

Appendix A

Provisioning Server IDLs

This appendix describes the Juniper Networks proprietary IDL set for the Provisioning Service. The IDLs define all of the methods for the network management system (NMS) to manage Juniper Networks devices. To get these references, the NMS must go through the `getManager(...)` of `emsSession` reference.

The provisioning objects of each IDL are also described.

This appendix contains the following sections:

- `unspRXDeviceMgr.idl` on page 231
- `unspRXCustomerMgr.idl` on page 249

unspRXDeviceMgr.idl

The following IDL is the definition of `unspRXDeviceMgr.idl`:

```
#ifndef unspRXDeviceMgr_idl
#define unspRXDeviceMgr_idl

// *****
// *
// * unspRXDeviceMgr.idl
// *
// *****

//Include list
#include "globaldefs.idl"
#include "common.idl"

#pragma prefix "nmcRX.unspCORBAbridge.provisioningservice.services.redstonecom.com"

module unspRXDeviceMgr
{
    /**
     * UnspRXDeviceMgrObjectType defined all possible object types for NMC-RX to manage
     */
    enum UnspRXDeviceMgrObjectType
    {
        // Layered Objects
        EMS,
        ManagedElement,
        PTP,
        AtmInterface,
        // Examples:
        // "Juniper NMC-RX"
        // "10.6.129.6"
        // "6/0" - where 6 is the slot number, and
        // 0 is the port number
        // "ATM6/0-atm-layer",
    }
}
```

```

    AtmSubInterface,          // "ATM6/0.1-atm1483-subif",
    AtmPvc,                  // "ATM6/0.1-atm1483-subif vpi=1,vci=600"

    PPP,                    // not supported
    PPPoE,                 // not supported
    PPPoESubInterface,     // not supported
    BridgedIP_rfc1483,     // "4/0.10"
    VirtualRouter,        // "default", "VR1"

    IpInterface,

    Customer,
    CustSite,
    BridgeInterface        // "BridgeIf6/0.1"
};

/**
 * UnspRXDeviceMgrObjectTypeList is a list UnspRXDeviceMgrObjectType
 */
typedef sequence<UnspRXDeviceMgrObjectType> UnspRXDeviceMgrObjectTypeList;

/**
 * UnspRXActionType defined all possible action types
 * for NMC-RX to manage E-series routers
 */
enum UnspRXActionType
{
    CREATE,
    CONFIGURE,
    LIST,
    VIEW,
    DELETE
};

/**
 * UnspRXActionTypeList is a list UnspRXActionType
 */
typedef sequence<UnspRXActionType>UnspRXActionTypeList;

/**
 * UnspRXDeviceMgr_I defines all methods to manage objects
 * under ERX. It is derived from common::Common_I, which is
 * one of the ids defined by TMF 814.
 */

interface UnspRXDeviceMgr_I : common::Common_I
{
    /**
     * This method should be used to list all objects of a particular type relative
     * to another object.
     *
     * The retrieval mechanism follows the paradigm established in TMF 814
     * and is described below:
     *
     * objectPath:    The name-value pair list of the objects you want to list relative to other
objects.
     *
     * Example:
     * ListAll AtmSubInterfaces relative to objectPath
     *     Name="EMS",           Value="Juniper NMC-RX"
     *     Name="ManagedElement", Value="10.6.129.6"
     *     Name="PTP",           Value="6/0"
     *     Name="AtmInterface",  Value="ATM6/0-atm-layer"

```

```

*
* ListAll AtmPvc relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"
*
* objectType:   The type of objects that you want to list
* Example:
*   UnspRXDeviceMgrObjectType.AtmSubInterfaces
*   UnspRXDeviceMgrObjectType.AtmPvc
*
* howMany:      The maximum number of objects to report in the first batch in nameList
*
* nameList:     First batch of howMany objects report to NMS
*
* nameIt:       Iterator to retrieve the remaining object after the first batch
* To allow the NMS to deal more easily with retrievals which may return a
* very large amount of data, iterators are used. They provide a mechanism to retrieve
* data batch by batch - the size of a batch is specified by the NMS via the
* how_many parameter in the next_n method of the iterator reference.
* For a more detailed explanation of the use of iterators, please refer to
* TMF814\supportingDocumentation\iterators.html
* Raises: globaldefs::ProcessingFailureException
* EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
* EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
* EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
* EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
* can support has been reached.
* EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void listAll(
    in globaldefs::NVList_T          objectPath,
    in UnspRXDeviceMgrObjectType    objectType,
    in long                          howMany,
    out globaldefs::NamingAttributesList_T nameList,
    out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);

/**
* This list works the same as the other one except that it returns only those objects created
* after timeStamp. Also, objects are sorted by timeStamp when they are returned.
*/
void listAllFromTime(
    in globaldefs::NVList_T          objectPath,
    in UnspRXDeviceMgrObjectType    objectType,
    in string                        timeStamp,
    in long                          howMany,
    out globaldefs::NamingAttributesList_T nameList,
    out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to create an object of a particular type.
*
* objectPath:   The name-value pair list of the objects you want to create.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"
*   Name="AtmInterface",  Value="ATM6/0-atm-layer"
*   Name="AtmSubInterfaces", Value="ATM6/0.1-atm1483-subif"
*   Name="AtmPvc",        Value="ATM6/0.1-atm1483-subif vpi=1,vci=600"

