

ES2008-SC DEFINITIONS ::= BEGIN

IMPORTS

    internet  
        FROM RFC1155-SMI  
MODULE-IDENTITY, OBJECT-TYPE, Integer32, IPAddress  
    FROM SNMPv2-SMI  
DisplayString  
    FROM SNMPv2-TC;

private            OBJECT IDENTIFIER ::= { internet 4 }  
enterprises        OBJECT IDENTIFIER ::= { private 1 }  
zyxel              OBJECT IDENTIFIER ::= { enterprises 890 }

products           OBJECT IDENTIFIER ::= { zyxel 1 }  
accessSwitch       OBJECT IDENTIFIER ::= { products 5 }

esSeries           OBJECT IDENTIFIER ::= { accessSwitch 8 }  
es2008              OBJECT IDENTIFIER ::= { esSeries 3 }  
es2008-gtp          OBJECT IDENTIFIER ::= { esSeries 4 }  
es2008-sc           OBJECT IDENTIFIER ::= { esSeries 5 }  
es2008-sc30         OBJECT IDENTIFIER ::= { esSeries 6 }

information MODULE-IDENTITY

    LAST-UPDATED "200204290000Z"  
    ORGANIZATION "ZyXEL Communications Co."  
    CONTACT-INFO  
        "ZyXEL Communications Co.

        6 Innovation Road II,  
        Science-Based Industrial Park,  
        Hsin-chu, 300 Taiwan

        Phone: +886-3-578-3942  
        Fax:   +886-3-578-2439  
        Email: sales@zyxel.com.tw"

DESCRIPTION

    "The MIB module for ES2008-SC"

REVISION "200210260000Z"

DESCRIPTION

    "Initial version of this MIB."

::= { zyxel 2 }

```

switchInfo          OBJECT IDENTIFIER ::= { es2008-sc 1 }
switchPortMgt       OBJECT IDENTIFIER ::= { es2008-sc 2 }
systemSTAMgt        OBJECT IDENTIFIER ::= { es2008-sc 3 }
tftpDownloadMgt     OBJECT IDENTIFIER ::= { es2008-sc 4 }
restartMgt          OBJECT IDENTIFIER ::= { es2008-sc 5 }
portMirrorMgt       OBJECT IDENTIFIER ::= { es2008-sc 6 }
igmpMgt             OBJECT IDENTIFIER ::= { es2008-sc 7 }

```

```

--
-- switchInfo
--

```

```

switchNumber OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION  "The total number of switches present on this system."
    ::= { switchInfo 1 }

```

```

switchInfoTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SwitchInfoEntry
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION  "Table of descriptive and status information about
                  switches in this system."
    ::= { switchInfo 2 }

```

```

switchInfoEntry OBJECT-TYPE
    SYNTAX      SwitchInfoEntry
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION  "An entry in the table, containing information
                  about a single switch in this system. "
    INDEX       { swUnitIndex }
    ::= { switchInfoTable 1 }

```

```

SwitchInfoEntry ::= SEQUENCE
{
    swUnitIndex          Integer32,
    swMainBoardHwVer     DisplayString,
    swMainBoardFwVer     DisplayString,

```

```

swAgentBoardHwVer      DisplayString,
swAgentBoardFwVer      DisplayString,
swAgentBoardPOSTCodeVer DisplayString,
swPortNumber           Integer32,
swPowerStatus          INTEGER,
swExpansionSlot1       INTEGER,
swExpansionSlot2       INTEGER,
swRoleInSystem         INTEGER
}

```

```

swUnitIndex OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "This object identifies the switch within the system
                for which this entry contains information. This
                value can never be greater than switchNumber."
    ::= { switchInfoEntry 1 }

```

```

swMainBoardHwVer OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(0..20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Hardware version of the main board."
    ::= { switchInfoEntry 2 }

```

```

swMainBoardFwVer OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(0..20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Firmware version of the main board."
    ::= { switchInfoEntry 3 }

```

```

swAgentBoardHwVer OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(0..20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Hardware version of the agent board."
    ::= { switchInfoEntry 4 }

```

```

swAgentBoardFwVer OBJECT-TYPE
    SYNTAX      DisplayString (SIZE(0..20))
    MAX-ACCESS  read-only
    STATUS      current

