SatLabs System Recommendations – Management & Control Planes Specifications

June 2010

info@satlabs.org | http://www.satlabs.org

Document history

Revision	Date	Changes
1.0	01-09-2008	First version. This document replaces the previous SSR v2 – Part 3: M&C, defined for Version 2.0. It is now an independent document, referenced by v2.1 SSR. Compared to this previous document, the only changes concern the following: - Minor editing - Clarification added on access modes and few MIB objects upgraded to Mandatory and RW, addition of DHCP as recommended on the LAN interface - Alignment with DVB-RCS MIB, revision “200807291200Z” - AVBDC repetition time is deprecated - ENHCLASSIFIER, INSTALL_LOG, FWDLINKSTATUS and RCST_PARA are changed from SatLabs options into technical features that can optionally be tested - fwdStartPopId removed from INDEX list, fwdStatusRowStatus object removed - some other indexes corrected according to recommendations in RFC2578: - phbIdentifier (shall be Not Accessible, and value starting from 1), - pidIndex (shall be Not Accessible) - fwdStatusIfReference removed as an INDEX - Correction of few inconsistencies with abbreviated terms ('Addr' or 'Address', 'IP' or 'Ip') in rcstNetwork and pktClass table. - RC instead of RW in requestClassTable and pidPoolTable - Added reference to SatLabs defined DVB-RCS MIB (Internet draft)
1.1	21-06-2010	Alignment to SatLabs DVB-RCS MIB as defined in RFC 5728. Addition of the specifications for the configuration file format and schema. Update of references: DVB-RCS specifications v1.5.1 and guidelines v1.4.1.

Editor: Stephane Combes, ESA/ESTEC (stephane.combes@esa.int,)

Inquiries related to the SatLabs Group should be directed to:

Xavier Lobao Pujolar - Chairman SatLabs Group

ESTEC P.O. Box 299
Noordwijk 2200 AG
Netherlands

Email: info@satlabs.org
Phone: +31 (0) 7156 53641
Fax: +31 (0) 7156 54093

Web site: www.satlabs.org

Table of Contents

List of Figures

List of Tables

1 References

[1] ETSI EN 301 790 v.1.5.1, “Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems” (2009-05)

[2] ETSI TR 101 790 v1.4.1, “Digital Video Broadcasting (DVB); Interaction channel for Satellite Distribution Systems; Guidelines for the use of EN 301 790”, (2009-07)

[3] ETSI EN 300 468 v1.9.1, "Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems" (2009-03)

[4] ETSI TS 102 006 v1.3.2, “Specification for system SW update in DVB systems” (2008-07)

[5] ETSI EN 302 307 v1.1.2, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications" (2006-06)

[6] SatLabs System Recommendations (Version 2.1)

[7] SatLabs System Recommendations - QoS Specifications, (Version 1.1)

[8] IETF STD 9 – RFC 959, File Transfer Protocol (1985-10)

[9] IETF STD 13 – RFC 1034, Domain names – concepts and facilities (1987-11)

[10] IETF STD 13 – RFC 1035, Domain names - implementation and specification (1987-11)

[11] IETF STD 17 - RFC1213, Management Information Base for Network Management of TCP/IP-based internets: MIB-II (March 1991-03)

[12] IETF STD 33 – RFC 1350, The TFTP Protocol (Revision 2) (1992 -07)

[13] IETF RFC 2131, Dynamic Host Configuration Protocol (1997-03)

[14] IETF RFC 2347, TFTP option format

[15] IETF RFC 2348, TFTP Blocksize Option (1998-05)

[16] IETF RFC 2349, TFTP Timeout Interval and Transfer Size Options (1998-05)

[17] IETF BCP 23 – RFC 2365, Administratively Scoped IP Multicast (1998-07)

[18] IETF STD 58 - RFC2578, Structure of Management Information, Version 2 (SMIv2) (1999-04)

[19] IETF STD 58 – RFC 2579, Textual Conventions for SMIv2 (1999-04)

[20] IETF STD 58 - RFC 2580, Conformance Statements for SMIv2 (1999-04)

[21] IETF RFC 2669, DOCSIS Cable Device MIB Cable Device Management Information Base for DOCSIS compliant Cable Modems and Cable Modem Termination Systems (1999-08)

[22] IETF RFC 2670, Radio Frequency (RF) Interface Management Information Base for MCNS/DOCSIS compliant RF interfaces (1999-08)

[23] IETF RFC 2863, The Interfaces Group MIB (2000-06)

[24] IETF RFC 3140, Per Hop Behavior Identification Codes (2001-06)

[25] IETF RFC 3289, Management Information Base for the Differentiated Services Architecture (2002-05)

[26] IETF RFC 3411, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks (2002-12)

[27] IETF STD 62 - RFC 3416, Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP) (2002-12)

[28] IETF STD 62 – RFC 3418, Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)

[29] IETF RFC 4001, Textual Conventions for Internet Network Addresses (2005-02)

[30] IETF RFC 5017, MIB Textual Conventions for Uniform Resource Identifiers (URIs) (2007-09)

[31] http://www.iana.org/assignments/ianaiftype-mib

[32] http://www.iana.org/assignments/smi-numbers

[33] IETF RFC5728, The SatLabs Group DVB-RCS MIB (2010-03)

[34] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization. International Standard 8824 (1987-12)

2 Acronyms

8PSK	8-ary Phase Shift Keying
16APSK	16-ary Phase Shift Keying
32APSK	32-ary Phase Shift Keying
ACM	Adaptive Coding and Modulation
ATM	Asynchronous Transfer Mode
AVBDC	Absolute VBDC
BPSK	Binary Phase Shift Keying
BER	Bit Error Rate
BSM	Broadband Satellite Multimedia – Ad Hoc working group in ETSI under the SES technical body
BTP	Burst Time Plan
BUC	Block Up-Converter
CCM	Constant Coding and Modulation
CNR	Carrier to Noise Ratio
CoS	Class of Service
CRA	Continuous Rate Assignment
CSC	Common Signaling Channel
CW	Continuous Wave
dBi	deciBel (isotropic)
dBm	Decibel (with respect to 1 mW)
DC	Direct Current
DHCP	Dynamic Host Control Protocol
DL	Downlink
DNS	Domain Name Server
DSCP	Diffserv Code Point
DTD	Document Type Definition
DVB	Digital Video Broadcast
ETSI	European Telecommunications Standards Institute
FCAPS	Fault, Configuration, Accounting, Performance, Security
FEC	Forward Error Correction
FL	Forward Link
FTP	File Transfer Protocol

GSE	Generic Stream Encapsulation
HM&C	Harmonized Management & Control
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
ID	Identifier
IDU	Indoor Unit
IGMP	Internet Group Management Protocol
IP	Internet Protocol
LAN	Local Area Network
LNB	Low Noise Block
LO	Local Oscillator
MAC	Medium Access Control
MIB	Management Information Base
MF-TDMA	Multi-Frequency TDMA
MMT	Multicast Mapping Table
MODCOD	Modulation/Coding
MPAF	MPEG Adaptation Field
MPE	Multi-Protocol Encapsulation
MPEG	Motion Picture Expert Group
MSDP	Multicast Software Distribution Protocol
MTU	Maximum Transfer Unit
NCC	Network Control Center
NIT	Network Information Table
NLID	Network Layer Information Descriptor
NMEA	National Marine Electronics Association
OAM	Operation, Administration and Maintenance
OBR	Out of Band Request
ODU	OutDoor Unit
OID	Object Identifier
PHB	Per Hop Behavior
PID	Packet Identifier
PMT	Program Map Table
QoS	Quality of Service
QPSK	Quadrature Pahse Shift Keying
RBDC	Rate Based Dynamic Capacity

RC	Request Class
RCS	Return Channel - Satellite
RCST	Return Channel Satellite Terminal
RL	Return Link
Rx	Reception
SES	Satellite Earth Stations, ETSI technical body
SLA	Service Level Agreement
SMI	Structure of Management Information
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SSPA	Solid State Power Amplifier
SW	Software
SWU	Software Upgrade
TBTP	Terminal Burst Time Plan
TDM	Time Division Multiplex
TDMA	Time Division Multiple Access
TFTP	Trivial File Transfer Protocol
TID	Transfer Identifier
TIM	Terminal Information Message
TIM-U	TIM Unicast
TS	Transport Stream
Tx	Transmission
UDP	User Datagram Protocol
UL	Uplink
VBDC	Volume Based Dynamic Capacity
VCI	Virtual Channel Identifier
VLAN	Virtual LAN
VPI	Virtual Path Identifier
XML	Extensible Mark-up Language

3 Introduction

This document forms the Management and Control (M&C) specifications part of the SatLabs System Recommendations, starting from Version 2.0 (see [6]).

It defines Management and Control architecture and specifications for DVB-RCS systems [1].

These specifications focus on terminal management through the air interface, based on the current DVB-RCS specifications. It is based on known and established standards (e.g. SNMP).

These specifications aim at enabling harmonised network support installation, commissioning, management and control of a RCST. In particular, they enable a controlled way of introducing a new SatLabs compatible RCST into the network so that the RCST is allowed into the network when its antenna is sufficiently aligned, the Software/Firmware has a sufficiently recent revision and it is sufficiently configured.

4 FCAPS

The RCST lifecycle poses different Management needs for different phases. This can be divided as per the FCAPS model:

1. FCAPS-Fault
2. FCAPS-Configuration
3. FCAPS-Accounting
4. FCAPS-Performance
5. FCAPS-Security

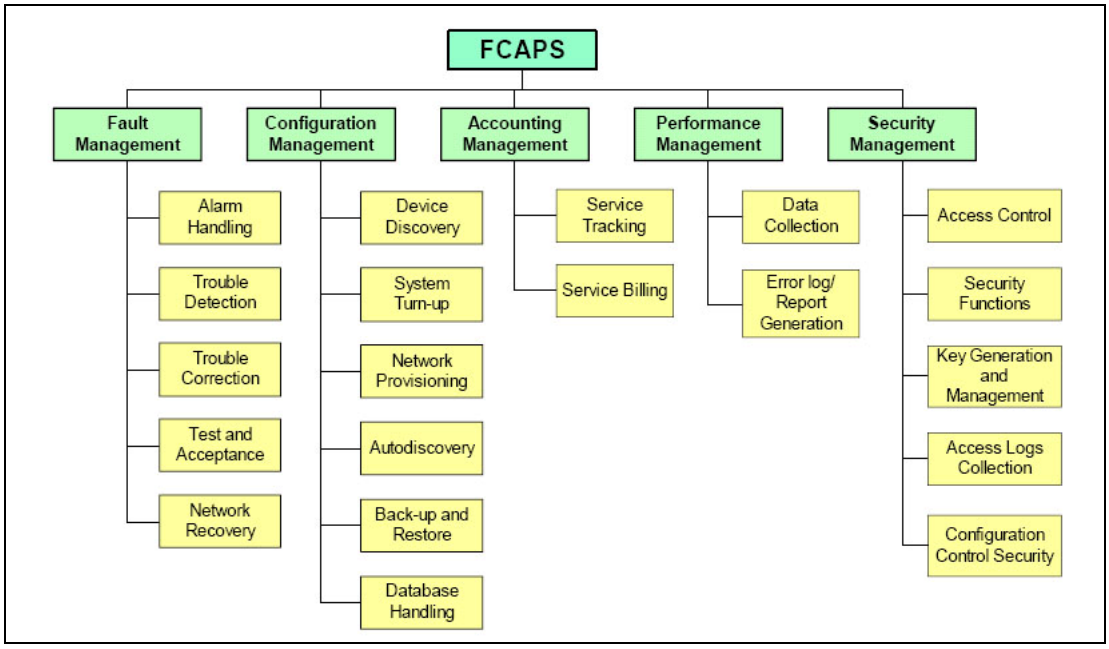

Figure 4-1 - FCAPS

The present recommendations mostly focus on Fault, Configuration and Performance.

5 Basic Capabilities

This section lists the protocols that must be supported by RCSTs or that can be supported as an option.

5.1 SNMP

The protocol support requirements are as follows:

1. All RCSTs will support SNMP/UDP/IP in their Satellite and Ethernet interfaces. The RCST will be an SNMP agent/client.
2. The RCSTs will support SNMPv2c ([27] and [28]).

The required MIB modules are the following:

1. MIB-II

Only the *interface* (if) and *system* (sys) subgroups of MIB-II [11] are mandatory for the Satellite and Ethernet Interfaces. The other subgroups might be implemented (vendor specific).

2. DVB-RCS MIB

The MIB specified in older version of the DVB-RCS guidelines [2] is considered obsolete. The DVB-RCS MIB mandated by SatLabs is defined IETF document [33] and recalled in Appendix G – MIB.

All the MIB parameters listed in the following sections refer to the DVB-RCS MIB, unless specified otherwise.

