6.53.7.

Ticket::Frontend::AgentTicketSearch###SearchArticleCSVTree

Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchArticleCSVTree'} = '0';</code>

6.53.8.

Ticket::Frontend::AgentTicketSearch###SearchCSVData

Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVData'} = ['TicketNumber', 'Age', 'Created', 'Closed', 'FirstLock', 'FirstResponse', 'State', 'Priority', 'Queue', 'Lock', 'Owner', 'UserFirstname', 'UserLastname', 'CustomerID', 'CustomerName', 'From', 'Subject', 'AccountedTime', 'ArticleTree', 'SolutionInMin', 'SolutionDiffInMin', 'FirstResponseInMin', 'FirstResponseDiffInMin'];</pre>

6.53.9.

Ticket::Frontend::AgentTicketSearch###ArticleCreateTime

Description:	Includes article create times in the ticket search of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ArticleCreateTime'} = '0';</pre>

6.53.10.

Ticket::Frontend::AgentTicketSearch###Defaults###Fulltext

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Fulltext'} = '';</code>
-----------------	---

6.53.11.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketNum

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketNumber'} = '';</code>

6.53.12.

Ticket::Frontend::AgentTicketSearch###Defaults###Title

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Title'} = '';</code>

6.53.13.

Ticket::Frontend::AgentTicketSearch###Defaults###From

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'From'} = '';</code>

6.53.14.

Ticket::Frontend::AgentTicketSearch###Defaults###To

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0

Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'To'} = '';</code>
-----------------	---

6.53.15.

Ticket::Frontend::AgentTicketSearch###Defaults###Cc

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Cc'} = '';</code>

6.53.16.

Ticket::Frontend::AgentTicketSearch###Defaults###Subject

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Subject'} = '';</code>

6.53.17.

Ticket::Frontend::AgentTicketSearch###Defaults###Body

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Body'} = '';</code>

6.53.18.

Ticket::Frontend::AgentTicketSearch###Defaults###CustomerID

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerID'} = '';</code>

6.53.19.

Ticket::Frontend::AgentTicketSearch###Defaults###CustomerUserLogin

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerUserLogin'} = '';</code>

6.53.20.

Ticket::Frontend::AgentTicketSearch###Defaults###StateIDs

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'StateIDs'} = [];</code>

6.53.21.

Ticket::Frontend::AgentTicketSearch###Defaults###QueueIDs

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'QueueIDs'} = [];</code>

6.53.22.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreateTimePointFormat

Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimePointFormat=year;TicketCreateTimePointStart=Last;TicketCreateTimePointEnd=Next"
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimePoint'} = '';</code>

6.53.23.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreate

Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimeStartYear=2010;TicketCreateTimeStartMonth=10;TicketCreate
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimeSlot'} = '';</code>

6.53.24.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketChar

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimePoint'} = '';</code>

6.53.25.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketChar

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimeSlot'} = '';</code>

6.53.26.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketClos

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimePoint'} = '';</code>

6.53.27.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketClose

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimeSlot'} = '';</code>

6.53.28.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketEscalation

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketEscalationTimePoint'} = '';</code>

6.53.29.

Ticket::Frontend::AgentTicketSearch###Defaults###TicketEscalation

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketEscalationTimeSlot'} = '';</code>

6.53.30.

Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreate

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimePoint'} = '';</code>

6.53.31.

Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreate

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimeSlot'} = '';</code>

6.53.32.

Ticket::Frontend::AgentTicketSearch###Defaults###SearchInA

Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'SearchInArchive'} = '';</code>

6.53.33.

Ticket::Frontend::CustomerTicketSearch###SearchArticleCSVTr

Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchArticleCSVTree'} = '0';</code>

6.53.34.

Ticket::Frontend::AgentTicketSearch###DynamicField

Description:	Dynamic fields shown in the ticket search screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and shown by default.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DynamicField'} = {};</code>

6.53.35.

Ticket::Frontend::AgentTicketSearch###Defaults###DynamicFi

Description:	Defines the default shown ticket search attribute for ticket search screen. Example: "Key" must have the name of the Dynamic Field in this case 'X', "Content" must have the value of the Dynamic Field depending on the Dynamic Field type, Text: 'a text', Dropdown: '1', Date/Time: 'Search_DynamicField_XTimeSlotStartYear=1974; Search_DynamicField_XTimeSlotStartMonth=01; Search_DynamicField_XTimeSlotStartDay=26; Search_DynamicField_XTimeSlotStartHour=00; Search_DynamicField_XTimeSlotStartMinute=00; Search_DynamicField_XTimeSlotStartSecond=00; Search_DynamicField_XTimeSlotStopYear=2013; Search_DynamicField_XTimeSlotStopMonth=01; Search_DynamicField_XTimeSlotStopDay=26; Search_DynamicField_XTimeSlotStopHour=23; Search_DynamicField_XTimeSlotStopMinute=59; Search_DynamicField_XTimeSlotStopSecond=59;' and or 'Search_DynamicField_XTimePointFormat=week; Search_DynamicField_XTimePointStart=Before; Search_DynamicField_XTimePointValue=7';.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'DynamicField'} = {};</code>

6.53.36.