```

DESCRIPTION "Firmware version of the agent board."  
::= { switchInfoEntry 5 }

swAgentBoardPOSTCodeVer OBJECT-TYPE

SYNTAX DisplayString (SIZE(0..20))  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION "POST code version of the agent board."  
::= { switchInfoEntry 6 }

swPortNumber OBJECT-TYPE

SYNTAX Integer32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION "The total port number of this switch (  
including expansion slot)."  
::= { switchInfoEntry 7 }

swPowerStatus OBJECT-TYPE

SYNTAX INTEGER  
{  
internalPower(1),  
redundantPower(2),  
internalAndRedundantPower(3)  
}  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION "Indicates the switch using internalPower(1),  
redundantPower(2) or both(3)"  
::= { switchInfoEntry 8 }

swExpansionSlot1 OBJECT-TYPE

SYNTAX INTEGER  
{  
hundredBaseFX2Port(1),  
thousandBaseSX(2),  
stackingModule4GB(3),  
hundredBaseFX1Port(4),  
thousandBaseLX(5),  
thousandBaseT(6),  
thousandBaseGBIC(7),  
stackingModule2GB(8),  
other(9),  
notPresent(10),

```

        tenHundredBaseT(11),
        thousandBaseSXMtrj2Port(12),
        thousandBaseSXSc2Port(13),
        thousandBaseLXSc2Port(14),
        hundredBaseFXMtrj2Port(15),
        thousandBaseLXMtrj(16),
        thousandBaseT2Port(17),
        thousandBaseGBIC2Port(18)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Type of expansion module in this switch slot 1."
    ::= { switchInfoEntry 9 }

```

swExpansionSlot2 OBJECT-TYPE

```

    SYNTAX INTEGER
    {
        hundredBaseFX2Port(1),
        thousandBaseSX(2),
        stackingModule4GB(3),
        hundredBaseFX1Port(4),
        thousandBaseLX(5),
        thousandBaseT(6),
        thousandBaseGBIC(7),
        stackingModule2GB(8),
        other(9),
        notPresent(10),
        tenHundredBaseT(11),
        thousandBaseSXMtrj2Port(12),
        thousandBaseSXSc2Port(13),
        thousandBaseLXSc2Port(14),
        hundredBaseFXMtrj2Port(15),
        thousandBaseLXMtrj(16),
        thousandBaseT2Port(17),
        thousandBaseGBIC2Port(18)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Type of expansion module in this switch slot 2."
    ::= { switchInfoEntry 10 }

```

swRoleInSystem OBJECT-TYPE

```

    SYNTAX INTEGER
    {

```

```

        master(1),
        backupMaster(2),
        slave(3)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Indicates the switch is master(1), backupMaster(2)
             or slave(3) in this system."
::= { switchInfoEntry 11 }

--
-- switchPortMgt
--

switchPortMgtTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SwitchPortMgtEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "Table of descriptive and status information about
               configuration of each switch ports(including expansion slot)
               in this system."
    ::= { switchPortMgt 1 }

switchPortMgtEntry OBJECT-TYPE
    SYNTAX      SwitchPortMgtEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "An entry in the table, containing information
               about configuration in one switch port of the switch."
    INDEX       { swUnitIndex, swPortMgtIndex }
    ::= { switchPortMgtTable 1 }

SwitchPortMgtEntry ::= SEQUENCE
{
    swPortMgtIndex      Integer32,
    swPortMgtPortType   INTEGER,
    swPortMgtSpeedDpxAdmin  INTEGER,
    swPortMgtSpeedDpxInUse  INTEGER,
    swPortMgtFlowCtrlAdmin  INTEGER,
    swPortMgtFlowCtrlInUse  INTEGER
}

swPortMgtIndex OBJECT-TYPE

```

SYNTAX Integer32  
 MAX-ACCESS not-accessible  
 STATUS current  
 DESCRIPTION "This object identifies the port within the switch  
 for which this entry contains information."  
 ::= { switchPortMgtEntry 1 }

swPortMgtPortType OBJECT-TYPE

SYNTAX INTEGER  
 {  
     hundredBaseTX(1),  
     hundredBaseFX(2),  
     thousandBaseSX(3),  
     thousandBaseLX(4),  
     thousandBaseT(5),  
     thousandBaseGBIC(6),  
     other(7)  
 }  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION "Indicates the port type."  
 ::= { switchPortMgtEntry 2 }

swPortMgtSpeedDpxAdmin OBJECT-TYPE

SYNTAX INTEGER  
 {  
     halfDuplex10(1),  
     fullDuplex10(2),  
     halfDuplex100(3),  
     fullDuplex100(4),  
     halfDuplex1000(5),  
     fullDuplex1000(6),  
     autoNegotiation(7)  
 }  
 MAX-ACCESS read-write  
 STATUS current  
 DESCRIPTION "Set the port speed and duplex mode as follows:  
     halfDuplex10(1) - 10Mbps and half duplex mode  
     fullDuplex10(2) - 10Mbps and full duplex mode  
     halfDuplex100(3) - 100Mbps and half duplex mode  
     fullDuplex100(4) - 100Mbps and full duplex mode  
     halfDuplex1000(5) - 1000Mbps and half duplex mode  
     fullDuplex1000(6) - 1000Mbps and full duplex mode  
     autoNegotiation(7) - let the switch to negotiate

with the other end of connection.

hundredBaseTX port can be set as

- halfDuplex10(1)
- fullDuplex10(2)
- halfDuplex100(3)
- fullDuplex100(4)
- autoNegotiation(7)

hundredBaseFX port can be set as

- halfDuplex100(3)
- fullDuplex100(4)

thousandBaseSX port can be set as

- halfDuplex1000(5)
- fullDuplex1000(6)
- autoNegotiation(7)

The actual operating speed and duplex of the port  
is given by swPortMgtSpeedDpxInUse."