The RCST management IP address and prefix length are respectively given by the MIB parameters *dvbRcsNetworkOamInetAddress* and *dvbRcsNetworkOamInetAddressPrefixLength* (see 17.2.3.1.2).
If the RCST supports the EXTNETWORK feature, the *dvbRcsNetworkOamInetAddressAssign* MIB parameter (see 17.2.3.1.2) indicates whether the OAM address is statically or dynamically assigned.
If the RCST supports the EXTNETWORK feature, the IP address of the management server in the NCC is given by the MIB parameter *dvbRcsNetworkNccMgtInetAddress* (see 17.2.3.1.2).

5.2 File Transfer

All RCSTs will have TFTP client [12] capabilities for their satellite interface.
FTP OPTION: a RCST can also implement FTP client [8].
TFTP and/or FTP are used for Configuration file download (see 7.2).
All RCST also support MSDP for Software upgrade (see 12 – Appendix B – SWU).

5.3 ICMP

All RCSTs will answer through the satellite return link to ping messages that are sent to their management IP address via satellite forward link.

The IP address of the default gateway on the Air Interface is given by the MIB parameter *dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress* (see 17.2.3.1.2).

5.4 DNS

DNS OPTION: a RCST can implement a DNS client (see [9] and [10]).

If DNS is supported, the IP address of the DNS server is given by the *dvbRcsPrimaryDnsServerInetAddress* and *dvbRcsSecondaryDnsServerInetAddress* MIB parameters (see 17.2.3.1.2).

If DNS is supported, hostname can replace IP addresses in some configuration parameters (see 17.2.3.1.2). Such DNS is only used for RCST management operation (e.g. to resolve TFTP or FTP host names). The DNS server is located in the NCC.

5.5 DHCP

A RCST should implement DHCP on the LAN side.

6 FCAPS – Fault

6.1 Alarm Handling and Trouble Detection

1. **Definition:** The RCST enables the NCC to detect troubles and handle the alarms. RCST alarms should be sent to an internal RCST log.
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the collection of information by the NCC. This allows the NCC to understand what might be the cause for trouble and try to solve the problem.
4. **Protocol Spec:** SNMP GET
5. **Basic functions:** n/a
6. **Basic parameters:**
   6.1. *dvbRcsRcstState* Group MIB (see section 17.2.3.1.6)
   6.2. From MIB-II, use the error counters from the *interface* (if) group (see section 17.2.4.3.2)

6.2 Trouble Correction

1. **Definition:** The RCST enables the NCC to correct troubles.
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the correction of RCST behavior from NCC.
4. **Protocol Spec:** SNMP SET, TIM (TIM can be used for transmission disable as specified in [1]).
5. **Basic functions:**
   5.1. Reboot.
   5.2. Transmission disable: done through TIM, see [1], or as an option, through SNMP (see 17.2.3.1.5)
   5.3. User transmission disable: used to disable user traffic (only RCST management traffic can be transmitted)
6. **Basic parameters:**
   6.1. *dvbRcsCtrlRebootCommand* (see section17.2.3.1.5)
   Forces the RCST to reboot:
   (1) - idle
   (2) - normal reboot (from current software load)
   (3) - reboot from alternate load (swap to alternate load before reboot)

   6.2. *dvbRcsCtrlRcstTxDisable* (for SNMPMISC option only: see section 17.2.3.1.5)
   Forces the RCST to stop transmission (as Transmit_Disabled defined in [1]).

   6.3. *dvbRcsCtrlUserTrafficDisable* (see section 17.2.3.1.5)
   Disables user traffic (only RCST management traffic can be transmitted).

   6.4. *dvbRcsCtrlRcstLogonCommand* (for SNMPMISC option only: see section 17.2.3.1.5)

Initiates a RCST logon.

6.5. *dvbRcsCtrlRcstLogoffCommand* (for SNMPMISC option only: see section 17.2.3.1.5)

Initiates a RCST logoff.

6.3 Test and Acceptance

1. **Definition:** The RCST enables the NCC to test its functionality and perform acceptance measurements.
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the NCC to get measurements of information that is needed for acceptance tests and initiate tests (such as CW).
4. **Protocol Spec:** SNMP GET and SET, TIM
5. **Basic functions:**
   - 5.1. Forward link CNR measurements
   - 5.2. Forward link BER measurements.
   - 5.3. CW On/Off
   - 5.4. RCST reboot
   - 5.5. Tx Disable, Traffic Disable, Wake-Up and Log-on and off at the MAC Layer as mandated by [1].
6. **Basic parameters:**

   6.1 dvbRcsFfwdStatusCnr (see section 0)

   Provides the RCST CNR on the Forward Link.

   6.2 dvbRcsFwdStatusBer (see section0)

   Provides the RCST BER on the Forward Link.

   6.3 *dvbRcsCtrlCwEnable* (see section 17.2.3.1.5)

   Forces the RCST to start CW transmission.

   6.4 *dvbRcsCtrlRebootCommand* (see section 17.2.3.1.5)

   6.5 *dvbRcsCtrlRcstTxDisable* (for EXTCONTROL feature only: see section 17.2.3.1.5)

   6.6 *dvbRcsCtrlRcstUserTrafficDisable* (see section 17.2.3.1.5)

   6.7 *dvbRcsCtrlRcstLogonCommand* (for EXTCONTROL feature only: see section 17.2.3.1.5)

   Initiates a RCST logon.

   6.8 *dvbRcsCtrlRcstLogoffCommand* (for EXTCONTROL feature only: see section 17.2.3.1.5)

   Initiates a RCST logoff, as Log_off defined in [1] (see 13 - Appendix C – Graceful Logoff Process from NCC).

7 FCAPS – Configuration

7.1 System Turn-up and Service Provisioning

1. **Definition:** The RCST enables the NCC to validate its configuration and allow it to go on-line.
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the NCC to check its Software image and trigger a Software download if needed.
4. **Protocol Spec:** SNMP GET and SET
5. **Basic functions:**

When the RCST boot, it will first verify that its SW image is adequate and only then will the RCST go online (see 12 – Software upgrade).

The transition between SW download and going online is vendor specific (some vendors might choose to go through re-boot to achieve this).

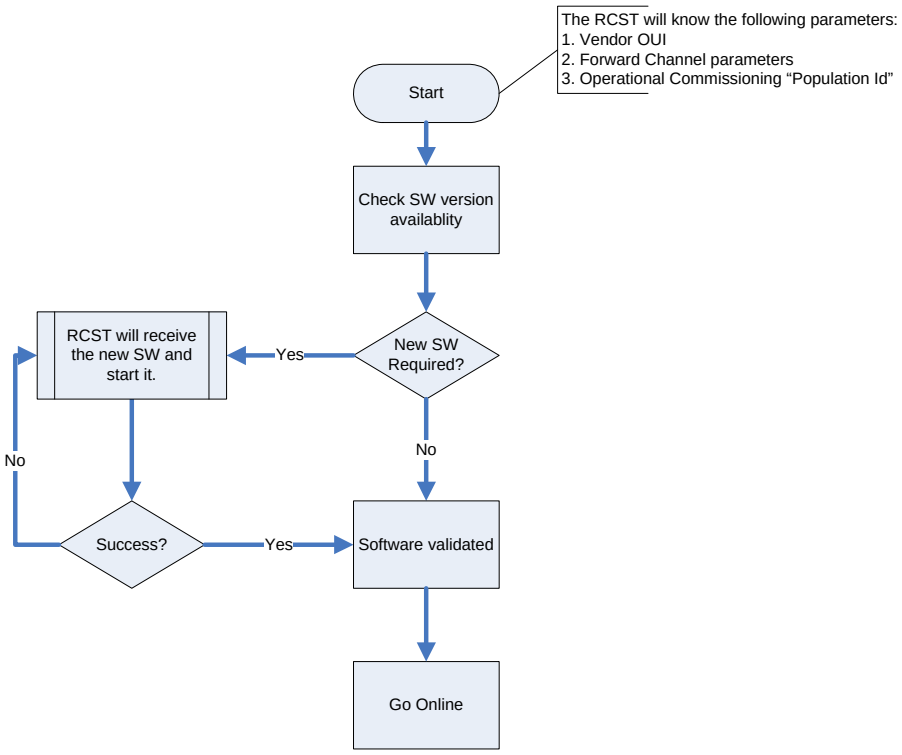

Figure 7-1 – System turn-up

6. **Basic parameters:**

   6.1 *dvbRcsRcstCurrentSoftwareVersion* (see 17.2.3.1.6)

   Is the current software version running on the RCST. It is also provided through the CSC burst (see [1]).

6.2 *dvbRcsRcstAlternateSoftwareVersion* (see 17.2.3.1.6)

Is the alternate software version downloaded on the RCST.

6.3 *dvbRcsCtrlRebootCommand* (see section 17.2.3.1.5)

Note: as the software version field of the CSC burst is only 8-bit long, it does not represent the exact software version. The value assigned in the CSC for a given software load should be vendor specific. Vendors will need to publish a mapping table that relates the value reported in the CSC burst to the actual software version. The NCC should use the MAC address (in the CSC) to discriminate between vendors that may report the same value in the CSC burst.

7.2 Installation and commissioning

1. **Definition**: Installation and commissioning of the RCST
2. **Applicable**: Yes
3. **Function Specification**: The RCST allows the NCC to detect it is in Installation mode, drive the RCST antenna alignment. Mandatory commissioning parameters are configured. As an optional feature (INSTALL_LOG feature), the RCST should allow the NCC to retrieve an installation log file.
4. **Protocol Spec**: SNMP GET and SET
5. **Basic functions**:

In order for the RCST to start working at the MAC level, the installer can use the LAN interface to configure some initial parameters (forward link frequency etc.) through SNMP. Supported parameters through this interface are listed in 16 - Appendix F – Mandatory commissioning and installation parameters supported via LAN interface.

Section 14describes the mandatory commissioning parameters.

Section 15 describes the mandatory installation parameters.

If the NLID/TIM-U option is supported by the RCST, the RCST management IP address can be configured this way.

The solution should minimize Return-Channel satellite traffic before Antenna alignment is achieved since until this phase is over, the risk for the RCST to interfere with other networks is high.

The RCST will have a “Population Id” parameter that could take different (it could be the same) values depending on the RCST operating mode: Installation and Operational. The "Population Id” will have a default value (Installation Population ID) that all RCSTs from all vendors will have when they leave the factory. This value will be "0". Nevertheless, the RCST shall give the Installer the option to change this value if needed. The value for the Operational mode (Operational Population ID) replacing the Installation Population ID will be given to the RCST as described in the following procedure.

1. When in Installation Mode, the RCST will:
    a. Use "Installation Population Id" as the RCST "Population Id" and Network ID
    b. Set the “RCST mode" field in the "RCST capability" structure to "Installation Mode" (see CSC burst description in [1] – section 6.2.3).

    Every RCST MUST allow the installer to set the above parameters.
2. The RCST will perform MAC logon to this channel.
3. The Hub will detect the fact that the RCST is in "Installation Mode" and will not assign the RCST any SYNC slots. The RCST will not fail due to that and will continue work (without any transmissions) but the TIM has to be received.
4. The NCC will instruct the RCST to enter CW mode and will send it relevant parameters for the alignment process every time interval.
    a. These parameters are SNMP objects
    b. This SNMP message will be sent to the terminal via SNMP/UDP/IP.
    c. The parameters are described in 14 - Appendix D – Mandatory Commissioning Parameters.
5. The RCST will use this information to assist in the antenna alignment process.
6. The RCST will transmit CW
7. The RCST will wait in this condition until the NCC informs it of the process result (Antenna Alignment State is SET to Success or Fail) or "CW Max Duration" period is over.
8. Once alignment is achieved, the RCST should:
    a. Set the “RCST mode" field in the "RCST capability" structure of the CSC burst (see [1]) to "Operational Mode".
    b. Load its "Operational Population Id" to the "Population Id".
    c. Start the Operational state machine.

When INSTALL_LOG feature is supported, an installation log file can be created on the installer’s computer and downloaded to the RCST. This log file can then be retrieved from the RCST by the NCC or by the installer via the LAN.

6. **Basic parameters:**

See 14 - Appendix D – Mandatory Commissioning Parameters,

15 - Appendix E - Mandatory Installation Parameters,

and 16 - Appendix F – Mandatory commissioning and installation parameters supported via LAN interface.

6.1 *dvbRcsRcstMode* (see 17.2.3.1.6)

Tells whether the RCST is in installation or operational mode. The same object also allows, through SET, to switch a RCST back into installation mode.