Ticket::Frontend::AgentTicketSearch###SearchCSVDynamicField

Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVDynamicField'} = {};</code>

6.53.37.

Ticket::Frontend::AgentTicketSearch###DefaultColumns

Description:	Columns that can be filtered in the ticket search result view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0

Config-Setting:	<pre> \$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '2', 'Owner' => '2', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '2', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '2', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' }; </pre>
-----------------	--

6.54. Frontend::Agent::Ticket::ViewStatus

6.54.1.

Ticket::Frontend::AgentTicketStatusView###ViewableTicketsPage

Description:	Shows all open tickets (even if they are locked) in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::AgentTicketStatusView'}- >{'ViewableTicketsPage'} = '50'; </pre>

6.54.2.

Ticket::Frontend::AgentTicketStatusView###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::AgentTicketStatusView'}- >{'SortBy::Default'} = 'Age'; </pre>

6.54.3.

Ticket::Frontend::AgentTicketStatusView###Order::Default

Description:	Defines the default ticket order (after priority sort) in the status view of the agent interface. Up: oldest on top. Down: latest on top.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'Order::Default'} = 'Down';</code>

6.54.4. Ticket::Frontend::AgentTicketStatusView###DefaultColumns

Description:	Columns that can be filtered in the status view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '2', 'Owner' => '2', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '2', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '2', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' };</code>

6.55. Frontend::Agent::Ticket::ViewWatch

6.55.1. Ticket::Frontend::AgentTicketWatchView###SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in the watch view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewWatch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'SortBy::Default'} = 'Age';</code>

6.55.2.

Ticket::Frontend::AgentTicketWatchView###Order::Default

Description:	Defines the default ticket order in the watch view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewWatch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'Order::Default'} = 'Up';</code>

6.55.3.

Ticket::Frontend::AgentTicketWatchView###DefaultColumns

Description:	Columns that can be filtered in the watch view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes and Dynamic Fields (DynamicField_NameX) are allowed.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewWatch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'DefaultColumns'} = { 'Age' => '2', 'Changed' => '1', 'CustomerID' => '2', 'CustomerName' => '1', 'CustomerUserID' => '1', 'EscalationResponseTime' => '1', 'EscalationSolutionTime' => '1', 'EscalationTime' => '1', 'EscalationUpdateTime' => '1', 'Lock' => '2', 'Owner' => '2', 'PendingTime' => '1', 'Priority' => '1', 'Queue' => '2', 'Responsible' => '1', 'SLA' => '1', 'Service' => '1', 'State' => '2', 'TicketNumber' => '2', 'Title' => '2', 'Type' => '1' };</code>

6.56. Frontend::Agent::Ticket::ViewZoom

6.56.1. Ticket::Frontend::PlainView

Description:	Shows a link to see a zoomed email ticket in plain text.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::PlainView'} = '0';</code>

6.56.2. Ticket::Frontend::ZoomExpand

Description:	Shows all the articles of the ticket (expanded) in the zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpand'} = '0';</code>

6.56.3. Ticket::Frontend::ZoomExpandSort

Description:	Shows the articles sorted normally or in reverse, under ticket zoom in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpandSort'} = 'normal';</code>

6.56.4. Ticket::ZoomAttachmentDisplayCount

Description:	Shows a count of icons in the ticket zoom, if the article has attachments.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ZoomAttachmentDisplayCount'} = '20';</code>

6.56.5. Ticket::ZoomTimeDisplay

Description:	Displays the accounted time for an article in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ZoomTimeDisplay'} = '0';</code>

6.56.6. Ticket::UseArticleColors

Description:	Shows colors for different article types in the article table.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::UseArticleColors'} = '0';</code>

6.56.7. Ticket::Frontend::TicketArticleFilter

Description:	Activates the article filter in the zoom view to specify which articles should be shown.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TicketArticleFilter'} = '0';</code>

6.56.8. Ticket::Frontend::HTMLArticleHeightDefault

Description:	Set the default height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HTMLArticleHeightDefault'} = '100';</code>

6.56.9. Ticket::Frontend::HTMLArticleHeightMax

Description:	Set the maximum height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HTMLArticleHeightMax'} = '2500';</code>

6.56.10. Ticket::Frontend::ZoomRichTextForce

Description:	Show article as rich text even if rich text writing is disabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomRichTextForce'} = '1';</code>

6.56.11.

Ticket::Frontend::AgentTicketZoom###DynamicField

Description:	Dynamic fields shown in the sidebar of the ticket zoom screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'DynamicField'} = {};</code>

6.57. Frontend::Agent::TicketOverview

6.57.1. Ticket::Frontend::Overview###Small

Description:	Allows having a small format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Overview'}->{'Small'} = { 'CustomerInfo' => '1', 'Module' => 'Kernel::Output::HTML::TicketOverviewSmall', 'ModulePriority' => '100', 'Name' => 'Small', 'NameShort' => 'S' };</code>

6.57.2.