```

```

*
* objectType: The type of objects that you want to list
* Example:
*   UnspRXDeviceMgrObjectType.AtmPvc
*
* attributesList: It will hold all attributes as a name-value pair list
*                 to pass to NMC-RX to create this object
*
*
* objectName: The object name has just been created by this method. It is assigned
*              by device instead of being specified by the user
*
* Raises: globaldefs::ProcessingFailureException
*         EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*         EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*         EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*         EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                         can support has been reached.
*         EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void create(
    in globaldefs::NVSLIST_T      objectPath,
    in UnspRXDeviceMgrObjectType objectType,
    in globaldefs::NVSLIST_T      attributesList,
    out string                    objectName )
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to configure objects
*
* objectPath: The name-value pair list of the object you want to configure.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"
*   Name="AtmInterface",  Value="ATM6/0-atm-layer"
*   Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*   Name="AtmPvc",        Value="ATM6/0.1-atm1483-subif vpi=1,vci=600"
*
* attributesList: It will hold all attributes as name value pair list
*                 to pass to NMC-RX to configure this object.
* Raises: globaldefs::ProcessingFailureException
*         EXCPT_INTERNAL_ERROR - Raised in case of a nonspecific EMS internal failure
*         EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*         EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*         EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                         can support has been reached.
*         EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void config(
    in globaldefs::NVSLIST_T      objectPath,
    in globaldefs::NVSLIST_T      attributesList)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to delete an object
*
* objectPath: The name-value pair list of the object you want to delete.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"

```

```

*         Name="AtmInterface",    Value="ATM6/0-atm-layer"
*         Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*         Name="AtmPvc",          Value="ATM6/0.1-atm1483-subif vpi=1,vci=600"
*   Raises: globaldefs::ProcessingFailureException
*     EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*     EXCPT_INVALID_INPUT  - Raised when the invalid input is passed into this method
*     EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*     EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                       can support has been reached.
*     EXCPT_NE_COMM_LOSS  - Raised when communication to managedElement is lost
**/
void delete(in globaldefs::NVSLIST_T  objectPath)
    raises(globaldefs::ProcessingFailureException);

/**
*   This method should be used to view an object
*
*   objectPath:    The name-value pair list of the object you want to view.
*   Example:
*     ListAll AtmSubInterfaces relative to objectPath
*     Name="EMS",          Value="Juniper NMC-RX"
*     Name="ManagedElement", Value="10.6.129.6"
*     Name="PTP",          Value="6/0"
*     Name="AtmInterface", Value="ATM6/0-atm-layer"
*     Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*     Name="AtmPvc",      Value="ATM6/0.1-atm1483-subif vpi=1,vci=600"
*
*   attributesList: It will hold all attributes as the name value pair list
*                   to pass back to NMS to view this object.
*   Raises: globaldefs::ProcessingFailureException
*     EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*     EXCPT_INVALID_INPUT  - Raised when the invalid input is passed into this method
*     EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*     EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                       can support has been reached.
*     EXCPT_NE_COMM_LOSS  - Raised when communication to managedElement is lost
**/
void view(
    in globaldefs::NVSLIST_T      objectPath,
    out globaldefs::NVSLIST_T     attributesList)
    raises(globaldefs::ProcessingFailureException);

/**
*   This method should be used to get the attributes descriptions in order
*   to create an object
*
*   objectPath:    The name-value pair list of the object you want to view.
*   Example:
*     ListAll AtmSubInterfaces relative to objectPath
*     Name="EMS",          Value="Juniper NMC-RX"
*     Name="ManagedElement", Value="10.6.129.6"
*     Name="PTP",          Value="6/0"
*     Name="AtmInterface", Value="ATM6/0-atm-layer"
*     Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*
*   objectType:    The type of objects that you want to create
*   Example:
*     UnspRXDeviceMgrObjectType.AtmPvc
*
*   attributesDescription: It will hold all attributes with their description
*                          as name-value pair list to pass back to NMS.
*                          The name will be the attribute name, but the value
*                          will have the attribute type and the description of