6.2 *dvbRcsRcstSystem* subgroup (see 17.2.3.1.1)

6.3 *dvbRcsRcstInstall* subgroup (see 17.2.3.1.3)

6.4 *dvbRcsFwdConfig* subgroup (see 17.2.3.2.1)

6.5 *dvbRcsFwdStatus* subgroup (see 0)

6.6 *dvbRcsRtnConfig* subgroup (see 17.2.3.3.1)

6.7 *dvbRcsRtnStatus* subgroup (see 17.2.3.3.2)

6.8 *dvbRcsRcstControl* subgroup (see section 17.2.3.1.5)
in particular *dvbRcsCtrlCwEnable:* Forces the RCST to start CW transmission.

6.9 *dvbRcsNetworkInstallLogFileDownloadUrl* (for INSTALL_LOG feature only: see section 17.2.3.1.2)
Full path of the installation log file to download.

6.10 *dvbRcsNetworkInstallLogFileUploadUrl* (for INSTALL_LOG feature only: see section 17.2.3.1.2)
Full path of the installation log file.

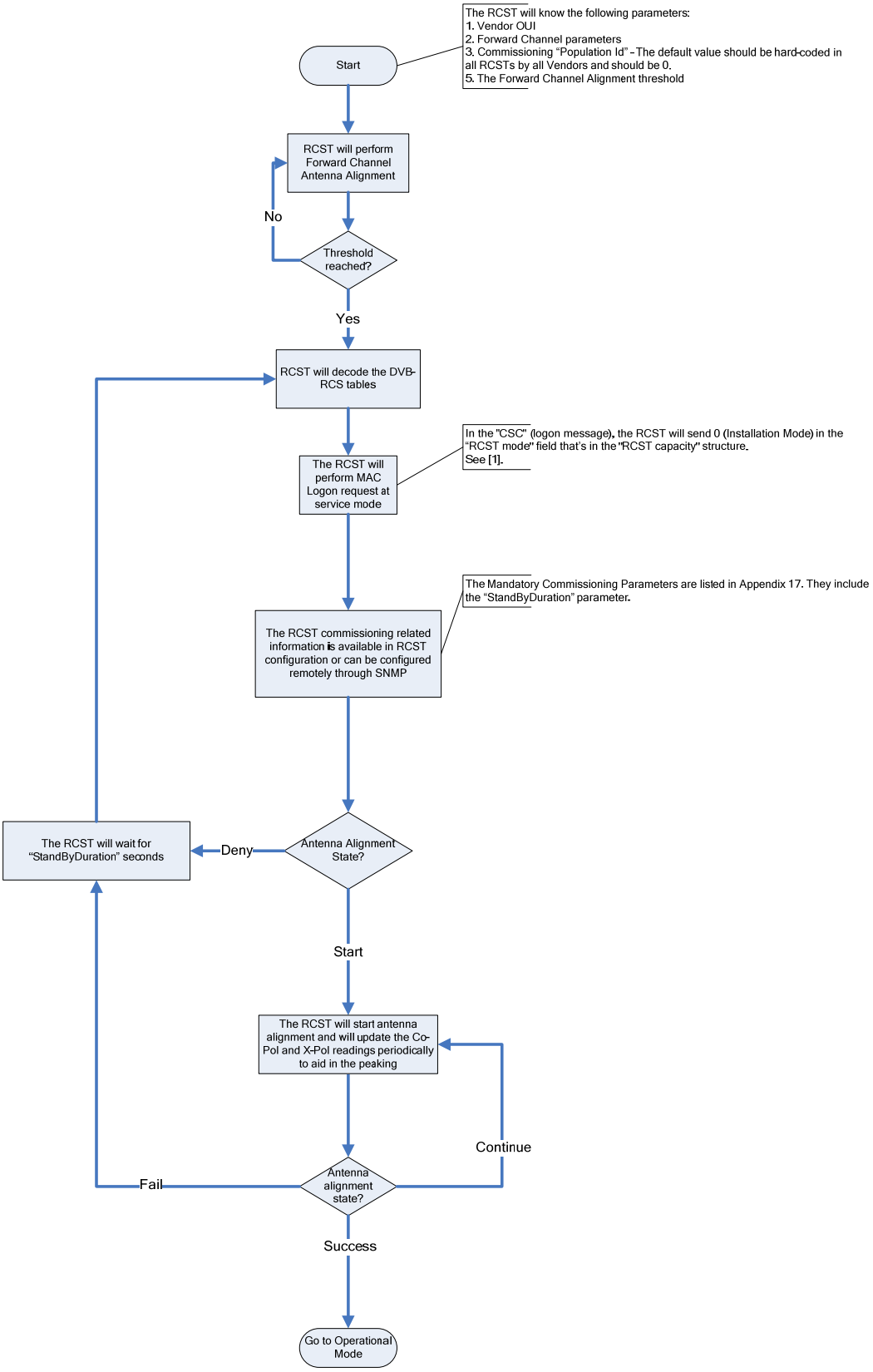

Figure 7-2 – RCST commissioning

7.3 Configuration delivery (Service provisioning)

1. **Definition**: RCST configuration
2. **Applicable**: Yes
3. **Function Specification**: The RCST allows the NCC to download a configuration file.
4. **Protocol Spec**: SNMP SET, TFTP (option FTP), TIM-unicast
5. **Basic functions**:

The need for configuration arises at two cases:

1. Installation time
2. Configuration Updates

The solution should allow the RCST to download the configuration file only when it is needed (configuration was changed).

TIM-unicast can be used to transfer the Return Interaction Path (RIP) descriptor [1]. RIP conveys the supported mapping of Request Classes (RC) to VPI/VCI (or PID pool). Please refer to SSR-Part 2: QoS Specifications [7] for the definition of PID pool. The Channel_ID field is used to indicate the RC associated with the VPI/VCI (or PID pool).

RCST SHALL support RIP descriptor [1]. The RIP parameters can be delivered via the configuration file as well.

TFTP (or FTP as an option) can be used to transfer the configuration file from the NCC (server) to the RCST (client).
The version of the configuration file is a Display String. The file version shall be contained in the file and its calculation is vendor specific.
The configuration file download will proceed as follows:

1. Configuration will start after the installation parameters were entered and after the RCST has performed MAC level logon.
2. The NCC will send SNMP SET *dvbRcsNetworkConfigFileDownloadUrl* (which includes the TFTP or FTP server IP address)
3. The NCC will send SNMP SET *dvbRcsCtrlDownloadFileCommand* for configuration file
4. The RCST will use TFTP or FTP and the above information to download the configuration file. Once downloaded, the RCST validates the configuration file and checks the file version. Upon a successful validation and check, *dvbRcsRcstDownloadedConfigFileVersion* is updated with the version of the file that was just downloaded.
5. The NCC will send SNMP SET *dvbRcsCtrlActivateConfigFileCommand* at the desired time of activation (immediately following the *dvbRcsCtrlDownloadFileCommand* or at a later time). In some RCST implementation, it may be required for the RCST to reboot in order to take into account the new

configuration file (vendor specific). Once the configuration file is activated, the *dvbRcsRcstActivatedConfigFileVersion* is updated with the file version.

Full configuration of common elements defined by HM&C has to be supported (not delta configuration with changes detection).

If a local change occurs at the RCST, it is superseded by the Configuration File received from NCC.

A configuration file is always sent in full for the common parameters.

No security requirement is associated to configuration file download.

6. Basic parameters:

Section 11 describes the mandatory parameters in the Configuration file.

The RC to Channel_ID mapping can be configured through RIP, the Configuration file or SNMP (Write option for parameters defined in 17.2.3.1.4). The latest received by an RCST always takes precedence and overwrites the configuration.

6.1 *dvbRcsRcstActivatedConfigFileVersion* (see section 17.2.3.1.6)

Version of the activated configuration file. Version is vendor specific.

6.2 dvbRcsRcstDownloadedConfigFileVersion (see section 17.2.3.1.6)

Version of the last downloaded configuration file. Version is vendor specific.

6.3 dvbRcsNetworkConfigFileDownloadUrl (see section 17.2.3.1.2)

Full path name for the configuration file download.

6.4 dvbRcsCtrlDownloadFileCommand (see section 17.2.3.1.5)

This variable shall initiate a RCST configuration file download process

(1) idle

(2) download RCST configuration file from TFTP/FTP server

6.5 dvbRcsCtrlActivateConfigFileCommand (see section 17.2.3.1.5)

Triggers the RCST to use the configuration file and update its parameters accordingly. Some RCST implementation may require a reboot for the parameters to take effect (vendor specific).

(1) idle

(2) activate

Note: at each reboot, the RCST update its parameters with the current configuration. After a new configuration file is downloaded, the RCST can update its parameters after receiving the *dvbRcsCtrlActivateConfigFileCommand* or at the next reboot, whichever comes first.

Configuration File format:

The syntax will be XML with the following requirements:

1. The file will be opened by any standard XML viewer/parser
2. Every parameter in the file will have a SNMP OID and this will be the unique identifier for the parameter
3. The minimal set of attributes are
   a. SNMP OID (it is important since it allows to make unique identification of objects in particular wrt private ones)
   b. Value
4. The following can be added/changed without any need for change in the minimal XML parser
   a. Additional information for a parameter; the additional information will be captured, but will not be processed. The parser will not crash
   b. Additional parameters; for additional parameters with unknown tags, the parser will capture the info and it will not be processed. The parser will not crash.
   c. Structured or flat XML

<**Element**>

```
<Parameter ID="1.3.6.1.4.1.7352.3.5.3.5.1.1.0">
        <Value>1</Value>
</Parameter>
<Parameter ID="1.3.6.1.4.1.7352.3.5.3.1.1.2.12">
        <Value>0</Value>
</Parameter>
```

</Element>

The first is a scalar parameter and the second is a table instance parameter.

Detailed configuration file format and schema can be found in 18-Appendix H – Configuration file format.

7.4 Software update

1. **Definition**: RCST software update
2. **Applicable**: Yes
3. **Function Specification**: The RCST allows the NCC to download software images and command RCST re-boot.
4. **Protocol Spec**: SNMP SET, MSDP (defined in section 12. Appendix B – SWU)
5. **Basic functions**:
   1. The Software Update (SWU) service will run over the forward link as a multicast IP service.

2. SWU will support multiple SW and control streams
3. SWU will allow the RCST to detect a change in its streamed SW version and to download this version.
4. Reboot command

The procedure makes use of SNMP and requires implementation of the RCST MIB. An RCST image will be distributed to the RCST in one binary file by the NCC. The RCST does not have to maintain one binary file for the image, but does have to provide the SW version upon request by the NCC. The SW version is a string and its structure is RCST vendor specific.

The RCST is required to maintain two images of the software. The images are referred to as the current and alternate images. The current image is the image the RCST is running. The alternate image is the other image and is not running. Therefore the RCST MIB includes two parameters for the associated SW versions (see 17.2.3.1.6):

- *dvbRcsRcstCurrentSoftwareVersion*
- *dvbRcsRcstAlternateSoftwareVersion*

The RCST software download process shall only update the alternate image.
The software image is received using MSDP as described in section 12. Appendix B – SWU.

It should be noted that the RCST will “take” the new software image if the following conditions are met:

- matches the identifier of the manufacturer (*manufID* parameter for MSDP in 12.7)
- the software version of the downloaded software is different from both the current and alternate software version in the RCST

A new image has to pass the download validation stage. The download validation is done after the file transfer of the new image. The way this validation is performed is vendor specific. It can fail for instance if the checksum is wrong or if the version is too old and cannot be supported by the RCST. Potential step back to a previous version by the RCST is also vendor specific. The software download status is declared a ‘success’ if this validation phase is successful.

After the successful download validation, the RCST may auto-switch to the alternate image (the new one), depending on vendor specific configuration. Such auto-switch will trigger a RCST log-off then log-on.
If no vendor specific configuration allows auto-switch, it will be necessary for the IDU to reboot for the new image to take effect. After the auto-switch or the reboot, the RCST is running the latest software as the current image, and its alternate image is the previous version of software.

In the case the validation had failed, the RCST will re-start receiving the software download stream.
The RCST software download also provides the mechanism to revert to the previous version of software.
This would involve the use of the *dvbRcsCtrlRebootCommand* from alternate load (see 17.2.3.1.5).

6. **Basic parameters:**

   6.1 dvbRcsRcstCurrentSoftwareVersion (see section 17.2.3.1.6)

   Current software version of the Software Download image.

   6.2 dvbRcsRcstAlternateSoftwareVersion (see section 17.2.3.1.6)

   Alternate software version of the Software Download image.

7.5 Device Discovery

Device discovery is not applicable in the short term.

The method in which the NCC will detect the fact that an RCST is available in the management plane is NCC related and not RCST and in any case should be handled in the long-term solution.

The NCC has various ways of achieving this, among which: using periodic pings, using SNMP GET, NCC MAC logon detection capabilities.