Ticket::Frontend::OverviewSmall###ColumnHeader

Description:	Shows either the last customer article's subject or the ticket title in the small format overview.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::OverviewSmall'}->{'ColumnHeader'} = 'LastCustomerSubject';</code>

6.57.3. Ticket::Frontend::Overview###Medium

Description:	Allows having a medium format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Overview'}->{'Medium'} = { 'CustomerInfo' => '0', 'Module' => 'Kernel::Output::HTML::TicketOverviewMedium', 'ModulePriority' => '200', 'Name' => 'Medium', 'NameShort' => 'M', };</code>

	<pre>'OverviewMenuModules' => '1', 'TicketActionsPerTicket' => '0' };</pre>
--	---

6.57.4. Ticket::Frontend::Overview###Preview

Description:	Shows a preview of the ticket overview (CustomerInfo => 1 - shows also Customer-Info, CustomerInfoMaxSize max. size in characters of Customer-Info).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Preview'} = { 'CustomerInfo' => '0', 'CustomerInfoMaxSize' => '18', 'DefaultPreViewLines' => '25', 'DefaultViewNewLine' => '90', 'Module' => 'Kernel::Output::HTML::TicketOverviewPreview', 'ModulePriority' => '300', 'Name' => 'Large', 'NameShort' => 'L', 'OverviewMenuModules' => '1', 'StripEmptyLines' => '0', 'TicketActionsPerTicket' => '0' };</pre>

6.57.5. Ticket::Frontend::Overview::PreviewArticleSenderTypes

Description:	Defines which article sender types should be shown in the preview of a ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview::PreviewArticleSenderTypes'} = { 'agent' => '1', 'customer' => '1', 'system' => '1' };</pre>

6.57.6. Ticket::Frontend::Overview::PreviewArticleLimit

Description:	Sets the count of articles visible in preview mode of ticket overviews.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview::PreviewArticleLimit'} = '5';</pre>

6.57.7.

Ticket::Frontend::Overview::PreviewArticleTypeExpanded

Description:	Defines wich article type should be expanded when entering the overview. If nothing defined, latest article will be expanded.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::Overview::PreviewArticleTypeExpanded'} = '';</code>

6.57.8.

Ticket::Frontend::OverviewSmall###DynamicField

Description:	Dynamic fields shown in the ticket small format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::OverviewSmall'}->{'DynamicField'} = {};</code>

6.57.9.

Ticket::Frontend::OverviewMedium###DynamicField

Description:	Dynamic fields shown in the ticket medium format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::OverviewMedium'}->{'DynamicField'} = {};</code>

6.57.10.

Ticket::Frontend::OverviewPreview###DynamicField

Description:	Dynamic fields shown in the ticket preview format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::OverviewPreview'}->{'DynamicField'} = {};</code>

6.58. Frontend::Agent::ToolBarModule

6.58.1. Frontend::ToolBarModule###1-Ticket::AgentTicketQueue

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'1-Ticket::AgentTicketQueue'} = { 'AccessKey' => 'q', 'Action' => 'AgentTicketQueue', 'CssClass' => 'QueueView', 'Icon' => 'icon-folder-close', 'Link' => 'Action=AgentTicketQueue', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Queue view', 'Priority' => '1010010' };</pre>

6.58.2. Frontend::ToolBarModule###2-Ticket::AgentTicketStatus

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'2-Ticket::AgentTicketStatus'} = { 'AccessKey' => 'o', 'Action' => 'AgentTicketStatusView', 'CssClass' => 'StatusView', 'Icon' => 'icon-list-ol', 'Link' => 'Action=AgentTicketStatusView', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Status view', 'Priority' => '1010020' };</pre>

6.58.3. Frontend::ToolBarModule###3-Ticket::AgentTicketEscalation

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'3-Ticket::AgentTicketEscalation'} = { 'AccessKey' => 'w', 'Action' => 'AgentTicketEscalationView',</pre>

```
'CssClass' => 'EscalationView',
'Icon' => 'icon-exclamation',
'Link' => 'Action=AgentTicketEscalationView',
'Module' => 'Kernel::Output::HTML::ToolBarLink',
'Name' => 'Escalation view',
'Priority' => '1010030'
};
```

6.58.4. Frontend::ToolBarModule###4-Ticket::AgentTicketPhone

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'4-Ticket::AgentTicketPhone'} = { 'AccessKey' => 'l', 'Action' => 'AgentTicketPhone', 'CssClass' => 'PhoneTicket', 'Icon' => 'icon-phone', 'Link' => 'Action=AgentTicketPhone', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New phone ticket', 'Priority' => '1020010' };</pre>

6.58.5. Frontend::ToolBarModule###5-Ticket::AgentTicketEmail

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'5-Ticket::AgentTicketEmail'} = { 'AccessKey' => 'e', 'Action' => 'AgentTicketEmail', 'CssClass' => 'EmailTicket', 'Icon' => 'icon-envelope', 'Link' => 'Action=AgentTicketEmail', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New email ticket', 'Priority' => '1020020' };</pre>