```

```

*                                     this attribute, seperated by the first ":".
* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*   EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*   EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*   EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                     can support has been reached.
*   EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getCreateAttributesDescription(
    in globaldefs::NVSLIST_T      objectPath,
    in UnspRXDeviceMgrObjectType  objectType,
    out globaldefs::NVSLIST_T     attributesDescription)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get the attributes description in order
* to config an object
*
* objectPath:    The name-value pair list of the object you want to view.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"
*   Name="AtmInterface",  Value="ATM6/0-atm-layer"
*   Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*
* attributesDescription: It will hold all attributes with their description
*                         as name-value pair list to pass back to NMS.
*                         The name will be the attribute name, but the value
*                         will have the attribute type and the description of
*                         this attribute, seperated by the first ":".
* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*   EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*   EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*   EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                     can support has been reached.
*   EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getConfigAttributesDescription(
    in globaldefs::NVSLIST_T      objectPath,
    out globaldefs::NVSLIST_T     attributesDescription)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get a list of object types on which you can perform a list of
* actions relative to other objects.
*
* objectPath:    The name-value pair list of the object you want to view.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*   Name="EMS",           Value="Juniper NMC-RX"
*   Name="ManagedElement", Value="10.6.129.6"
*   Name="PTP",           Value="6/0"
*   Name="AtmInterface",  Value="ATM6/0-atm-layer"
*   Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*
* objectTypeList: To hold a list of object types
*
* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure

```

```

*      EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*      EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*      EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                     can support has been reached.
*      EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getListableObjectTypeSelections(
    in globaldefs::NVSLIST_T      objectPath,
    out UnspRXDeviceMgrObjectList objectTypeList)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get a list of object types on which you can create
* actions relative to other objects.
*
* objectPath:    The name-value pair list of the object you want to view.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*     Name="EMS",           Value="Juniper NMC-RX"
*     Name="ManagedElement", Value="10.6.129.6"
*     Name="PTP",           Value="6/0"
*     Name="AtmInterface",  Value="ATM6/0-atm-layer"
*     Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*
* objectTypeList: To hold a list of object types
*
* Raises: globaldefs::ProcessingFailureException
*     EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*     EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*     EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*     EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                     can support has been reached.
*     EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getCreatableObjectTypeSelections(
    in globaldefs::NVSLIST_T      objectPath,
    outUnspRXDeviceMgrObjectList objectTypeList)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get a list of types of actions you can perform
* relative to other objects.
*
* objectPath:    The name-value pair list of the objects you want to view.
* Example:
* ListAll AtmSubInterfaces relative to objectPath
*     Name="EMS",           Value="Juniper NMC-RX"
*     Name="ManagedElement", Value="10.6.129.6"
*     Name="PTP",           Value="6/0"
*     Name="AtmInterface",  Value="ATM6/0-atm-layer"
*     Name="AtmSubInterfaces",Value="ATM6/0.1-atm1483-subif"
*
* objectTypeList: To hold a list of action types
*
* Raises: globaldefs::ProcessingFailureException
*     EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*     EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*     EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*     EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                     can support has been reached.
*     EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getSupportedActions(

```

```

        in globaldefs::NVSLIST_T objectPath,
        outUnspRXActionTypeList actionTypeList)
        raises(globaldefs::ProcessingFailureException);

    };
};
#endif

```

Provisioning Objects

The following sections describe the provisioning objects for this IDL.

Managed Element

A managed element is an E-series device or node.

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = ManagedElement; Value = 10.10.10.10
- Associated MIBs
 - junisystem.mi2
 - rfc1213.mib

Table 75: Attributes (Managed Element)

Attribute	Characteristic	Description	MIB Entry
SYS_CONTACT	View, configure	Contact person for the device	sysContact
SYS_LOCATION	View	Physical location of the device	sysLocation
SYS_SWVERSION	View	Software version currently running on the device	juniSystemSwVersion
SYS_SW_BUILD_DATE	View	Build date of the software currently running on the system	juniSystemSwBuildDate
SYS_UPTIME	View	Current system uptime for the device	sysUpTime
SYS_DEVICE_TYPE	View	Type of device Values: <ul style="list-style-type: none"> ■ 1 = ERX 700 ■ 2 = ERX 1400 	None
ERX_NODE_SYS_NAME	View	ERX node system name for the device The name is displayed on the prompt of the command-line interface. It is not the same as the name assigned by the NMC-RX user when initially discovering the node.	sysName

PTP

The Physical Termination Point (PTP) object represents either a module specified at a particular slot, or a line interface specified as a slot and port.