DEFVAL { autoNegotiation }  
::= { switchPortMgtEntry 3 }

swPortMgtSpeedDpxInUse OBJECT-TYPE

SYNTAX INTEGER  
{  
    halfDuplex10(1),  
    fullDuplex10(2),  
    halfDuplex100(3),  
    fullDuplex100(4),  
    halfDuplex1000(5),  
    fullDuplex1000(6)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The operating speed and duplex mode of the  
switched port."

::= { switchPortMgtEntry 4 }

swPortMgtFlowCtrlAdmin OBJECT-TYPE

SYNTAX INTEGER  
{  
    enabled(1),  
    disabled(2),  
    backPressure(3),  
    dot3xFlowControl(4)  
}

MAX-ACCESS read-write



STATUS current

DESCRIPTION "(1) Flow control mechanism is enabled.  
 If the port type is hundredBaseTX or thousandBaseSX:  
 When the port is operating in halfDuplex mode, the port uses backPressure flow control mechanism. When the port is operating in fullDuplex mode, the port uses IEEE 802.3x flow control mechanism.  
 If the port type is hundredBaseFX:  
 When the port is operating in halfDuplex mode, the port uses backPressure flow control mechanism. When the port is operating in fullDuplex mode, Flow control mechanism will not function.  
 (2) Flow control mechanism is disabled.  
 (3) Flow control mechanism is backPressure.  
 when the port is in fullDuplex mode.This flow control mechanism will not function.  
 (4) Flow control mechanism is IEEE 802.3x flow control.  
 when the port is in halfDuplex mode.This flow control mechanism will not function.  
 hundredBaseTX and thousandBaseSX port can be set as:  
 enabled(1),  
 disabled(2),  
 backPressure(3),  
 dot3xFlowControl(4).  
 hundredBaseFX port can be set as:  
 enabled(1),  
 disabled(2),  
 backPressure(3).  
 The actual flow control mechanism is used given by swPortMgtFlowCtrlInUse."  
 DEFVAL { enabled }  
 ::= { switchPortMgtEntry 5 }

swPortMgtFlowCtrlInUse OBJECT-TYPE

SYNTAX INTEGER  
 {  
 backPressure(1),  
 dot3xFlowControl(2),  
 none(3)  
 }

MAX-ACCESS read-only

STATUS current

DESCRIPTION "(1) BackPressure flow control machanism is used.  
 (2) IEEE 802.3 flow control machanism is used.

```

        (3) Flow control mechanism is disabled. "
        ::= { switchPortMgtEntry 6 }

--
-- systemSTAMgt
--

systemSTAStatus OBJECT-TYPE
    SYNTAX      INTEGER
                {
                    enabled(1),
                    disabled(2)
                }
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION  "Global spanning tree status.
                  (1) Spanning tree protocol is enabled.
                  (2) Spanning tree protocol is disabled. "
    DEFVAL       { enabled }
    ::= { systemSTAMgt 1 }

--
-- tftpDownloadMgt
--

tftpDownloadServerIP OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION  "The IP address of a TFTP server from which a
                  firmware image can be downloaded."
    DEFVAL { '00000000'H }
    ::= { tftpDownloadMgt 1 }

--tftpDownloadMainBoardFwFileName OBJECT-TYPE
--
--    SYNTAX      DisplayString (SIZE(0..80))
--    MAX-ACCESS   read-write
--    STATUS       current
--    DESCRIPTION  ""
--    DEFVAL       { "" }
--    ::= { tftpDownloadMgt 2 }

--tftpDownloadMainBoardFwSelected OBJECT-TYPE
--
--    SYNTAX      INTEGER

```

```

--          {
--              selected(1),
--              notSelected(2)
--          }
--      MAX-ACCESS  read-write
--      STATUS      current
--      DESCRIPTION "Setting this object as selected. The system will
download
--                  main board firmware when the download action be
triggered."
--      DEFVAL      { notSelected }
--      ::= { tftpDownloadMgt 3 }

```

```

tftpDownloadAgentBoardFwFileName OBJECT-TYPE
    SYNTAX      DisplayString(SIZE(0..80))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION ""
    DEFVAL      { "" }
    ::= { tftpDownloadMgt 2 }