6.58.6. Frontend::ToolBarModule###6-Ticket::AgentTicketProcess

Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0

Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'6-Ticket::AgentTicketProcess'} = { 'AccessKey' => 'p', 'Action' => 'AgentTicketProcess', 'CssClass' => 'ProcessTicket', 'Icon' => 'icon-th-large', 'Link' => 'Action=AgentTicketProcess', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New process ticket', 'Priority' => '1020030' };</pre>
-----------------	---

6.58.7. Frontend::ToolBarModule###6-Ticket::TicketResponsible

Description:	Agent interface notification module to see the number of tickets an agent is responsible for.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'6-Ticket::TicketResponsible'} = { 'CssClass' => 'Responsible', 'CssClassNew' => 'Responsible New', 'CssClassReached' => 'Responsible Reached', 'Icon' => 'icon-user', 'IconNew' => 'icon-user', 'IconReached' => 'icon-user', 'Module' => 'Kernel::Output::HTML::ToolBarTicketResponsible', 'Priority' => '1030010' };</pre>

6.58.8. Frontend::ToolBarModule###7-Ticket::TicketWatcher

Description:	Agent interface notification module to see the number of watched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'7-Ticket::TicketWatcher'} = { 'CssClass' => 'Watcher', 'CssClassNew' => 'Watcher New', 'CssClassReached' => 'Watcher Reached', 'Icon' => 'icon-eye-open', 'IconNew' => 'icon-eye-open', 'IconReached' => 'icon-eye-open', 'Module' => 'Kernel::Output::HTML::ToolBarTicketWatcher', 'Priority' => '1030020' };</pre>

6.58.9. Frontend::ToolBarModule###8-Ticket::TicketLocked

Description:	Agent interface notification module to check the used charset.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'8-Ticket::TicketLocked'} = { 'CssClass' => 'Locked', 'CssClassNew' => 'Locked New', 'CssClassReached' => 'Locked Reached', 'Icon' => 'icon-lock', 'IconNew' => 'icon-lock', 'IconReached' => 'icon-lock', 'Module' => 'Kernel::Output::HTML::ToolBarTicketLocked', 'Priority' => '1030030' };</pre>

6.58.10. Frontend::ToolBarModule###9-Ticket::TicketSearchProfile

Description:	Agent interface module to access search profiles via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'9-Ticket::TicketSearchProfile'} = { 'Block' => 'ToolBarSearchProfile', 'Description' => 'Search-Template', 'MaxWidth' => '40', 'Module' => 'Kernel::Output::HTML::ToolBarTicketSearchProfile', 'Name' => 'Search-Template', 'Priority' => '1990010' };</pre>

6.58.11. Frontend::ToolBarModule###10-Ticket::TicketSearchFulltext

Description:	Agent interface module to access fulltext search via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'10-Ticket::TicketSearchFulltext'} = { 'Block' => 'ToolBarSearchFulltext', 'CSS' => 'Core.Agent.Toolbar.FulltextSearch.css', 'Description' => 'Fulltext search', 'Module' => 'Kernel::Output::HTML::ToolBarGeneric', 'Name' => 'Fulltext-Search', 'Priority' => '1990020', 'Size' => '10' };</pre>

6.58.12. Frontend::ToolBarModule###11-CICSearchCustomerID

Description:	Agent interface module to access CIC search via nav bar.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'11-CICSearchCustomerID'} = { 'Block' => 'ToolBarCICSearchCustomerID', 'CSS' => 'Core.Agent.Toolbar.CICSearch.css', 'Description' => 'CustomerID Search', 'Module' => 'Kernel::Output::HTML::ToolBarGeneric', 'Name' => 'CustomerID search', 'Priority' => '1990030', 'Size' => '10' };</pre>

6.58.13. Frontend::ToolBarModule###11-CICSearchCustomerUser

Description:	Agent interface module to access CIC search via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'11-CICSearchCustomerUser'} = { 'Block' => 'ToolBarCICSearchCustomerUser', 'CSS' => 'Core.Agent.Toolbar.CICSearch.css', 'Description' => 'CustomerUser Search', 'Module' => 'Kernel::Output::HTML::ToolBarGeneric', 'Name' => 'Customer user search', 'Priority' => '1990040', 'Size' => '10' };</pre>

6.59. Frontend::Customer

6.59.1.

Ticket::Frontend::CustomerDisableCompanyTicketAccess

Description:	This option will deny the access to customer company tickets, which are not created by the customer user.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerDisableCompanyTicketAccess'} = '0';</pre>

6.59.2.