- Sample Object Path

The object path for a module includes the slot number.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0

The object path for a line interface or port includes the slot number and the port number.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0/0

- Associated MIBs

- juniSystem.mi2

Table 76: Attributes (PTP)

Attribute	Characteristic	Description	MIB Entry
ADMIN_STATUS	Configure	Administrative status of the module	juniSystemModuleAdminStatus. <slotNumber>.1
ENTITY_INDEX	View	SNMP Index used to represent the object in the entity mib	entPhysicalIndex
IF_INDEX	View	Index used for SNMP management	ifIndex
SERIAL_NUMBER	View	Module Serial Number	juniSystemModuleSerialNumber. <slotNumber>.1
IOA_SERIAL_NUMBER	View	Module IOA Serial Number	juniSystemModuleSerialNumber. <slotNumber>.2
MODULE_TYPE	View	Module type	juniSystemModuleCurrentType. <slotNumber>.1

Port (OC3/OC12)

This type of provisioning object has the following associated MIBs:

- rfc1213.mib
- juniSonet.mi2

Table 77: Attributes (Port: OC3/OC12)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
DESCRIPTION	Configure	Logical description of the interface	ifAlias
ADMIN_STATUS	Configure	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus
SONET_INDEX	View	Sonet SNMP Index used to represent the object	ifIndex
LOOPBACK	Configure	Specifies the loopback mode for the interface	juniSonetMediumLoopbackConfig
SYNC_STANDARD	Configure	Specifies the Sync Standard (SDH or Sonet) to be used for the interface	juniSonetMediumType
XMIT_CLOCK	Configure	Specifies the Transmit Clock Source to use for the interface	juniSonetMediumTimingSource
LINE_INTERFACE_TYPE	View	Line interface type	3 = OC3 2-port 12 = OC12 ATM 1-port 14 = OC3 ATM 4-port

Port (T3 ATM)

This type of provisioning object has the following associated MIBs:

- rfc1213.mib
- juniDs3.mi2
- rfc1407.mib

Table 78: Attributes (Port: T3 ATM)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
DESCRIPTION	Configure	Logical description of the interface	ifAlias
ADMIN_STATUS	Configure	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus
LINE_BUILDOUT	Configure	Specifies the line build out in meters for the interface	juniDs3LineLength
FRAMING_TYPE	Configure	Specifies the framing to be used for the interface	juniDs3LineType
LOOPBACK	Configure	Specifies the loopback mode for the interface	juniSonetMediumLoopbackConfig

Table 78: Attributes (Port: T3 ATM) (continued)

Attribute	Characteristic	Description	MIB Entry
SCRAMBLE	Configure	Specifies the Scramble mode to use for the interface	juniDs3CellScramblerConfig
XMIT_CLOCK	Configure	Specifies the Transmit Clock Source to use for the interface	dsx3TransmitClockSource
LINE_INTERFACE_TYPE	View	Line interface type	4 = T3 ATM 3-port

ATM Interface

The OC3-2 modules do not support MAX VPI and MAX VCI.

■ Sample Object Path

The object path for a module includes the slot number.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0/0
- Name = AtmInterface; Value = ATM0/0

■ Associated MIBs

- rfc1213.mib
- rfc1695.mi2
- juniAtm.mi2
- af-nm-0095_001_mib.mi2
(atmfM4IfLoopbackLocationCode defined in this MIB)

Table 79: Attributes (ATM Interface)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
DESCRIPTION	Create, configure	Logical description of the interface	ifAlias
ADMIN_STATUS	Configure	Administrative status of the module	juniSystemModuleAdminStatus. < slotNumber > . 1
OPER_STATUS	View	Operational status of the interface	ifOperStatus
MAX_VPI_BITS	Create	Maximum configurable value for a VPI on the interface	atmInterfaceMaxActiveVpiBits
MAX_VCI_BITS	Create	Maximum configurable value for a VCI on the interface	atmInterfaceMaxActiveVciBits
ILMI_VPI	Create, configure	VPI of the ILMI of a major ATM interface	juniAtmIfIlmiVpi
ILMI_VCI	Create, configure	VCI of the ILMI of a major ATM interface	juniAtmIfIlmiVci

Table 79: Attributes (ATM Interface) (continued)

Attribute	Characteristic	Description	MIB Entry
ILMI_VCD	Create	An integer identifier for the ILMI, used in conjunction with the command-line interface. The value must be unique among VCs configured on the same ATM interface.	juniAtmIfIlmiVcd
ILMI_ADMIN_STATUS	Create, configure	Administrative status of ILMI in the ATM interface	juniAtmIfIlmiAdminState
ILMI_POLL_FREQUENCY	Create, configure	The amount of time in seconds between successive transmissions of ILMI messages on the interface for the purpose of detecting loss of ILMI connectivity. The distinguished value zero disables ILMI connectivity procedures on the interface.	juniAtmIfIlmiPollFrequency
UNI_VERSION	Create, configure	Use to specify the User Network Interface (UNI) the router should use when ILMI link auto determination is unsuccessful or ILMI is disabled.	juniAtmIfUniVersion
OAM_ADMIN_STATE	Create, configure	Administrative status of OAM for the interface	juniAtmIfOamCellRxAdminState
CAC_ADMIN_STATE	Create, configure	Administrative status of CAC on the ATM major interface	juniAtmIfCacAdminState
CAC_UBR_WEIGHT	Create, configure	Bandwidth associated with every UBR and UBR with PCR connection configured on the ATM major interface	juniAtmIfCacUbrWeight
CAC_SUBSCRIPTION_BANDWIDTH	Create, configure	The subscribed bandwidth of the ATM major interface. If this value is not specified or set to 0, the effective port bandwidth is used.	juniAtmIfCacSubscriptionBandwidth
CAC_AVAILABLE_BANDWIDTH	View	Available bandwidth of the ATM major interface	juniAtmIfCacAvailableBandwidth
LOOPBACK_LOCATION_ID	Create, configure	Specifies the code that shall exist in incoming OAM Loopback cells that are to be looped back at the interface	atmfM4IfLoopbackLocationCode