7.6 Upload Configuration File

1. **Definition**: Configuration file upload from the RCST to the NCC
2. **Applicable**: Yes
3. **Function Specification**: The RCST allows the NCC to command configuration file upload
4. **Protocol Spec**: SNMP SET, TFTP (option FTP)
5. **Basic functions**:

The RCST will support the upload of its configuration to the NCC in the form of a configuration file.

1. The NCC will send SNMP SET *dvbRcsNetworkConfigFileUploadUrl*
2. The NCC will send SNMP SET *dvbRcsCtrlUploadFileCommand* for configuration file
3. The terminal starts the configuration file upload immediately with TFTP (or FTP as an option).

The uploaded configuration file has to reflect the current RCST configuration. This means that any parameter change (through SNMP for instance) has to be reflected in the uploaded file.

The RCST has to update the version parameter in the configuration file if there has been a change in the RCST configuration since the previous version. The format of this version parameter is vendor specific.

6. **Basic parameters:**

   6.1 dvbRcsNetworkConfigFileUploadUrl (see section 17.2.3.1.2)

   Full path name for the configuration file upload.

   6.2 dvbRcsCtrlUploadFileCommand (see section 17.2.3.1.5)

   This variable shall initiate a RCST upload process

(1) idle

(2) upload RCST configuration file to TFTP/FTP server

(3) upload RCST event/alarm log file to TFTP/FTP server

(4) upload RCST installation log file to TFTP/FTP server (for INSTALL_LOG feature only)

8 FCAPS – Accounting

8.1 Service tracking

N/A

8.2 Service Billing

N/A

9 FCAPS – Performance

9.1 Data Collection

1. **Definition:** Collect data from RCST interfaces
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the NCC to collect data regarding the traffic on each RCST interface
4. **Protocol Spec:** SNMP GET
5. **Basic functions:**

Objects from the *interface* (if) and *ifMIBObjects* subgroups of mib-2 (32-bit format) shall be used (see sections 17.2.4.3.2 and 17.2.4.3.3):

- *ifInOctects*
- *ifOutOctects*
- *ifInUcastPkts*
- *ifOutUcastPkts*
- *ifInErrors*
- *ifOutErrors*
- *ifInDiscards*
- *ifOutDiscards*
- *ifInMulticastPkts*
- *ifOutMulticastPkts*
- *ifInBroadcastPkts*
- *ifOutBroadcastPkts*

9.2 Error/Log report generation

1. **Definition:** Generate error/log report
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the NCC to trigger the upload of an event log file and view the error counters from mib-2.
4. **Protocol Spec:** SNMP GET and SET, TFTP (option FTP)
5. **Basic functions:**

Error counters from the objects in the *interface* subgroup (if) of mib-2, as defined in section 9.1, shall be used.

SNMP SET is used to set some parameters to get an error log from the terminal to a TFTP Server.

- The NCC will send SNMP SET *dvbRcsNetworkLogFileUploadUrl*
- The NCC will send SNMP SET *dvbRcsCtrlUploadFileCommand* for event log file
- The terminal starts the event log file upload immediately with TFTP (or FTP as an option).

6. Basic parameters:

6.1 *dvbRcsRcstLogUpdated* (see section 17.2.3.1.6)

Indicates the existence of an updated event log file.

6.2 *dvbRcsNetworkLogFileUploadUrl* (see section 17.2.3.1.2)

Full path of the event log file.

6.3 *dvbRcsCtrlUploadFileCommand* (see section 17.2.3.1.5)

This variable shall initiate a RCST upload process:

(3) upload RCST event/alarm log file to TFTP/FTP server

9.3 Trending

N/A

10 FCAPS – Security

10.1 RCST access control

1. **Definition:** Access control at the RCST
2. **Applicable:** Yes
3. **Function Specification:** The RCST allows the NCC to trigger a MAC level logoff.
4. **Protocol Spec:** SNMP SET
5. **Basic functions:**

MAC level log-off

Only MAC level (as already defined by the DVB-RCS standard) is mandatory for access control.

The NCC can use SNMP to activate MAC level logoff, as an option.

Section 13 shows this process.

6. Basic parameters:

dvbRcsCtrlRcstLogoffCommand (see section 17.2.3.1.5)

Initiates a RCST logoff.

10.2 Collect RCST access information

N/A

11 Appendix A – Unified set of configuration parameters

List of configuration parameters that each vendor configuration file MUST have:

1. Configuration file identifier (string type)
2. Ethernet IP address (parameter *dvbRcsNetworkLanInetAddress* in section 17.2.3.1.2)
3. Ethernet IP prefix length (parameter *dvbRcsNetworkLanInetAddressPrefixLength* in section 17.2.3.1.2)
4. SNMP write community string (char string)
5. SNMP read community string (char string)
6. SLA/QoS parameters (see section 17.2.3.1.4)
7. Operational Population ID (parameter *dvbRcsFwdStartPopId* in section 17.2.3.2.1)

The support of all these parameters through SNMP GET is also mandatory except for the SNMP community strings.

12 Appendix B – SWU

12.1 Introduction

The present chapter defines a solution allowing interoperability for *system software update* services and receivers. It is denoted MSDP (Multicast Software Download Protocol). It has been selected to minimize interdependencies between the parties involved. In particular:

- It defines how to locate the stream containing the *system software update service* in a network.
- It defines the signaling information used to locate the *system software update service* in a transport stream (via the PMT or MMT and the information part of the service).
- It defines the transmission of the actual *system software update service* as a standardized IP multicast.
- The protocol is based on OUIs (Organization Unique Identifier) for identifying manufacturer.
- It defines components that can be used to enhance the system software update functionality in an upward compatible way. This provides a standard mechanism for carrying additional information, e.g. update scheduling information, extensive selection and targeting information, action notification, filtering descriptors.

This solution offers all the required functionality as compared with [4], but eliminates the need for costly servers and complex software implementations in the RCST which are associated with the previously proposed techniques. The present solution is standards-based.

12.2 Scope

DVB-RCS terminal software is complex. In order to guarantee the functionality of a terminal as well as increasing its functionality once deployed in the field a software update service is required. The present chapter specifies a mechanism for signaling a software update service and the means to carry the data for such a software update service.

The mechanism takes advantage of the IP capabilities present in a DVB-RCS terminal to keep the lower layer implementation simple and unchanged from the current version of the DVB-RCS specification [1]. It also takes advantage of the multicast capabilities of DVB-S and DVB-S2.

The present chapter does not define the mandatory character of this protocol in a specific context. For example, it does not specify the triggers for software updates or the distribution schedule. Furthermore, it does not preclude the use of proprietary mechanisms for doing a software update. This allows a network to support horizontal market model DVB-RCS terminals. Equally it allows terminals requiring a software update service to be deployed in a network independent way.

12.3 Overview of the Basic Protocol

Any file transfer begins with a request to write a file (WRQ message) or an information (INFO message) telling where the file is located. The connection is automatically opened and the file is sent in fixed length blocks of typically 512 bytes. Each data packet contains one block of data (DATA message). A data packet of less than 512 bytes indicates termination of a transfer.

Most errors cause termination of the connection. Errors are caused by three types of events: not being able to satisfy the request (e.g. access violation), receiving a packet which cannot be explained by a delay or duplication in the network (e.g., an incorrectly formed packet), and losing access to a necessary resource (e.g., memory resources exhausted or access denied during a transfer).

MSDP recognizes only one error condition that does not cause termination, the source port of a received packet being incorrect.

This protocol is very restrictive, in order to simplify implementation. For example, the fixed length blocks makes allocation straightforward.

12.4 Relation to other Protocols

The basis for MSDP is TFTP elements as specified in [12] modified to apply for one way file transfer associated with multicast. TFTP options as specified by [14], [15] and [16] are also used. In addition, application specific options are defined. The TFTP elements are amended with an optional information carousel that supports scaling and increased speed of commissioning.

The MSDP is implemented on top of the Datagram protocol (UDP). Since Datagram is implemented on the Internet protocol, packets will have an Internet header, a Datagram header, and a MSDP header. Additionally, the packets will have an MPE or GSE header for transfer via DVB-S or S2. The MPE/GSE is assumed to apply a multicast MAC address that conforms to RFC1112 i.e. as for an Ethernet multicast MAC address. This is the only dependency on MPE/GSE. This aspect is readily transferable to other encapsulation methods, should they be adopted in future — assuming that they provide multicast functionality. In the following, we refer to MPE as required without repeating this comment.

As shown in Figure 12-1, the order of the contents of a packet will be MPE/GSE header, Internet header, Datagram header, MSDP header and the remainder of the MSDP packet. (This may or may not be data depending on the type of packet as specified in the MSDP header.) MSDP does not specify any of the values in the Internet header. On the other hand, the source and destination port fields of the Datagram header (its format is given in the appendix) are used by MSDP and the length field reflects the size of the MSDP packet. The Transfer IDentifiers (TID's) used by MSDP are

passed to the Datagram layer to be used as ports; therefore they must be between 0 and 65,535. The initialization of TID's is discussed in the section on initial connection protocol.

The MSDP header consists of a 2 byte opcode field which indicates the packet's type (e.g., DATA, etc.) These opcodes and the formats of the various types of packets are discussed further in the section on MSDP packets.

Figure 12-1 - Order of Headers

12.5 Basic MSDP Packet Formats

MSDP supports three types of packets, all of which have been mentioned above:

opcode	operation
2	Write request (WRQ)
3	Data (DATA)
255	Information (INFO)

The MSDP header of a packet contains the opcode associated with that packet.

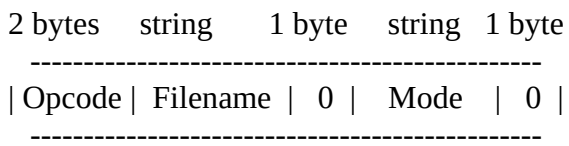

Figure 12-2 - WRQ packet

WRQ packets (opcode 2) have the format shown in Figure 12-2. The file name is a sequence of bytes in netascii terminated by a zero byte. The mode field contains the string "octet" (or any combination of upper and lower case, such as "OCTET", “Octet", etc.) in netascii. Octet mode is used to transfer a file that is in the 8-bit format of the indicated target type.

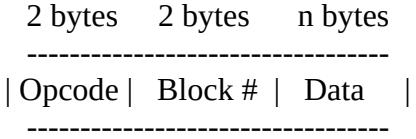

Figure 12-3 - DATA packet

Data is actually transferred in DATA packets depicted in Figure 12-3. DATA packets (opcode = 3) have a block number and data field. The block numbers on data packets begin with one and increase by one for each new block of data. This

restriction allows the program to use a single number to discriminate between new packets and duplicates. The data field is from zero to N bytes long. If it is exactly N bytes long, the block is not the last block of data. If it is from zero to (N-1) bytes long, it signals the end of the transfer. If the file ends with a final data segment of N bytes the transfer will be terminated by a block with a zero length data field. The default value of N is 512. Another block size can be indicated by the parameter ”blksize”.

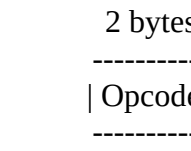

Figure 12-4 - INFO packet

Control information may be transferred regularly in INFO packets depicted in Figure 5-3. INFO packets (opcode = 255) are typically carrying additional parameters. INFO packets may occur anywhere in the stream.

12.6 Parameter transfer

The parameter transfer mechanism specified in this document allows file transfer parameters to be conveyed prior to the transfer using a mechanism that is consistent with MSDP's Request Packet format.

MSDP parameters are appended to the Write Request and Information packets.

Parameters are appended to an MSDP Write Request packet as follows:

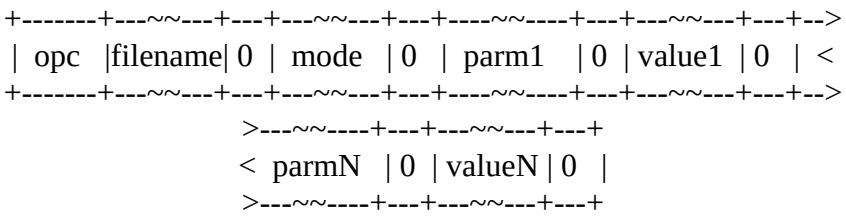

Figure 12-5 – MSDP parameters in WRQ packet

opc

The opcode field contains a 2, for Write Requests, as defined in [12].

filename

The name of the file to be read or written, as defined in [12]. This is a NULL-terminated field.

mode

The mode of the file transfer: "netascii", "octet", or "mail", as defined in [12]. This is a NULL-terminated field.

parm1

The name of the first parameter in case-insensitive ASCII (e.g. blksize). This is a NULL-terminated field.

value1

The value associated with the first parameter, in case-insensitive ASCII. This is a NULL-terminated field.

parmN, valueN

The final parameter/value pair. Each NULL-terminated field is specified in case-insensitive ASCII.