Ticket::Frontend::CustomerTicketOverviewCustomEmptyText

Description:	Custom text for the page shown to customers that have no tickets yet.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketOverviewCustomEmptyText'} = { 'Button' => 'Create your first ticket', 'Text' => 'Please click the button below to create your first ticket.', 'Title' => 'Welcome!' };</pre>

6.59.3. Frontend::CustomerUser::Item###15-OpenTickets

Description:	Customer item (icon) which shows the open tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'15-OpenTickets'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Open;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '0', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Open tickets' };</pre>

6.59.4. Frontend::CustomerUser::Item###16-OpenTicketsForCustomerUserLogin

Description:	Customer item (icon) which shows the open tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'16-OpenTicketsForCustomerUserLogin'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Open;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '1', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', };</pre>

```
'Text' => 'Open tickets of customer'
};
```

6.59.5. Frontend::CustomerUser::Item###17-ClosedTickets

Description:	Customer item (icon) which shows the closed tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'17-ClosedTickets'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Closed;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '0', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Closed tickets' };</pre>

6.59.6. Frontend::CustomerUser::Item###18-ClosedTicketsForCustomerUserLogin

Description:	Customer item (icon) which shows the closed tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'18- ClosedTicketsForCustomerUserLogin'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Closed;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '1', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Closed tickets of customer' };</pre>

6.59.7. CustomerFrontend::CommonObject###QueueObject

Description:	Path of the file that stores all the settings for the QueueObject object for the customer interface.
Group:	Ticket

SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerFrontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue';</code>

6.59.8. CustomerFrontend::CommonObject###TicketObject

Description:	Path of the file that stores all the settings for the TicketObject for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerFrontend::CommonObject'}->{'TicketObject'} = 'Kernel::System::Ticket';</code>

6.59.9. CustomerFrontend::CommonParam###Action

Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerFrontend::CommonParam'}->{'Action'} = 'CustomerTicketOverview';</code>

6.59.10. CustomerFrontend::CommonParam###TicketID

Description:	Default ticket ID used by the system in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerFrontend::CommonParam'}->{'TicketID'} = '';</code>

6.60. Frontend::Customer::ModuleMetaHead

6.60.1. CustomerFrontend::HeaderMetaModule###2-TicketSearch

Description:	Module to generate html OpenSearch profile for short ticket search in the customer interface.
Group:	Ticket

SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::HeaderMetaModule'}->{'2-TicketSearch'} = { 'Action' => 'CustomerTicketSearch', 'Module' => 'Kernel::Output::HTML::CustomerHeaderMetaTicketSearch' };</pre>

6.61.

Frontend::Customer::ModuleRegistration

6.61.1.

CustomerFrontend::Module###CustomerTicketOverview

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketOverview'} = { 'Description' => 'Overview of customer tickets', 'NavBar' => [{ 'AccessKey' => 't', 'Block' => '', 'Description' => 'Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets', 'LinkOption' => '', 'Name' => 'Tickets', 'NavBar' => 'Ticket', 'Prio' => '100', 'Type' => 'Menu' }, { 'AccessKey' => 'm', 'Block' => '', 'Description' => 'My Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets', 'LinkOption' => '', 'Name' => 'My Tickets', 'NavBar' => 'Ticket', 'Prio' => '110', 'Type' => 'Submenu' }, { 'AccessKey' => 'c', 'Block' => '', 'Description' => 'Company Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=CompanyTickets', 'LinkOption' => '', 'Name' => 'Company Tickets', 'NavBar' => 'Ticket', 'Prio' => '120', 'Type' => 'Submenu' }], 'NavBarName' => 'Ticket', 'Title' => 'Overview' };</pre>

6.61.2. CustomerFrontend::Module###CustomerTicketMessage

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}->{'CustomerTicketMessage'} = { 'Description' => 'Create tickets', 'NavBar' => [{ 'AccessKey' => 'n', 'Block' => '', 'Description' => 'Create new Ticket', 'Link' => 'Action=CustomerTicketMessage', 'LinkOption' => '', 'Name' => 'New Ticket', 'NavBar' => 'Ticket', 'Prio' => '100', 'Type' => 'Submenu' }], 'NavBarName' => 'Ticket', 'Title' => 'New Ticket' }; </pre>

6.61.3. CustomerFrontend::Module###CustomerTicketZoom

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}->{'CustomerTicketZoom'} = { 'Description' => 'Ticket zoom view', 'Loader' => { 'JavaScript' => ['Core.Customer.TicketZoom.js', 'Core.UI.Popup.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' }; </pre>

6.61.4. CustomerFrontend::Module###CustomerTicketPrint

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0

Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketPrint'} = { 'Description' => 'Customer Ticket Print Module', 'NavBarName' => '', 'Title' => 'Print' };</pre>
-----------------	--

6.61.5. CustomerFrontend::Module###CustomerTicketAttachment

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketAttachment'} = { 'Description' => 'To download attachments', 'NavBarName' => '', 'Title' => '' };</pre>

6.61.6. CustomerFrontend::Module###CustomerTicketSearch

Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketSearch'} = { 'Description' => 'Customer ticket search', 'NavBar' => [{ 'AccessKey' => 's', 'Block' => '', 'Description' => 'Search', 'Link' => 'Action=CustomerTicketSearch', 'LinkOption' => '', 'Name' => 'Search', 'NavBar' => 'Ticket', 'Prio' => '300', 'Type' => 'Submenu' }], 'NavBarName' => 'Ticket', 'Title' => 'Search' };</pre>