ATM Subinterface

■ Sample Object Path

The object path for a module includes the slot number.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0/0
- Name = AtmInterface; Value = ATM0/0
- Name = AtmSubInterface; Value = ATM0/0.33

- Associated MIBs
 - rfc1213.mib
 - juniAtm.mi2
 - juniAutoconf.mi2
 - juniSubscriber.mi2
 - juniTpl.mi2
 - juniQos.mi2

Table 80: Attributes (ATM Subinterface)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	The index used for SNMP management	ifIndex
NAME	View	The name of the interface	ifDescr
DESCRIPTION	Create, configure	The logical description of the interface	ifAlias
ADMIN_STATUS	Create, configure	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus
IP_AUTOCONFIG	Create, configure	Enables the auto configuration feature for the IP layer	juniAutoConfEnable.ifIndex.ip
PPP_AUTOCONFIG	Create, configure	Enables the auto configuration feature for the PPP layer	juniAutoConfEnable.ifIndex.ppp
PPPOE_AUTOCONFIG	Create, configure	Enables the auto configuration feature for the PPPoE layer	juniAutoConfEnable.ifIndex.pppoe
SUBSCRIBER_ENABLE	Create, configure	Enables appending of subscriber information for the IP layer	juniSubscrLocalControl
SUBSCRIBER_NAME_PREFIX	Create, configure	Indicates whether the Subscriber Name is a prefix rather than a full name	juniSubscrLocalNamePrefix
SUBSCRIBER_PASSWORD_PREFIX	Create, configure	Indicates whether the Subscriber Password is a prefix rather than a full password	juniSubscrLocalPasswordPrefix
IP_PROFILE_NAME	Create, configure	Name of the Profile to associate with the IP Encapsulation	None
IP_PROFILE_INDEX	View	SNMP Index of the Profile to associate with the IP Encapsulation	juniProfAssignIfProfileId.ifIndex.ip
PPP_PROFILE_NAME	Create, configure	Name of the Profile to associate with the PPP Encapsulation	None
PPP_PROFILE_INDEX	View	SNMP Index of the Profile to associate with the PPP Encapsulation	juniProfAssignIfProfileId.ifIndex.ppp
PPPOE_PROFILE_NAME	Create, configure	Name of the Profile to associate with the PPPoE Encapsulation	None

Table 80: Attributes (ATM Subinterface) (continued)

Attribute	Characteristic	Description	MIB Entry
PPPOE_PROFILE_INDEX	View	SNMP Index of the Profile to associate with the PPPoE Encapsulation	juniProfAssignIfProfileId.ifIndex.pppoe
ANY_PROFILE_NAME	Create, configure	Name of the Profile to be used as a wildcard	None
ANY_PROFILE_INDEX	View	SNMP Index of the Profile to be used as a wildcard	juniProfAssignIfProfileId.ifIndex.any
SUBSCRIBER_NAME	Create, configure	Local name that distinguishes the subscriber	juniSubscrLocalName
SUBSCRIBER_PASSWORD	Create, configure	Local password to be used for the subscriber	juniSubscrLocalPassword
SUBSCRIBER_DOMAIN	Create, configure	Subscriber's domain	juniSubscrLocalDomain
QOS_PROFILE_INDEX	View	SNMP index of the attached QoS Profile	juniQosIfAttachQosProfileIndex
QOS_PROFILE_NAME	Create, configure	Name of the attached QoS Profile	None

ATM PVC

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = ManagedElement; Value = 10.10.10.10
 - Name = PTP; Value = 0/0
 - Name = AtmInterface; Value = ATM0/0
 - Name = AtmSubInterface; Value = ATM0/0.33
 - Name = AtmPVC; Value = ATM0/0.33.44.55
- Associated MIBs
 - juniAtm.mi2

Table 81: Attributes (ATM PVC)