INFO messages have exactly the same layout as WRQ ones. The only difference is their opcode (255 instead of 2) and the fact that they are repeated at a higher rate than the WRQ. The WRQ are sent only once per loop of the data carousel, on the same redirected IP address and UDP port as the data, just before the file is sent. INFO messages may be sent at a higher rate and are sent on the default multicast group and port. The maximum length of INFO messages or write requests is 512 bytes.

The parameter names and values are all NULL-terminated, in keeping with the original request format. If multiple options are to be negotiated, they are appended to each other. The order in which parameters are specified is not significant. The maximum size of a request packet is 512 octets.

12.7 Parameters

Table 12-1 – MSDP parameters

Parameter	*Required functionality (O/M)*	*Presence of the parameter in message (O/M)*	Occurrence	Function	Value
blksize	***M***	***O***	WRQ, INFO	Set the DATA block size to another value than the default of 512 byte	Decimal number of bytes
Tsize	***M***	***M***	WRQ, INFO	Indicate the total transfer size	Decimal number of bytes
manufID	***M***	***M***	WRQ, INFO	Indicate OUI.	24 bit OUI as decimal value
Vendor specific parameters	***O***	***O***	WRQ, INFO	Maximum 10 vendor specific parameters. Server is supposed to support that many parameters. RCST implementation shall not consider the server is able to handle more.	Manufacturer specific
ver	***M***	***O***	WRQ, INFO	Current SW version in the SW distribution carousel, respective to the manufID and vendor specific parameters	Manufacturer specific
minver	***O***	***O***	WRQ, INFO	Indicates the minimum SW version required for log-on, with respect to manufID and vendor specific parameters	Manufacturer specific
method	***O***	***O***	WRQ, INFO	Indicates if the SW update method is different from the default ”immediate”. It can also be ”pending”, i.e. awaiting the next RCST restart.	”immediate” ”pending”
timeout	***O***	***O***	WRQ, INFO	Indicates the timeout when waiting for the next DATA packet, default value is TBC sec.	Decimal seconds
mgroup	***O***	***O***	INFO	Set a custom multicast group address respective to the manufID and vendor specific parameters	Dot separated decimal
port	***O***	***O***	INFO	Set a custom UDP port respective to the manufID and vendor specific parameters	Decimal
PID	***O***	***O***	INFO	Provide PID used for specific download	Decimal number of bytes

An M indicates parameters and functionality that must be supported. An O indicates parameters and functionality that may or should be supported. It some cases the lack of support of the latter type of functionality must be compensated through the capability of manual configuration at the RCST to allow the RCST to be entered into a system that utilizes all capabilities of the MSDP.

If a parameter occurs in an INFO message and the occurrence column states “WRQ, INFO” it should also occurs in the WRQ message.

The MSDP transmitter has to provide the mandatory parameters and may supply the other parameters as required for functionality and consistency.

12.8 Initial Connection Protocol

A transfer may be established by sending an INFO message on the default multicast group and UDP port. In this case the terminal will redirect to a new IP address and port and it will start reading the file on this multicast address and UDP port. A write request should be sent on the redirected IP address and UDP port to signal the beginning of the file. The terminal implementation may either wait for this write request and obtain the data blocks of the file in order (starting from block number 1) or it may just pick in anywhere in the data carousel (not waiting for the write requests) and it may download the file until all block numbers of that file have been received. There should be only one file per redirected IP address and port. In case we introduce new software for a certain terminal we first have to start the data carousel for this software and after that we can start sending the INFO messages. When we redraw old software we first have to stop sending the INFO messages and after that we stop the data carousel.

A transfer may also be established by sending WRQ messages on the default multicast group and port, that the RCST keeps listening even after redirection. In this case the terminal will use the default multicast IP address and UDP port for obtaining the data stream.

If an INFO messages does not contain any redirection a write request is to be expected on the default multicast group and UDP port.

The default multicast group and UDP port is 239.192.0.1 and 49152 unless specified otherwise in the RCST configuration.

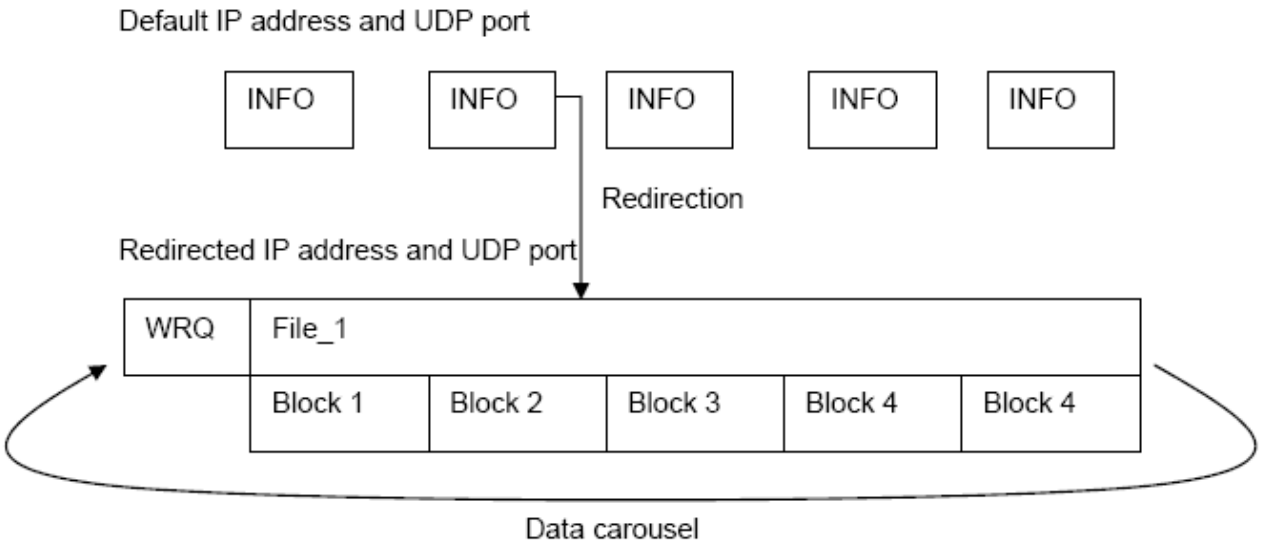

Figure 12-6 – MSDP redirection and carousel

12.9 Service location

Once the IP/DVB service is identified, the RCST will parse the PMT where it may get the PID value carrying the Multicast PID Mapping Table (MMT). From the MMT the RCST will know on which PID(s) it will get its multicast IP traffic.

Before initial logon the service has to be located in a PID that is identified by the PMT to carry MPE or by the MMT as applicable for the specific network. After initial logon and later logons, the service can be located in a PID as defined either by PMT, FIP or MMT as applicable for the specific network. In this functionality the SW distribution function is not differing from other IP multicast.

12.10 Signal sequence and Timing

The RCST has to be capable of receiving DATA packets at a pace of up to 50 kbps. This allows the RCST time to access the data storage. An RCST may have capability to support even higher rates. This is subject to manufacturer specification.

If the RCST has not received the next data packet within a given timeout it shall terminate the file reception and it shall retune to the default multicast group and UDP port.

In case the RCST implementation waits for the write request before storing any data packets, the RCST shall retune to the default multicast group and UDP port if such a request could not be received within 30 minutes.

An RCST that is not engaged in receiving DATA packets shall be capable of decoding INFO packets and write requests (WRQ) on its default multicast group and UDP port.

12.11 Flow diagram

The following procedure occurs after every log-on of the RCST.

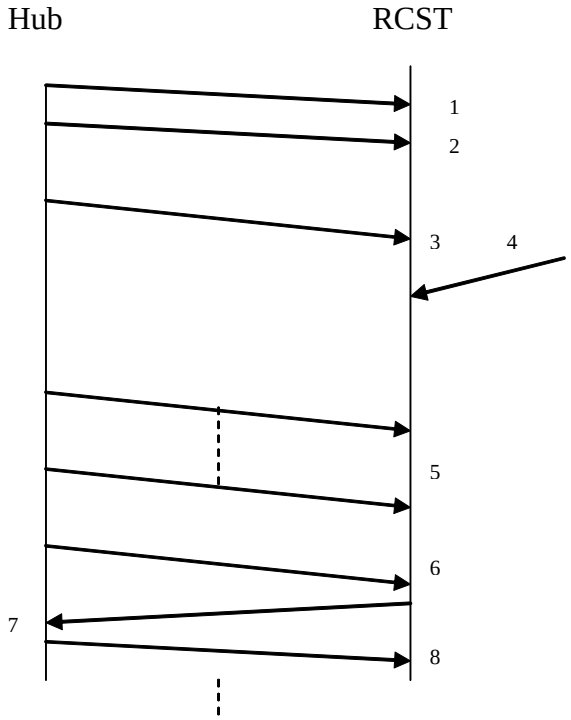

Figure 12-7 – MSDP flow diagram

1. An RCST in initialization mode tunes in to the FL and finds the PID for MMT in the PMT (or uses the PID assigned for IP if MMT is not referenced).
2. It finds the mapping of multicast addresses to PIDs in the MMT (if MMT is provided)
3. It opens up for reception on the configured multicast address and port (normally the default values) and decodes the stream. The stream may include manufacturer specific receiver redirection to another multicast address and port. It may also include additional vendor specific information.
4. The operator may directly set the multicast address and port to be used as entry point (can compensate for lack of redirection information)
5. The RCST opens up PID, Multicast address and port where it expects to find SW update, and receives a file.

As SW update download is completed, the RCST replaces the alternate SW load with the new downloaded SW and updates the *dvbRcsRcstAlternateSoftwareVersion* MIB parameter (see 17.2.3.1.6) accordingly.

6. In parallel the RCST will acquire the TBTP
7. The RCST can send logon request in CSC slot
8. The hub will respond with TIM-U

Vendor specific configuration can prevent the RCST to log-on until a given SW version has been downloaded. SW version can indeed be checked in the RCST capability field of the CSC burst (see [1]). Otherwise the RCST log-on will proceed in parallel with the SW download.

As an RCST continuously listens to the Forward Link Signaling, the SW download can be triggered at any time when multicast address and port are found.

12.12 Instant for SW download

PMT will indicate either:

- a PID to generally be used for reception of IP (including multicast)
- a PID to use for reception of the MMT (as recommended by SatLabs)

MMT will indicate mapping from supported multicast groups to PID. This mapping is used to identify the PID to be used for a specific multicast group.

12.13 PID differentiation

The SWU information channel can run alone on the default address (locally scoped, all systems) or can be multiplexed with a SW stream/carousel. SW streams can be separated on different multicast addresses that map to different PIDs through MMT.

12.14 Definition of multicast IP address

The SW information channel should be located on the default multicast address. Vendor specific redirection information should be located in this channel. Alternatively the target multicast address and port can be configured manually at the RCST.

The default multicast address should be under control by the network operator and should not be used for user traffic. It is within the local network control block address range. Note that if IGMP (Internet Group Management Protocol) is in use on the FL general IGMP queries can also occur addressed to this address. These will not interfere with MSDP that uses UDP.

The hub should block user traffic on the multicast addresses assigned to SW update to avoid any possibility of conflict. It is e.g. possible to select custom SWU multicast addresses from the Organization Local Scope multicast addresses. Another possibility is to use non-conflicting addresses from the Local Network Control Block, but note that packets with these addresses will not be forwarded by IP routers.

From [17]:

239.192.0.0/14 is defined to be the IPv4 Organization Local Scope, and is the space from which an organization should allocate sub- ranges when defining scopes for private use.

12.15 Transfer error handling

The RCST should filter duplicate packets and should also detect missing packets through the consecutive block numbering. The SW acceptance process of the RCST should include vendor specific consistency control of the received data.

12.16 Vendor specific methods

Additional vendor specific parameters should be included as required. The protocol is designed to be flexible (building on the TFTP).

The RCST should discard any unknown parameters.

12.17 Location of assigned PID

The RCST will detect the PID on which it will listen for the SWU information stream in the following manner:

Before logon:

- either directly on a PID identified by the PMT if the MMT is not defined
- or through MMT lookup

After logon

- either through MMT lookup if the MMT is defined, or
- through the PMT, alternatively also
- through the Forward Interaction Path descriptor (see [1]) received as logon response

13 Appendix C – Graceful Logoff Process from NCC

Figure 13-1 illustrates the graceful logoff process from NCC (for SNMPMISC option only).