6.62. Frontend::Customer::Preferences

6.62.1. CustomerPreferencesGroups###ShownTickets

Description:	Defines all the parameters for the ShownTickets object in the customer preferences of the customer interface.
Group:	Ticket

SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPreferencesGroups'}->{'ShownTickets'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '15' => '15', '20' => '20', '25' => '25', '30' => '30' }, 'DataSelected' => '25', 'Key' => 'Tickets per page', 'Label' => 'Number of displayed tickets', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserShowTickets', 'Prio' => '4000' }; </pre>

6.62.2. CustomerPreferencesGroups###RefreshTime

Description:	Defines all the parameters for the RefreshTime object in the customer preferences of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPreferencesGroups'}->{'RefreshTime'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '' => 'off', '10' => '10 minutes', '15' => '15 minutes', '2' => ' 2 minutes', '5' => ' 5 minutes', '7' => ' 7 minutes' }, 'DataSelected' => '', 'Key' => 'Refresh interval', 'Label' => 'Ticket overview', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '4000' }; </pre>

6.63. Frontend::Customer::Ticket::ViewNew

6.63.1.

Ticket::Frontend::CustomerTicketMessage###NextScreenAfterNew

Description:	Determines the next screen after new customer ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'NextScreenAfterNewTicket'} = 'CustomerTicketOverview';</code>
-----------------	--

6.63.2.

Ticket::Frontend::CustomerTicketMessage###Priority

Description:	Allows customers to set the ticket priority in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Priority'} = '1';</code>

6.63.3.

Ticket::Frontend::CustomerTicketMessage###PriorityDefault

Description:	Defines the default priority of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'PriorityDefault'} = '3 normal';</code>

6.63.4.

Ticket::Frontend::CustomerTicketMessage###Queue

Description:	Allows customers to set the ticket queue in the customer interface. If this is set to 'No', QueueDefault should be configured.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Queue'} = '1';</code>

6.63.5.

Ticket::Frontend::CustomerTicketMessage###QueueDefault

Description:	Defines the default queue for new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'QueueDefault'} = 'Postmaster';</code>

6.63.6.

Ticket::Frontend::CustomerTicketMessage###TicketType

Description:	Allows customers to set the ticket type in the customer interface. If this is set to 'No', TicketTypeDefault should be configured.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'TicketType'} = '1';</code>

6.63.7.

Ticket::Frontend::CustomerTicketMessage###TicketTypeDefault

Description:	Defines the default ticket type for new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'TicketTypeDefault'} = 'default';</code>

6.63.8.

Ticket::Frontend::CustomerTicketMessage###Service

Description:	Allows customers to set the ticket service in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Service'} = '1';</code>

6.63.9.

Ticket::Frontend::CustomerTicketMessage###SLA

Description:	Allows customers to set the ticket SLA in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1

Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SLA'} = '1';</code>

6.63.10.

Ticket::Frontend::CustomerTicketMessage###ServiceMandatory

Description:	Sets if service must be selected by the customer.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'ServiceMandatory'} = '0';</code>

6.63.11.

Ticket::Frontend::CustomerTicketMessage###SLAMandatory

Description:	Sets if SLA must be selected by the customer.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SLAMandatory'} = '0';</code>

6.63.12.

Ticket::Frontend::CustomerTicketMessage###StateDefault

Description:	Defines the default state of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'StateDefault'} = 'new';</code>

6.63.13.

Ticket::Frontend::CustomerTicketMessage###ArticleType

Description:	Defines the default type for article in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'ArticleType'} = 'webrequest';</code>

6.63.14.

Ticket::Frontend::CustomerTicketMessage###SenderType

Description:	Sender type for new tickets from the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SenderType'} = 'customer';</code>

6.63.15.

Ticket::Frontend::CustomerTicketMessage###HistoryType

Description:	Defines the default history type in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryType'} = 'WebRequestCustomer';</code>

6.63.16.

Ticket::Frontend::CustomerTicketMessage###HistoryComment

Description:	Comment for new history entries in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryComment'} = '';</code>

6.63.17. CustomerPanelSelectionType

Description:	Defines the receipt target of the tickets ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionType'} = 'Queue';</code>

6.63.18. CustomerPanelSelectionString

Description:	Determines the strings that will be shown as receipt (To:) of the ticket in the customer interface. For Queue as
--------------	--

	CustomerPanelSelectionType, "<Queue>" shows the names of the queues, and for SystemAddress, "<Realname> <<Email>>" shows the name and email of the receiptent.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionString'} = '<Queue>';</code>

6.63.19. CustomerPanelOwnSelection

Description:	Determines which queues will be valid for ticket's receiptents in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelOwnSelection'} = { 'Junk' => 'First Queue', 'Misc' => 'Second Queue' };</code>

6.63.20. CustomerPanel::NewTicketQueueSelectionModule

Description:	Module for To-selection in new ticket screen in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanel::NewTicketQueueSelectionModule'} = 'Kernel::Output::HTML::CustomerNewTicketQueueSelectionGeneric';</code>

6.63.21. Ticket::Frontend::CustomerTicketMessage###DynamicField

Description:	Dynamic fields options shown in the ticket message screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required. NOTE. If you want to display these fields also in the ticket zoom of the customer interface, you have to enable them in CustomerTicketZoom###DynamicField.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'DynamicField'} = {};</code>

6.64. Frontend::Customer::Ticket::ViewPrint

6.64.1.