Attribute	Characteristic	Description	MIB Entry
NAME	View	Name of the interface	ifDescr
VPI	Create	Virtual path identifier	INDEX
VCI	Create	Virtual circuit identifier	INDEX
AAL5_ENCAP	Create	Encapsulation for the ATM Adaptation Layer 5 (AAL5)	juniAtmSubIfVccType
MBS	Create, configure	Maximum burst size (in cells)	juniAtmSubIfVccMbs
PCR	Create, configure	Peak cell rate (kbps)	juniAtmSubIfVccPcr
SCR	Create, configure	Sustainable cell rate (kbps)	juniAtmSubIfVccScr

Table 81: Attributes (ATM PVC) (continued)

Attribute	Characteristic	Description	MIB Entry
SERVICE_CATEGORY	Create, configure	Servicecategory	juniAtmSubIfVccServiceCategory
OAM_STATUS	Create, configure	Enables generation of OAM F5 loopback cells	juniAtmCircuitOamAdminStatus
OAM_LOOPBACK	Create, configure	Time interval in seconds between transmissions of OAM F5 loopback cells	juniAtmCircuitOamLoopbackFrequency
VCD	View	Virtual circuit descriptor	juniAtmSubIfVccVcd

Virtual Router

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = ManagedElement; Value = 10.10.10.10
 - Name = VirtualRouter; Value = default
 - Name = AtmPVC; Value = ATM0/0.33.44.55
- Associated MIBs
 - junRouter.mi2
 - junSscClient.mi2
 - junAaa.mi2
 - junRadClient.mi2
 - junDhcp.mi2

Table 82: Attributes (Virtual Router)

Attribute	Characteristic	Description	MIB Entry
VROUTER_NAME	View	Name of the virtual router	juniRouterName
SSCC_PRIMARY_PORT	View	TCP port number for the primary SSCC server. A value of zero indicates the port is unconfigured.	juniSscClientPrimaryPort
SSCC_SECONDARY_PORT	View	TCP port number for the secondary SSCC server. A value of zero indicates the port is unconfigured.	juniSscClientSecondaryPort
SSCC_TERTIARY_PORT	View	TCP port number for the tertiary SSCC server. A value of zero indicates the port is unconfigured.	juniSscClientTertiaryPort
SSCC_SWITCHOVER_TIMEOUT	View	Server switchover timeout in seconds. The SSCCClient begins with the primary server and proceeds rotationally to secondary, tertiary, primary, etc., as timeouts occur.	juniSscClientServerSwitchoverTimeout

Table 82: Attributes (Virtual Router) (continued)

Attribute	Characteristic	Description	MIB Entry
SSCC_SNMP_READWRITE_COMMUNITY	View	SSCC SNMP Read Write community string	None
SSCC_SNMP_READ_COMMUNITY	View	The SSCC SNMP read only Community String	None
SSCC_PRIMARY_ADDRESS	View	IP address of the primary SSCC server. A value of 0.0.0.0 indicates the server address is unconfigured.	juniSscClientPrimaryAddress
SSCC_SECONDARY_ADDRESS	View	IP address of the secondary SSCC server. A value of 0.0.0.0 indicates the server address is unconfigured.	juniSscClientSecondaryAddress
SSCC_TERTIARY_ADDRESS	View	IP address of the tertiary SSCC server. A value of 0.0.0.0 indicates the server address is unconfigured.	juniSscClientTertiaryAddress
PROTOCOL_STATUS_SSCC_CLIENT	View	Status of the SSCC Client Protocol	juniRouterProtocolRowStatus

IP Interface

IP interfaces are stacked on a wide variety of interface types and configurations. Object paths reflect these various configurations.

- Sample Object Path

The following shows the object path for an IP interface that is stacked on top of an ATM subinterface.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0/0
- Name = AtmInterface; Value = ATM0/0
- Name = AtmSubInterface; Value = ATM0/0.33
- Name = IpInterface; Value = IP0/0.33

- Associated MIBs

- junIp.mi2

Table 83: Attributes (IP Interface)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
DESCRIPTION	View	Logical description of the interface	ifAlias
ADMIN_STATUS	View	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus

Table 83: Attributes (IP Interface) (continued)

Attribute	Characteristic	Description	MIB Entry
CUSTOMER_NAME	View	Name of the customer	None
CUSTOMER_IDENTITY	View	Unique identifier for a particular customer	None
CUSTOMER_ACCOUNT_ID	View	Account ID for the customer	None
CUSTOMER_SITE_NAME	View	Name to represent the site for a particular customer	None
CUSTOMER_SITE_IDENTITY	View	Unique identifier to represent the site for a particular customer	None
CUSTOMER_SITE_ACCOUNT_ID	View	Account ID to represent the site for a particular customer	None
IP_CREATE_TIME	View	Timestamp of the creation of the IP Interface. Creation is considered to be relevant to the NMC-RX application. If the interface is created via the CLI then the timestamp will reflect the first time the NMC-RX application was aware of the interface, not the time the interface was created on the CLI.	None

Customer and Customer Site

Customer objects can be associated with IP interfaces. The Device Manager provides methods for managing customer and customer site objects. These objects are discussed in “unspRXCustomerMgr.idl” on page 249.