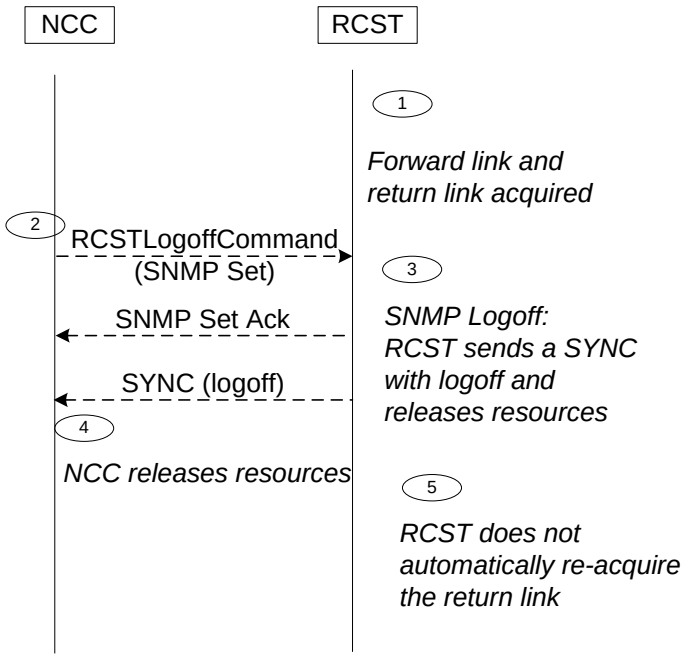

Figure 13-1 - Graceful Logoff Process from NCC

The graceful logoff process from NCC is described as follows:

1. The RCST has acquired the forward and return links.
2. An operator requests an RCST to logoff from the NCC. The NCC sends the SNMP Set – Logoff command (see *dvbRcsCtrlRcstLogoffCommand* parameter in 17.2.3.1.5) to the RCST.
3. The RCST receives the SNMP Set – Logoff command. The RCST replies with an SNMP Ack to the NCC. The RCST sends the SYNC with the logoff message. After, the RCST stops to transmit the SYNC burst. The RCST releases resources.
4. The Hub receives the SYNC with the logoff message. The Hub can release RL QoS resources.
5. The NCC keeps the management IP address association with the terminal for the next logon.
6. The RCST does not automatically re-acquire the return link unless there is traffic coming in. The RCST does not release the forward link

14 Appendix D – Mandatory Commissioning Parameters

The Commissioning Parameters that MUST be supported by all RCSTs are the following:

- MIB parameters defined in section *dvbRcsRcstInstall* MIB subgroup, defined in 17.2.3.1.3.
    a. Antenna Alignment State (Start, Deny, Continue, Stop, Success, Fail)
    b. CW Frequency [Hz]
    c. CW Max Duration [sec]
    d. CW Power [dBm]
    e. Co-pol reading [dB]
    f. X-pol reading [dB]
    g. Co-pol target [dB]
    h. X-pol target [dB]
    i. Time to wait in stand-by mode [sec]
    j. Target Es/N0 [dB]
- Operational Population Id (in 17.2.3.2.1)
- CW Enable command (in 17.2.3.1.5)
- Default IF power level (section 17.2.3.3.1)

The above mandatory MIB parameters SHALL be accessible through SNMP GET/SET, at least when RCST user traffic is disabled (see object *dvbRcsCtrlUserTrafficDisable* in 17.2.3.1.5).

15 Appendix E - Mandatory Installation Parameters

The Installation Parameters that MUST be supported by all RCSTs are the following:

- MIB parameters:
    a. Forward Configuration standard (see section 17.2.3.2.1)
    b. Forward Configuration Frequency and Polarization (see section 17.2.3.2.1)
    c. Forward Configuration symbol rate (see section 17.2.3.2.1)
    d. Forward Configuration modulation and inner FEC (see section 17.2.3.2.1)
    e. Forward Configuration roll-off for DVB-S2 (see section 17.2.3.2.1)
    f. Forward Status Network ID (see section 0)
    g. Forward Status Network name (see section0)
    h. Forward Status CNR and Received Power (see 0)
    i. Installation population ID (see section 17.2.3.2.1 and 0)
    j. RCST Management IP address (see section 17.2.3.1.2)
    k. RCST Location (minimum latitude, longitude and altitude) (see section 17.2.3.1.1)
    l. Default values for capacity parameters per RC (see section 17.2.3.1.4)

The support of the previous parameters is mandatory in the RCST MIB. They SHALL be accessible through SNMP GET/SET, at least when RCST user traffic is disabled (see object *dvbRcsCtrlUserTrafficDisable* in 17.2.3.1.5).

- SNMP write community string
- SNMP read community string

These previous parameters can be configured through CLI, HTTP or through configuration file download.

16 Appendix F – Mandatory commissioning and installation parameters supported via LAN interface

The mandatory parameters that MUST be accessible through the LAN interface are:

[*dvbRcs*RcstSystem] *dvbRcsSystemMibRevision*

[*dvbRcsRcstSystem*] *dvbRcsSystemLocation*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduAntennaGain*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduSspa*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduTxType*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduRxType*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduRxBand*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduRxLO*

[*dvbRcsRcstSystem*] *dvbRcsSystemOduTxLO*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkOamInetAddressType*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkOamInetAddress*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkOamInetAddressPrefixLength*

If EXTNETWORK feature is supported, [*dvbRcsRcstNetwork*] *dvbRcsNetworkOamInetAddressAssign*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkLanInetAddressType*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkLanInetAddress*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkLanInetAddressPrefixLength*

If EXTNETWORK feature is supported,

[*dvbRcsRcstNetwork*] *dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressType*,

dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress

and *dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressPrefixLength*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkConfigFileDownloadUrl*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkConfigFileUploadUrl*

If INSTALL_LOG feature is supported, [*dvbRcsRcstNetwork*] *dvbRcsNetworkInstallLogFileDownloadUrl*

[*dvbRcsRcstNetwork*] *dvbRcsNetworkLogFileUploadUrl*

If INSTALL_LOG feature is supported, [*dvbRcsRcstNetwork*] *dvbRcsNetworkInstallLogFileUploadUrl*

[*dvbRcsRcstInstall*] *dvbRcsInstallCwFrequency*

[*dvbRcsRcstInstall*] *dvbRcsInstallCwPower*

[*dvbRcsRcstQos*] Default values for capacity parameters per RC

[*dvbRcsRcstControl*] *dvbRcsCtrlRebootCommand*

If EXTCONTROL feature is supported, [*dvbRcsRcstControl*] *dvbRcsCtrlRcstTxDisable*

[*dvbRcsRcstControl*] *dvbRcsCtrlCwEnable*

IfEXTCONTROL feature is supported, [*dvbRcsRcstControl*] *dvbRcsCtrlOduTxReferenceEnable*

If EXTCONTROL feature is supported, [*dvbRcsRcstControl*] *dvbRcsCtrlOduTxDCEnable*

If EXTCONTROL feature is supported, [*dvbRcsRcstControl*] *dvbRcsCtrlOduRxDCEnable*

[*dvbRcsRcstControl*] *dvbRcsCtrlDownloadFileCommand*

[*dvbRcsRcstControl*] *dvbRcsCtrlUploadFileCommand*

[*dvbRcsRcstControl*] *dvbRcsCtrlActivateConfigFileCommand*

If EXTCONTROL feature is supported, [*dvbRcsRcstControl*] *dvbRcsCtrlRcstLogonCommand*

If EXTCONTROL feature is supported, [*dvbRcsRcstControl*]*dvbRcsCtrlRcstLogoffCommand*

[*dvbRcsRcstState*] *dvbRcsRcstMode*

[*dvbRcsRcstState*] *dvbRcsRcstFwdLinkStatus*

If EXTSTATUS feature is supported, [*dvbRcsRcstState*]*dvbRcsRcstRtnLinkStatus*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartPopId*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartFrequency*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartPolar*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartFormat*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartRolloff*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartInnerFec*

[*dvbRcsFwdConfig*] *dvbRcsFwdStartSymbolRate*

[*dvbRcsFwdStatus*] *dvbRcsFwdStatusNetId*

[*dvbRcsFwdStatus*] *dvbRcsFwdStatusCnr*

[*dvbRcsFwdStatus*] *dvbRcsFwdStatusBer*

[*dvbRcsRtnConfig*] *dvbRcsRtnConfigDefIfLevel*

Note that MIB parameters that are optional in the MIB are also optional through LAN access. But if an optional parameter is supported in the MIB and if it is in the above list, its access through the LAN interface has to be supported.

17 Appendix G – MIB

17.1 Structure for DVB-RCS interface MIB

Management information is viewed as a collection of managed objects residing in a virtual information store, termed the Management Information Base (MIB). Collections of related objects are defined in MIB modules.

The DVB-RCS interface MIB [33] is located under the *iso.org.dod.internet.mgnt.mib-2* branch. Three sub-groups, *system*, *interfaces* and *ifMIB*, will be used to identify the DVB-RCS interface objects.

The DVB-RCS specific elements of the MIB are included into the *mib-2* subtree starting with 1.3.6.1.2.1.10, which corresponds to *iso.org.dod.internet.mgmt.mib-2.transmission*.
IANA has assigned Transmission number 239 and name *dvbRcsMib*, corresponding to the DVB-RCS MAC layer (see [32]).
The MIB module defining the DVB-RCS Transmission MIB is labeled *dvbRcsMib* and will be denoted DVB-RCS MIB in the following. It is identified as follows:

Table 17-1 - Object Identifier for the DVB-RCS MIB

Descriptor	OBJECT IDENTIFIER value
dvbRcsMib	{ mib-2 transmission 239 }

This DVB-RCS MIB module is formally specified in an ASN.1 MIB file description which is annexed to the present document. Its formal definition in [33] takes precedence over the description of the MIB that follows here below.

In addition to these standard MIB modules, the *sysObjectID* from the system group of the mib-2 is used to provide an OID pointer to the vendor-specific DVB-RCS MIB.

The following MIB sets are required in the DVB-RCS MIB of a RCST, as shown on Figure 17-1:

1. A DVB-RCS interface MIB definition which requires only the *system* (RFC 1213 [11]) and the *interfaces* (RFC 2863 [23]) sub-groups. DVB-RCS specific elements, defined in the DVB-RCS MIB, are situated in the mib-2 *transmission* branch.
2. A DVB-RCS vendor-specific MIB (vendor-specific)

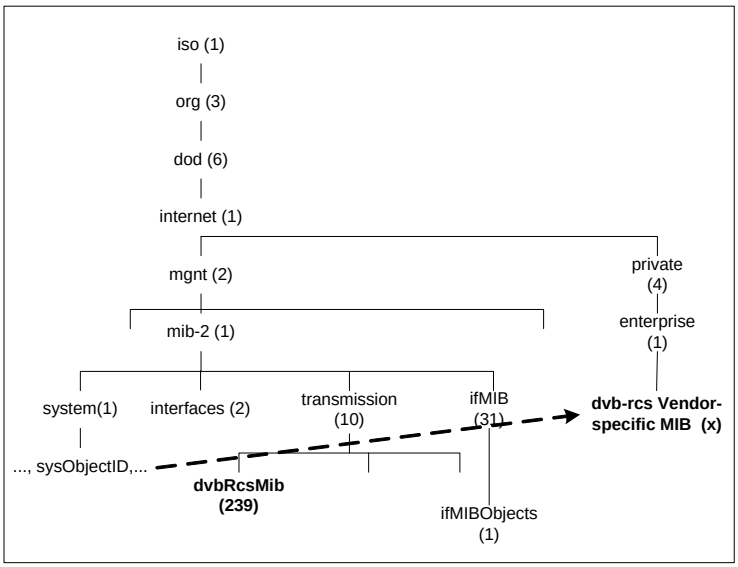

Figure 17-1 - DVB-RCS interface MIB Tree

17.1.1 DVB-RCS Interface MIB Definition

The DVB-RCS MIB is described in this section. IANA [31] has assigned three *ifType* labels for DVB-RCS. These values are important as they are the standardized way of identifying the interface type. Interface type is not identified by a convention related to the index values – the index value has local significance only. The media specific MIB for DVB-RCS should specify the use of the parameters of [11] for these interface types. Each DVB-RCS terminal will thus have at least the three following interfaces:

- *dvbRcsMacLayer* (239), -- DVB-RCS MAC Layer
- *dvbTdm* (240), -- DVB Satellite TDM
- *dvbRcsTdma* (241), -- DVB-RCS TDMA

The additional Ethernet interface is used on the LAN side of the RCST.

Table 17-2 – ifType labels

ifType Labels	ifTtype value	Description
dvbRcsMacLayer	239	DVB-RCS MAC Layer represents the air interface of a Return Channel Satellite Terminal (RCST), as defined in the DVB-RCS Standard. This interface supports star and mesh networks and is bi-directional.
dvbTdm	240	DVB-RCS Physical link based on Time Division Multiplexing. It corresponds to the forward link of a RCS transparent system or the downlink of a RCS regenerative system. It is based on either DVB-S or DVB-S2 standard. In the DVB-RCS context, this interface is uni-directional.
dvbRcsTdma	241	DVB-RCS Physical link based on Time Division Multiple Access. It corresponds to the return or mesh link of a RCS transparent system or the uplink of a RCS regenerative system. It is based on the DVB-RCS standard. In the context of star transparent and mesh regenerative DVB-RCS systems, this interface is uni-directional. In the context of mesh transparent DVB-RCS systems, this interface is bi-directional.