Ticket::Frontend::CustomerTicketPrint###DynamicField

Description:	Dynamic fields shown in the ticket print screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketPrint'}->{'DynamicField'} = {};</code>

6.65. Frontend::Customer::Ticket::ViewSearch

6.65.1. Ticket::CustomerTicketSearch::SearchLimit

Description:	Maximum number of tickets to be displayed in the result of a search in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::SearchLimit'} = '5000';</code>

6.65.2.

Ticket::CustomerTicketSearch::SearchPageShown

Description:	Number of tickets to be displayed in each page of a search result in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::SearchPageShown'} = '40';</code>

6.65.3. Ticket::CustomerTicketSearch::SortBy::Default

Description:	Defines the default ticket attribute for ticket sorting in a ticket search of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::SortBy::Default'} = 'Age';</code>

6.65.4. Ticket::CustomerTicketSearch::Order::Default

Description:	Defines the default ticket order of a search result in the customer interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::Order::Default'} = 'Down';</code>

6.65.5.

Ticket::Frontend::CustomerTicketSearch###ExtendedSearchCondition

Description:	Allows extended search conditions in ticket search of the customer interface. With this feature you can search w. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'ExtendedSearchCondition'} = '1';</code>

6.65.6.

Ticket::Frontend::CustomerTicketSearch###SearchCSVData

Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVData'} = ['TicketNumber', 'Age', 'Created', 'Closed', 'State', 'Priority', 'Lock', 'CustomerID', 'CustomerName', 'From', 'Subject'];</code>

6.65.7.

Ticket::Frontend::CustomerTicketSearch###DynamicField

Description:	Dynamic fields shown in the ticket search screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket

SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'DynamicField'} = {};</code>

6.65.8.

Ticket::Frontend::CustomerTicketSearch###SearchOverviewDynamic

Description:	Dynamic fields shown in the ticket search overview results screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchOverviewDynamicField'} = {};</code>

6.65.9.

Ticket::Frontend::CustomerTicketSearch###SearchCSVDynamic

Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVDynamicField'} = {};</code>

6.66. Frontend::Customer::Ticket::ViewZoom

6.66.1.

Ticket::Frontend::CustomerTicketZoom###NextScreenAfterFollow

Description:	Determines the next screen after the follow up screen of a zoomed ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'NextScreenAfterFollowUp'} = 'CustomerTicketOverview';</code>

6.66.2.

Ticket::Frontend::CustomerTicketZoom###ArticleType

Description:	Defines the default type of the note in the ticket zoom screen of the customer interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'ArticleType'} = 'webrequest';</code>

6.66.3. Ticket::Frontend::CustomerTicketZoom###SenderType

Description:	Defines the default sender type for tickets in the ticket zoom screen of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'SenderType'} = 'customer';</code>

6.66.4. Ticket::Frontend::CustomerTicketZoom###HistoryType

Description:	Defines the history type for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryType'} = 'FollowUp';</code>

6.66.5. Ticket::Frontend::CustomerTicketZoom###HistoryComment

Description:	Defines the history comment for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryComment'} = '';</code>

6.66.6. Ticket::Frontend::CustomerTicketZoom###Priority

Description:	Allows customers to change the ticket priority in the customer interface.
--------------	---

Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'Priority'} = '1';</code>

6.66.7.

Ticket::Frontend::CustomerTicketZoom###PriorityDefault

Description:	Defines the default priority of follow up customer tickets in the ticket zoom screen in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'PriorityDefault'} = '3 normal';</code>

6.66.8.

Ticket::Frontend::CustomerTicketZoom###State

Description:	Allows choosing the next compose state for customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'State'} = '1';</code>

6.66.9.

Ticket::Frontend::CustomerTicketZoom###StateDefault

Description:	Defines the default next state for a ticket after customer follow up in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateDefault'} = 'open';</code>

6.66.10.

Ticket::Frontend::CustomerTicketZoom###StateType

Description:	Defines the next possible states for customer tickets in the customer interface.
--------------	--

Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateType'} = ['open', 'closed'];</pre>

6.66.11.

Ticket::Frontend::CustomerTicketZoom###AttributesView

Description:	Shows the activated ticket attributes in the customer interface (0 = Disabled and 1 = Enabled).
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'AttributesView'} = { 'Owner' => '0', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '0', 'SLA' => '0', 'Service' => '0', 'State' => '1', 'Type' => '0' };</pre>

6.66.12.

Ticket::Frontend::CustomerTicketZoom###DynamicField

Description:	Dynamic fields shown in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DynamicField'} = {};</pre>

6.66.13.