ATM Bridge Support

There are two object types added to support Bridge Interfaces over ATM: BridgedIP_rfc1483 (Bridged Ethernet) and Bridge Interface.

■ Sample Object Path

The following shows the object path for an IP interface that is stacked on top of an ATM subinterface.

- Name = EMS; Value = JUNIPER_NMCRX
- Name = ManagedElement; Value = 10.10.10.10
- Name = PTP; Value = 0/0
- Name = AtmInterface; Value = ATM0/0
- Name = AtmSubInterface; Value = ATM0/0.33
- Name = BridgedIP_rfc1483; Value = BRG-ET0/0.33

■ Associated MIBs

- RFC1213.mi2

Table 84: Attributes (ATM Bridge Support)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
DESCRIPTION	Create, configure	Logical description of the interface	ifAlias
ADMIN_STATUS	Create, configure	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus

Bridge Interface

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = ManagedElement; Value = 10.10.10.10
 - Name = PTP; Value = 0/0
 - Name = AtmInterface; Value = ATM0/0
 - Name = AtmSubInterface; Value = ATM0/0.33
 - Name = BridgedIP_rfc1483; Value = BRG-ET0/0.33
 - Name = BridgeInterface; Value = BridgeIf0/0.33
- Associated MIBs
 - juniBridge.mi2
 - juniBridgingMgr.mi2
 - rfc1213.mi2

Table 85: Attributes (Bridge Interface)

Attribute	Characteristic	Description	MIB Entry
IF_INDEX	View	Index used for SNMP management	ifIndex
NAME	View	Name of the interface	ifDescr
ADMIN_STATUS	Create, configure	Administrative status of the interface	ifAdminStatus
OPER_STATUS	View	Operational status of the interface	ifOperStatus
BRIDGE_GROUP_NAME	Create	Name of the associated bridge group	juniBridgingMgrBridgeGroupName
MAX_LEARN_COUNT	Create, configure	Maximum number of MAC addresses that can be learned on the interface	juniBridgeIfMaxLearnCount
SUBSCRIBER_POLICY	Create, configure	Indicates whether the type of the interface is subscriber (value = 1) or subscriber trunk (value = 2)	juniBridgingMgrBridgeGroupSPolicyIndex
BRIDGE_GROUP_INDEX	View	Index of the associated bridge group	ifIndex of the bridge group the interface is referencing

unspRXCustomerMgr.idl

The following IDL is the definition of `unspRXCustomerMgr.idl`:

```
#ifndef unspRXCustomerMgr_idl
#define unspRXCustomerMgr_idl

// *****
// *
// * unspRXCustomerMgr.idl *
// *
// *****

//Include list
#include "globaldefs.idl"
#include "common.idl"

#pragma prefix "nmcRX.unspcorbabridge.provisioningservice.services.redstonecom.com"

module unspRXCustomerMgr
{
    /**
     * UnspRXCustomerMgrObjectType defined all possible object types for NMC-RX to manage
     */
    enum UnspRXCustomerMgrObjectType
    {
        // Layered Objects           // Examples:
        EMS,                        // "Juniper NMC-RX"
        Customer,                   // "ABC Store Customer"
        CustSite                     // "Westford Customer Site"
    };

    /**
     * UnspRXCustomerMgrObjectTypeList is a list UnspRXCustomerMgrObjectType
     */
    typedef sequence<UnspRXCustomerMgrObjectType> UnspRXCustomerMgrObjectTypeList;

    /**
     * UnspRXActionType defined all possible action types
     * for NMC-RX to manage ERX/MRX
     */
    enum UnspRXCustomerActionType
    {
        LIST,
        VIEW
    };

    /**
     * UnspRXActionTypeList is a list UnspRXActionType
     */
    typedef sequence<UnspRXCustomerActionType>UnspRXCustomerActionTypeList;

    /**
     * UnspRXCustomerMgr_I defines all methods to manage objects
     * under ERX/MRX. It is derived from common::Common_I, which is
     * one of the idls defined by TMF 814.
     */
}
```

```

interface UnspRXCustomerMgr_I : common::Common_I
{
    /**
     * This method should be used to list all objects of a particular type relative
     * to another object.
     *
     * The retrieval mechanism follows the paradigm established in TMF 814
     * and is described below:
     *
     * objectPath:    The name-value pair list of objects you want to list relative to another
object.
     *
     * Example:
     * ListAll Customer relative to objectPath
     *     Name="EMS",           Value="Juniper NMC-RX"
     *
     * ListAll Customer Site relative to objectPath
     *     Name="EMS",           Value="Juniper NMC-RX"
     *     Name="Customer",      Value="ABC Store Customer"
     *
     * objectType:    The type of objects that you want to list
     * Example:
     * UnspRXCustomerMgrObjectType.Customer
     * UnspRXCustomerMgrObjectType.CustSite
     *
     * howMany:       The maximum number of objects to report in the first batch in nameList
     *
     * nameList:      First batch of howMany objects report to NMS
     *
     * nameIt:        Iterator to retrieve the remaining number of objects after the first batch
     * In order to allow the NMS to deal more easily with retrievals, which may return a
     * very large amount of data, iterators are used. They provide a mechanism to retrieve
     * data batch by batch - the size of a batch is specified by the NMS via the
     * how_many parameter in the next_n method of the iterator reference.
     * For a more detailed explanation of the use of iterators, please refer to
     * TMF814\supportingDocumentation\iterators.html
     *
     * Raises: globaldefs::ProcessingFailureException
     * EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
     * EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
     * EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
     * EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
     * can support has been reached.
     *
     * EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
     */
    void listAll(
        in globaldefs::NVSLIST_T          objectPath,
        in UnspRXCustomerMgrObjectType    objectType,
        in long                            howMany,
        out globaldefs::NamingAttributesList_T    nameList,
        out globaldefs::NamingAttributesIterator_I    nameIt)
        raises(globaldefs::ProcessingFailureException);

    /**
     * This method should be used to view objects
     *
     * objectPath:    The name-value pair list of the object you want to configure.
     * Example:
     * View Customer Site relative to objectPath
     *     Name="EMS",           Value="Juniper NMC-RX"
     *     Name="Customer",      Value="ABC Store Customer"
     *     Name="CustSite",      Value="Westford Customer Site"
     *
     * attributesList: It will hold all attributes as a name-value pair list
     * to pass back to NMS to view this object.