Two scenarios, the Access network and the transparent mesh over multiple beams are presented with the structure of different interfaces.

This list of *ifType* labels provides full flexibility to define a DVB-RCS network. Two examples are given to illustrate only some possibilities. Figure 17-2 illustrates a DVB-RCS Access network with DVB-S and DVB-RCS. In this network, ATM encapsulation is used in the DVB-RCS uplink. DVB-S2 can be used instead of DVB-S in the down link. MPEG can be used as well instead of ATM logical link. The *ifType* for ATM is 80 and the *ifType* for MPEG is 214.

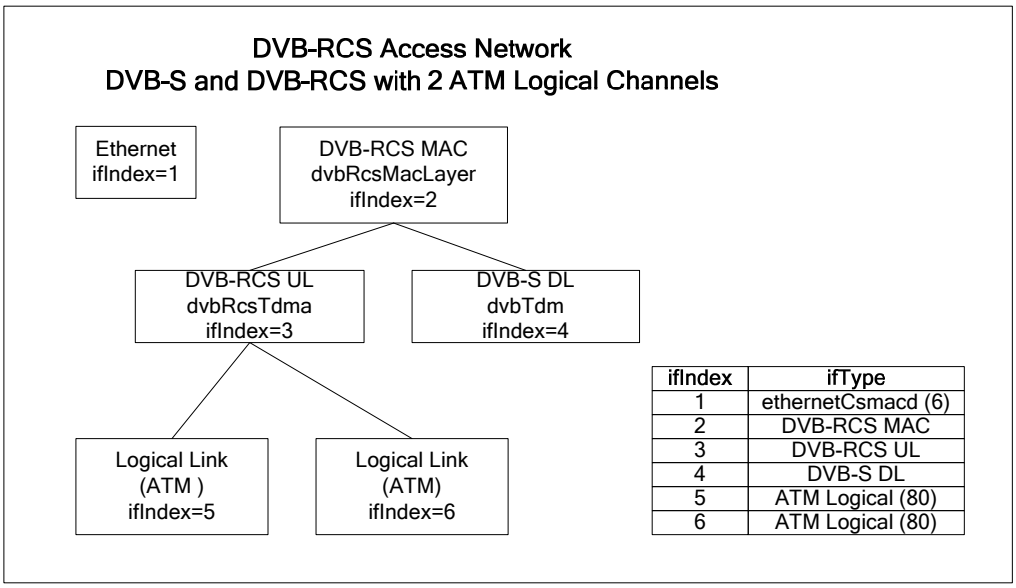

Figure 17-2 - DVB-RCS Access Network

The *ifStackTable* implementation for this example is:

ifStackHigherLayer	*ifStackLowerLayer*
0	1
0	2
1	0
2	3
2	4
3	5
3	6
4	0
5	0
6	0

The Figure 17-3 illustrates a DVB-RCS Access network and transparent mesh over multiple beams with DVB-S2 and DVB-RCS. In this network, ATM encapsulation is used in the DVB-RCS uplink.

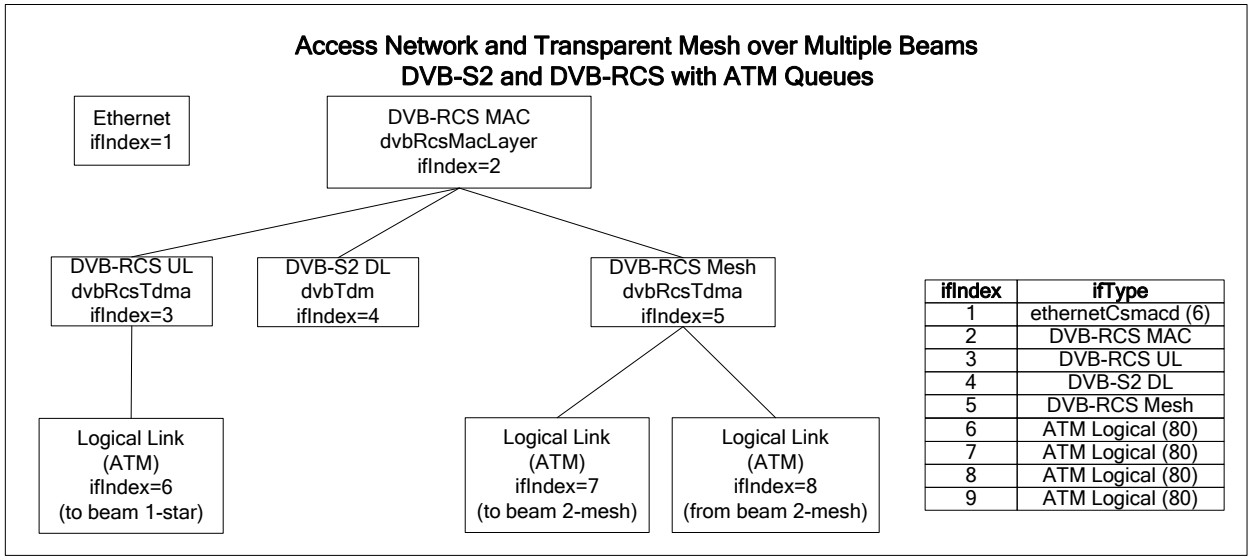

Figure 17-3 - DVB-RCS Access Network and Transparent Mesh with ATM in the Return Link

The *ifStackTable* implementation for this example is:

ifStackHigherLayer	*ifStackLowerLayer*
0	1
0	2
1	0
2	3
2	4
2	5
3	6
4	0
5	7
5	8
6	0
7	0
8	0

Per DVB-RCS interface, only some counters from the interfaces group of MIB-II will be supported. The Table 17-3 presents the (M)andatory and (O)ptional counters per DVB-RCS interface. The counters for DVB-RCS MAC, DVB-RCS uplink and DVB-S or DVB-S2 down link interfaces are required. The counters for other sub-interfaces (ATM and MPEG) are optional.

Table 17-3 - MIB-II Counters for DVB-RCS Interfaces

OID	Name	dvbRcsMacLayer	dvbRcsTdma	dvbTdm	ATM Logical Link (UL)	MPEG Logical Link (UL)	ATM Logical Link (DL)
	Interfaces MIB						
2	ifTable						
2.1	ifEntry						
2.1.1	ifIndex	M	M	M	O	O	O
2.1.2	ifDescr	M	M	M	O	O	O
2.1.3	ifType	M	M	M	O (80)	O (214)	O (80)
2.1.4	ifMtu	M (1500)	M (1500)	M (1500)	O (53)	O (188)	O (53)
2.1.5	ifSpeed		M	M			
2.1.6	ifPhysAddress	M (MAC @)			O (ATM VCC)	O (PID)	O (ATM VCC)
2.1.7	ifAdminStatus	M	M	M	O	O	O
2.1.8	ifOperStatus	M	M	M	O	O	O
2.1.9	ifLastChange	M	M	M	O	O	O
2.1.10	ifInOctets	M	M (Rx only)	M			O
2.1.11	ifInUcastPkts	M	M (Rx only)	M			O
2.1.13	ifInDiscards	M	M (Rx only)	M			O
2.1.14	ifInErrors	M	M (Rx only)	M			O
2.1.15	ifInUnknownProtos	M	M (Rx only)	M			O
2.1.16	ifOutOctets	M	M		O	O	
2.1.17	ifOutUcastPkts	M	M		O	O	

OID	Name	dvbRcsMacLayer	dvbRcsTdma	dvbTdm	ATM Logical Link (UL)	MPEG Logical Link (UL)	ATM Logical Link (DL)
2.1.19	ifOutDiscards	M	M		O	O	
2.1.20	ifOutErrors	M	M		O	O	
	ifMIBObjects MIB						
1	ifXTable						
1.1	ifXEntry						
1.1.1	ifName	M	M	M	O	O	O
1.1.2	ifInMulticastPkts	M	M (Rx only)	M			O
1.1.3	ifInBroadcastPkts	M	M (Rx only)	M			O
1.1.4	ifOutMulticastPkts	M	M		O	O	
1.1.5	ifOutBroadcastPkts	M	M		O	O	

17.1.2 DVB-RCS MIB Content

The DVB-RCS MIB specification covers the following object groups:

- Objects for terminal installation
- Objects for QoS
- Objects for network configuration
- Objects for return link configuration
- Objects for terminal status
- Objects to command the terminal (reboot, logoff, logon, download file, upload file)
- Object pointing to an optional proprietary device MIB
- Units of conformance, which are object groups allowing to specify conformance statements for SatLabs defined options or features(see [6] for the definition of SatLabs options and features).

The units of conformance group objects into formal object groups in order to support specification of requirements to conformance to mandatory object groups and optional object groups. A corresponding MODULE-COMPLIANCE specification is also defined in the MIB module following the recommendations of [20].

17.2 DVB-RCS MIB Definition

17.2.1 Access Rights

The MAX-ACCESS clause defines whether it makes "protocol sense" to read, write and/or create an instance of the object, or to include its value in a notification. This is the maximal level of access for the object. (This maximal level of access is independent of any administrative authorization policy.) These values are ordered, from least to greatest: "not-accessible", "accessible-for-notify", "read-only", "read-write", "read-create". If any columnar object in a conceptual row has "read-create" as its maximal level of access, then no other columnar object of the same conceptual row may have a maximal access of "read write". (Note that "read-create" is a superset of "read-write".) The write and read access rights of any SNMP object are defined/identified according to the different users/entities. In the RCST MIB definition within the present document, the following notations are used in the scope of the access rights:

- 'W' stands for 'Write' access
- 'R' stands for 'Read' access
- ‘C’ stands for ‘Create’
- 'N' stands for 'Not-Accessible' access

The access rights to a particular SNMP object are defined cross-checking both the maximum level of access of that SNMP object and the access rights granted to the entity according to its community name.

Table 17-4 describes the relationship between SNMPv2 MIB MAX-ACCESS Value and Protocol Access Mode.

Table 17-4 - relationship between SNMPv2 MIB MAX-ACCESS Value and Protocol Access Mode

SNMPv2 Protocol Operation		
MAX-ACCESS **Value**	READ-ONLY	READ-WRITE
read-only	Available for get and trap operations	
read-write	for get and trap operations	Available for get, set, and trap operations
read-create	Available for get and trap operations	Available for get, set, create, and trap operations
accessible-for-notify	Available for trap operations	
not-accessible	Unavailable	

The table shall be understood as follows. Defining an SNMP object, one has to give it a maximum access. This access is described by the different rows of the table. But every object belongs to the different views defined for each community. Moreover, its access rights are defined in relation with this community. This can be read in the columns of Table 17-4. The intersection of each row and each column defines a kind of 'availability' of the SNMP object as far as the SNMP actions (get, set, trap, etc) are concerned. Hence, when the view access rights of an SNMP object in a particular view are defined as 'Read-Only' while its maximum access rights are 'Read-Write', this means that the object is somehow available to GET and TRAP operations. Indeed, this object is at most readable and, hence, no SET action can be done.

17.2.2 SNMP Objects Syntax

Each SNMP object is of a specific type. There exist numerous types and those are defined in different RFCs. The following comments emanate from those RFCs. MIB modules are written using an adapted subset of OSI's Abstract Syntax Notation One, ASN.1 [34]. The Structure of Management Information (SMI) specified by IETF defines that adapted subset and assigns a set of associated administrative values. This section summarizes the main object types and textual conventions used in the SatLabs recommended MIB modules and refer to the relevant standards where full definitions can be found.

INTEGER, OCTET STRING and OBJECT IDENTIFIER are base types defined in [34].

Integer32 and Unsigned32 are defined in RFC 2578 (SMIv2) [18]. The *Integer32* type represents integer-valued information between -2^{31} and $2^{31}-1$ inclusive (-2147483648 to 2147483647 decimal) while *Unsigned32* represents integer-valued information between 0 and $2^{32}-1$ inclusive (0 to 4294967295 decimal). This type is indistinguishable from the *INTEGER* type. Both the *INTEGER* and *Integer32* types may be sub-typed to be more constrained than the *Integer32* type. The *INTEGER* type (but not the *Integer32* type) may also be used to represent integer-valued information as named-number enumerations. In this case, only those named-numbers so enumerated may be present as a value. Note that although it is recommended that enumerated values start at 1 and be numbered contiguously, any valid value for *Integer32* is allowed for an enumerated value and, further, enumerated values need not be contiguously assigned.