Ticket::Frontend::CustomerTicketZoom###FollowUpDynamicField

Description:	Dynamic fields options shown in the ticket reply section in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1

Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'FollowUpDynamicField'} = {};</code>

6.67. Frontend::Customer::TicketOverview

6.67.1.

Ticket::Frontend::CustomerTicketOverviewSortable

Description:	Controls if customers have the ability to sort their tickets.
Group:	Ticket
SubGroup:	Frontend::Customer::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketOverviewSortable'} = '';</code>

6.67.2.

Ticket::Frontend::CustomerTicketOverview###ColumnHeader

Description:	Shows either the last customer article's subject or the ticket title in the small format overview.
Group:	Ticket
SubGroup:	Frontend::Customer::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'ColumnHeader'} = 'TicketTitle';</code>

6.67.3.

Ticket::Frontend::CustomerTicketOverview###Owner

Description:	Show the current owner in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::TicketOverview
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'Owner'} = '0';</code>

6.67.4.

Ticket::Frontend::CustomerTicketOverview###Queue

Description:	Show the current queue in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::TicketOverview
Valid:	1
Required:	1

Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'Queue'} = '0';</code>
-----------------	---

6.67.5.

Ticket::Frontend::CustomerTicketOverview###DynamicField

Description:	Dynamic fields shown in the ticket overview screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Customer::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'DynamicField'} = {};</code>

6.68. Frontend::Queue::Preferences

6.68.1. QueuePreferences###Comment2

Description:	Parameters of the example queue attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Queue::Preferences
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'QueuePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the queue comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::QueuePreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' };</code>

6.69. Frontend::SLA::Preferences

6.69.1. SLAPreferences###Comment2

Description:	Parameters of the example SLA attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::SLA::Preferences
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SLAPreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the sla comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::SLAPreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' };</code>

6.70. Frontend::Service::Preferences

6.70.1. ServicePreferences###Comment2

Description:	Parameters of the example service attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Service::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$self->{'ServicePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the service comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::ServicePreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' }; </pre>

Приложение С. Благодарности


OTRS является проектом с открытым кодом и, мы хотим поблагодарить многих людей за их помощь и поддержку. Конечно же следующий список далеко неполный и мы приносим свои извинения за это! Просто напишите нам, если не наши себя в этом списке.

Следующие люди внесли особый вклад в проект и являются его активными сторонниками:

- Robert Kehl, создавший Win32-инсталлятор для релиза 1.x. Больше тебе спасибо, Robert!
- Torsten Werner, который поддерживает Debian-installer для OTRS. Спасибо большое, Torsten, теперь OTRS также доступен для Debian-сообщества.
- Nils Jeppe (зеркало Гамбург, Германия), Bryan Fullerton (зеркало Торонто, Канада), Eberhard Mönkeberg (зеркало Геттинген, Германия), Timo Dreger (зеркало Дюссельдорф, Германия) и Netmonic (зеркало Вена, Австрия), которые поддерживают зеркальные копии нашего ftp-сервера. Большое спасибо, благодаря вашей помощи всегда есть возможность быстро скачать OTRS!
- Anja Schneider, которая помогает с переводом этого руководства и коррекции и пересмотр немецких текстов. Большое спасибо за вашу помощь и терпение, Anja!
- Через списки рассылки OTRS мы получаем множество идей, советов по усовершенствованию системы, обновления и исправления. Также здесь доступна большая поддержка для всех пользователей. Большое спасибо всем людям, которые участвуют в наших списках рассылки за вашу активную помощь!

Также, мы хотели бы выразить большую благодарность следующим людям:

- Martin Scherbaum
- Carsten Gross
- Harald Müller
- Stefan Schmidt
- Milisav Radmanic
- Uli Hecht
- Norman Walsh
- Heiko Baumann
- Atif Ghaffar
- Pablo Ruiz Garcia
- Dan Rau
- Christoph Kaulich
- Mark Jackson
- Diane Shieh
- Bernard Choppy

-
- Carl Bailey
 - Phil Davis
 - Edwin D. Vinas
 - Lars Müller
 - Vladimir Gerdjikov
 - Fred van Dijk
 - Sebastien Guilbaud
 - Wiktor Wodecki
 - Arnold Ligtoet
 - Antti Kämäräinen
 - Nicolas Goralski
 - Gilberto Cezar de Almeida
 - Jorge Becerra
 - Eddie Urenda
 - Stella Power
 - Andreas Haase
 - Reiner Keller
 - Covert Jake
 - Moshe Leibovitch
 - Björn Jacke
 - Remo Catelotti
 - Alfons Obermeyer
 - Michael Rech
 - Danie Theron
 - Richard Paradies
 - Art Powell
- 

Приложение D. GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document, that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters), and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats, suitable for input to text formatters. A copy made in an otherwise Transparent file format, whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include: plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include: PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location,

containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously, at no charge, using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location, until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version, as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various

documents with a single copy that is included in the collection, provided that you follow the rules of this License, for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections, in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software

Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

. How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.