```

```

* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*   EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*   EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*   EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                   can support has been reached.
*   EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void view(
    in globaldefs::NVSLIST_T      objectPath,
    out globaldefs::NVSLIST_T     attributesList)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get a list of object types on which you can perform list
* actions relative to other objects.
*
* objectPath:    The name-value pair list of the object you want to view.
* Example:
*   ListAll AtmSubInterfaces relative to objectPath
*       Name="EMS",           Value="Juniper NMC-RX"
*
* objectTypeList: To hold a list of object types
*
* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*   EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*   EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*   EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                   can support has been reached.
*   EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getListableObjectTypeSelections(
    in globaldefs::NVSLIST_T      objectPath,
    out UnsprXCustomerMgrObjectTypeList objectTypeList)
    raises(globaldefs::ProcessingFailureException);

/**
* This method should be used to get a list of types of action that you can perform
* relative to other objects.
*
* objectPath:    The name-value pair list of the object you want to view.
* Example:
*   ListAll AtmSubInterfaces relative to objectPath
*       Name="EMS",           Value="Juniper NMC-RX"
*
* objectTypeList: To hold a list of action types
*
* Raises: globaldefs::ProcessingFailureException
*   EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure
*   EXCPT_INVALID_INPUT - Raised when the invalid input is passed into this method
*   EXCPT_ENTITY_NOT_FOUND - Raised when objectPath does not reference an existing object
*   EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when maximum number of iterators that the EMS
*                                   can support has been reached.
*   EXCPT_NE_COMM_LOSS - Raised when communication to managedElement is lost
**/
void getSupportedActions(
    in globaldefs::NVSLIST_T      objectPath,
    out UnsprXCustomerActionTypeList actionTypesList)
    raises(globaldefs::ProcessingFailureException);
};
};
#endif

```

Provisioning Objects

The following sections describe the provisioning objects for this IDL.

Customer

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = Customer; Value = ACME Inc
- Associated MIBs
 - There are no associated MIBs

Table 86: Attributes (Customer)

Attribute	Characteristic	Description	MIB Entry
CUSTOMER_NAME	View	Name of the customer	None
CUSTOMER_IDENTITY	View	Unique identifier for a particular customer	None
CUSTOMER_ACCOUNT_ID	View	Account ID for the customer	None

Customer Site

- Sample Object Path
 - Name = EMS; Value = JUNIPER_NMCRX
 - Name = Customer; Value = ACME Inc
 - Name = CustomerSite; Value = Anytown
- Associated MIBs
 - There are no associated MIBs

Table 87: Attributes (Customer Site)

Attribute	Characteristic	Description	MIB Entry
CUSTOMER_SITE_NAME	View	Name of the customer site	None
CUSTOMER_SITE_IDENTITY	View	Unique identifier to represent this site for a particular customer	None
CUSTOMER_SITE_ACCOUNT_ID	View	Account ID for the customer site	None