```

```

tftpDownloadAgentBoardFwDownloadMode OBJECT-TYPE
    SYNTAX      INTEGER
                {
                    permanent(1),
                    temporary(2)
                }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION "Indicates whether a newly upgraded firmware
version should write to flash. When this object
is temporary(2), following a successful upgrade
the system will switch to run the new firmware but
will not upgrade the new firmware to flash. That
means after a power cycle, system will run the
firmware residing the flash.
When this object is permanent(1), following a
successful firmware upgrade, the flash will be
upgraded and the system will automatically switch
to run the new firmware."
    DEFVAL      { permanent }
    ::= { tftpDownloadMgt 3 }

```

```

--tftpDownloadAgentBoardFwSelected OBJECT-TYPE

```

```

--          SYNTAX          INTEGER
--                               {
--                               selected(1),
--                               notSelected(2)
--                               }
--          MAX-ACCESS      read-write
--          STATUS          current
--          DESCRIPTION      "Setting this object as selected. The system will
download
--                               agent board firmware when the download action be
triggered."
--          DEFVAL          { notSelected }
--          ::= { tftpDownloadMgt 4 }

```

```

tftpDownloadStatus OBJECT-TYPE
    SYNTAX          INTEGER
                    {
                    active(1),
                    notActive(2)
                    }
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION      "Setting this object to active(1) trigger the TFTP
download action.
Setting this object to notActive(2) has no effect.
The system always returns the value notActive(2)
when this object is read."
    ::= { tftpDownloadMgt 4 }

```

```

--
-- restartMgt
--

```

```

restartOptionPOST OBJECT-TYPE
    SYNTAX          INTEGER
                    {
                    enabled(1),
                    disabled(2)
                    }
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION      "Setting this object as enabled. The system will do POST
when it restart"
    DEFVAL          { enabled }

```

::= { restartMgt 1 }

restartOptionReloadFactoryDefault OBJECT-TYPE

SYNTAX INTEGER

{  
 enabled(1),  
 disabled(2)  
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION "Setting this object as enabled. The system will do factory  
reset when it restart"

DEFVAL { disabled }

::= { restartMgt 2 }

restartOptionKeepIpSetting OBJECT-TYPE

SYNTAX INTEGER

{  
 enabled(1),  
 disabled(2)  
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION "Setting this object as enabled. The system will keep IP  
setting when it do factory reset."

DEFVAL { disabled }

::= { restartMgt 3 }

restartOptionKeepUserAuthentication OBJECT-TYPE

SYNTAX INTEGER

{  
 enabled(1),  
 disabled(2)  
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION "Setting this object as enabled. The system will keep user  
authentication setting when it do factory reset."

DEFVAL { disabled }

::= { restartMgt 4 }

restartAction OBJECT-TYPE

SYNTAX INTEGER

{

```

        active(1),
        notActive(2)
    }
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Setting this object to active(1) trigger the system
            restart.
            Setting this object to notActive(2) has no effect.
            The system always returns the value notActive(2)
            when this object is read."
::= { restartMgt 5 }

--
-- portMirrorMgt
--

portMirrorStatus OBJECT-TYPE
    SYNTAX INTEGER
    {
        enabled(1),
        disabled(2)
    }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION "Port mirroring function status.
                (1) mirroring function is enabled.
                (2) mirroring function is disabled."
    ::= { portMirrorMgt 1 }

portMirrorSnifferPort OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION "Linear port number of sniffer port to which all
                frames to/from mirrored ports are sent. Frames
                are only mirrored if the portMirrorStatus object
                is set to enabled(1)."
    ::= { portMirrorMgt 2 }

portMirrorMirroredPort OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-write
    STATUS current

```

DESCRIPTION "Linear port number of mirrored port. The traffic of  
mirrored port will be 'copied' to sniffer port."  
::= { portMirrorMgt 3 }

--  
-- igmpMgt  
--

igmpStatus OBJECT-TYPE

SYNTAX INTEGER  
{  
enabled(1),  
disabled(2)  
}  
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION "Parameter to enable or disable IGMP snooping on the device.  
When enabled, the device will examine IGMP packets and set  
up filters for IGMP ports. "  
DEFVAL { enabled }  
::= { igmpMgt 1 }

igmpQueryCount OBJECT-TYPE

SYNTAX INTEGER (2..10)  
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION "Maximum number of queries that have not been heard on the  
system before the system starts taking action to solicit  
reports."  
DEFVAL { 2 }  
::= { igmpMgt 2 }

igmpReportDelay OBJECT-TYPE

SYNTAX INTEGER (5..30)  
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION "Timeout value (seconds) between IGMP reports received on a  
port  
for an IP Multicast Address that can pass before the system  
sends an IGMP Query out the port and removes it from the  
list."  
DEFVAL { 10 }  
::= { igmpMgt 3 }

END