The *TimeTicks* type, defined in SMIv2 [18], represents a non-negative integer which represents the time, modulo 2^{32} (4 294 967 296 decimal), in hundredths of a second between two epochs. When objects are defined using this type, the description of the object identifies both of the reference epochs. The *TimeStamp* textual convention is defined in [19] and is based on the *TimeTicks* type. With a *TimeStamp*, the first reference epoch is defined as the time when *sysUpTime* (MIB-II system SNMP object) was zero, and the second reference epoch is defined as the current value of *sysUpTime*.

RowStatus type is a textual convention defined in [19] and shall be implemented as such. This syntax is mainly used to declare dynamic tables.

SnmpAdminString is a textual convention, defined in RFC 3411 (SNMP FRAMEWORK MIB) [26], representing an octet string containing administrative information, preferably in human-readable form.

InetAddressType, *InetAddress*, *InetAddressPrefixLength* and *InetPortNumber* are textual conventions, defined in RFC 4001 (INET ADDRESS MIB) [29], allowing to represent both IPv4 and IPv6 Internet addresses. A generic Internet address consists of two objects: one whose syntax is *InetAddressType*, and another whose syntax is *InetAddress*. The value of the first object determines how the value of the second is encoded. The *InetAddressPrefixLength* object can be added to identify the Internet network address prefix.
SatLabs implementations are only required to support IPv4 addresses (*ipv4* type).
InetPortNumber represents a 16 bit port number of an Internet transport layer protocol. Port numbers are assigned by IANA.

Uri is a textual convention, defined in RFC 5017 (URI TC MIB) [30], representing a Uniform Resource Identifier as an OCTET STRING.

Dscp and *DscpOrAny* are textual conventions, defined in RFC 3289 (DIFFSERV DSCP TC MIB) [25], representing Differentiated Services Code-Points. They are represented by Integer32 (0..63) and Integer32 (-1 | 0..63) respectively. The value -1 is used to indicate a wild card i.e. any value.

17.2.3 DVB-RCS MIB Definition

The DVB-RCS MIB shall be included into the *mib-2 transmission* sub-tree: with 1.3.6.1.2.1.10, which corresponds to *iso.org.dod.internet.mgnt.mib-2.transmission*.
The different sub-groups of the DVB-RCS MIB are illustrated in Figure 17-4.

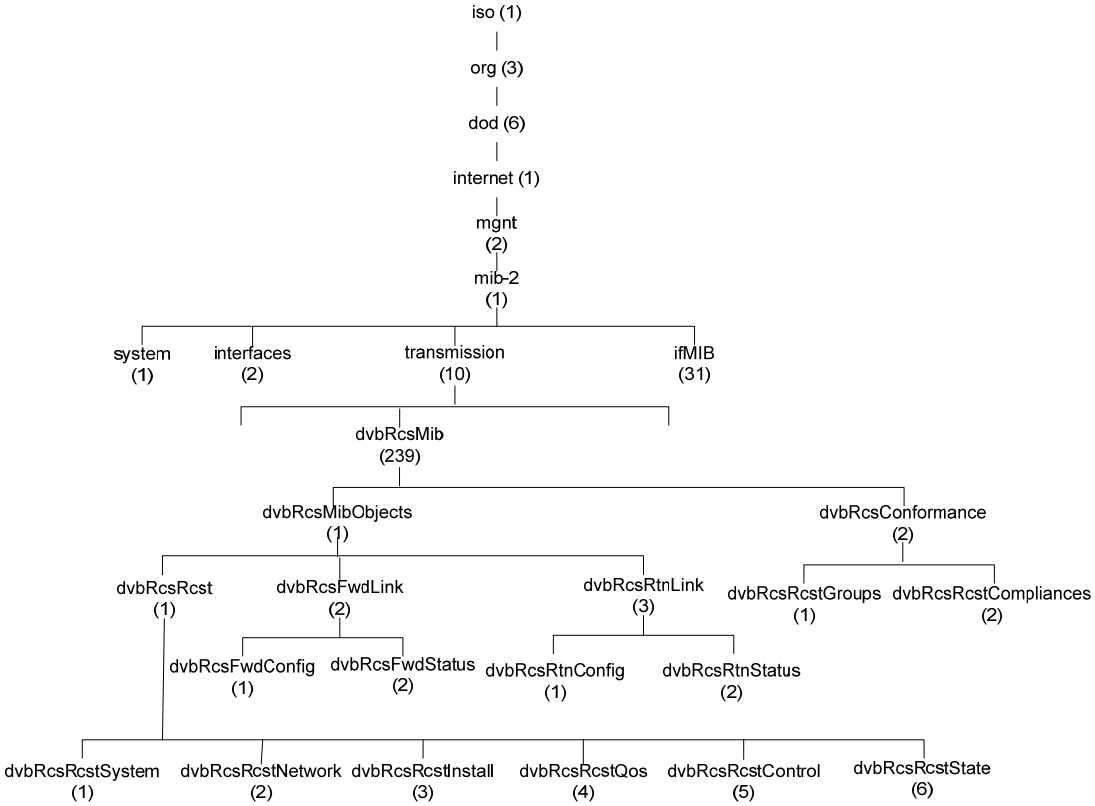

Figure 17-4 - DVB-RCS MIB Structure

The *dvbRcsMibObjects* sub-tree includes all the managed objects of the DVB-RCS MIB. These objects are defined in the following sections.

The *dvbRcsConformance* sub-tree includes the compliance statements for DVB-RCS terminals that are compliant with the present SatLabs SSR. The managed objects are grouped into formal object groups (i.e. units of conformance) according to the relation to specific SatLabs options. The conformance statements (MODULE-COMPLIANCE specification in *dvbRcsRcstCompliances* sub-tree) and the units of conformance (*dvbRcsRcstGroups* sub-tree) are specified in the MIB file which is annexed to the present document.

SatLabs compliance is achieved if at least one access level (see 17.2.1) allows the requested access mode, as defined in the ‘Access’ column of the following tables defining the MIB.

17.2.3.1 dvbRcsRcst group (1)

17.2.3.1.1 dvbRcsRcstSystem subgroup (1)

The MIB objects in this group, defined in Table 17-5, gather some basic information that would allow anyone to trace the history - the life - of the RCST as well as to get a complete description of its constitution on the component point of view, including the SatLabs options support statement and the ODU structural entities (LNB, BUC and antenna). Many of the parameters will be defined at installation.

This group contains description parameters related to the RCST type (ODU type) and location. These parameters are believed to stay unchanged once it has been defined during installation. Modification of hardware equipment, maintenance operations and geographical re-location may require an update of those MIB objects. Note that *dvbRcsRcstSystem.dvbRcsSystemLocation* object gives the location of the ODU antenna, which is needed for network operation, while the *system.sysLocation* (MIB-II SNMP OID) provides the location of the IDU unit, which can not be used for the same purpose.

Table 17-5 - *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstSystem* sub-group definition

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
1	...dvbRcsSystemMibRevision	DisplayString	R	This object allows the SNMP agent to report the implemented MIB module revision. The supported REVISION of this module is reported.	M	7.2
2	…dvbRcsSystemSatLabsProfilesDeclaration	DvbRcsSatLabsProfileMap See MIB file for	R	Indicates the SatLabs profiles supported as defined in the SatLabs System Recommendations. A value of 1 indicates that the respective option is supported. The mapping to the profiles is to be understood as described here. (0) refers to the most	M	See [6]

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
		Textual Convention.		significant bit. dvbs(0) -> DVBS profile (DVB-S support) dvbs2ccm(1) -> DVB-S2 CCM profile (CCM support) dvbs2acm(2) -> DVB-S2 ACM profile (CCM, VCM and ACM support)		
3	…dvbRcsSystemSatLabsOptionsDeclaration	DvbRcsSatLabsOptionMap See MIB file for Textual Convention.	R	Indicates the SatLabs options supported as defined in the SatLabs System Recommendations. A value of 1 indicates that the respective option is supported. The mapping to the options is to be understood as described here. (0) refers to the most significant bit. mpegTrf(0) -> MPEG_TRF coarseSync(1) -> COARSE_SYNC wideHop(2) -> WIDE_HOPP fastHop(3) -> FAST_HOPP dynamicMfTdma(4) -> Dynamic_MF_TDMA contentionSync(5) -> CONTENTION_SYNC qpskLow(6) -> QPSKLOW mod16Apsk(7) -> 16APSK mod32Apsk(8) -> 32APSK normalFec(9) -> NORMALFEC multiTs(10) -> MULTITS	M	See [6]

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
				gsTs(11) -> GSTS enhQoS(12) -> ENHQOS pep(13) -> PEP http(14) -> HTTP ftp(15) -> FTP dns(16) -> DNS chIdStrict(17) -> CHID_STRICT nlid(18) -> NLID snmpMisc(19) -> SNMPMISC The support of specific options mandates the support of specific objects and access levels.		
4	…dvbRcsSystemSatLabsFeaturesDeclaration	DvbRcsSatLabsFeatureMap See MIB file for Textual Convention.	R	Indicates the optional compatibility features and minor options supported as defined in the SatLabs System Recommendations. A value of 1 indicates that the respective feature is supported. The mapping to the features is to be understood as described here. (0) refers to the most significant bit. rcstPara(0) -> RCST_PARA installLog(1) -> INSTALL_LOG enhClassifier(2) -> ENHCLASSIFIER routeId(3) -> ROUTE_ID oduList(4) -> ODULIST	M	See [6]

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
				extNetwork(5) -> EXTNETWORK extControl(6) -> EXTCONTROL extConfig(7) -> EXTCONFIG extStatus(8) -> EXTSTATUS mpaf(9) -> MPAF The support of specific features mandates the support of specific objects and access levels.		
5	...dvbRcsSystemLocation	DisplayString	R W	Physical location of the ODU antenna expressed as Longitude, Latitude and Altitude. The string shall have 31 characters in the following format: <xxxx.xxx>,<a>,<yyyyy.yyy>,<b>,<zzzz.z>,M where x, y and z represents digits, a=N or S, b=E or W, Reading the digits from left to right: 'x' – 7 latitude digits; x digits 1-2 contain the degrees, x digits 3-7 contain the minutes in decimal; 'y' – 8 longitude digits; y digits 1-3 contain the degrees, y digits 4-8 contain the minutes in decimal; 'z' 5 altitude digits; meters above sea level in decimal; '.' is the decimal point;	M	7.2

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
				‘,’ is the field separator; ‘M’ is the indicator for altitude meters. This format is a modified subset of the NMEA 0183 (National Marine Electronics Association, Interface Standard) format for Global Positioning System Fix Data. This location and the satellite position are used to calculate the RCST-satellite path delay. Note: The *system.sysLocation* object of MIB-II provides physical location of the IDU unit.		
6	...dvbRcsSystemOduAntenna Size	DisplayString UNITS “cm”	R R W	This object gives the diameter of the antenna. This value shall be given in centimeters. It is defined at Installation. The object can be used in conjunction with environmental requirements.	M (read) if ODULIST is not supported (write)	7.2
7	…dvbRcsSystemOduAntenna Gain	Unsigned32 UNITS “x0.1 dBi”	R R W	This field describes the antenna peak gain of the ODU and shall be defined by the installer. The gain shall be given in tenth of dBi for more flexibility, i.e. 46.5 dBi will be represented by 465. Defined at installation.	M (read) if ODULIST is not supported (write)	7.2
8	...dvbRcsSystemOduSspa	Unsigned32 UNITS “x0.1 W”	R R W	This field describes the power level of the Solid State Power Amplifier installed in the ODU and shall be defined by the installer. The power shall be given in tenth of a Watt, i.e. 0.5 W	M (read) if ODULIST is not supported	7.2

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
				will be represented by 5, 1 W by 10 and 2 W by 20. Defined at installation.	(write)	
9	…dvbRcsSystemOduTxType	SnmpAdminString	R R W	Describes the type of transmitter installed in the ODU. Write access is only required if the RCST does not provide the information.	M (read) if ODULIST is not supported (write)	7.2
10	…dvbRcsSystemOduRxType	SnmpAdminString	R R W	Describes the type of LNB installed in the ODU, with information such as vendor type, output type (single, twin, quad,…), etc.. Write access is only required if the RCST does not provide the information.	M (read) if ODULIST is not supported (write)	7.2
11	…dvbRcsSystemOduRxBand	INTEGER	R R W	Describes whether High Band or Low Band is selected in the LNB. Specifying High Band results in activation of an 18-26 kHz tone with 0.4-0.8 Vpp in the Rx IFL cable. (0) oduHighRxBand (1) oduLowRxBand	M (read) if ODULIST is not supported (write)	7.2
12	…dvbRcsSystemOduRxLO	Unsigned32 UNITS “x100 Hz”	R R W	Frequency of LNB Local Oscillator (in 100 Hz)	M (read) if ODULIST is not supported (write)	7.2
13	…dvbRcsSystemOduTxLO	Integer32 UNITS “x100 Hz”	R R W	Frequency of BUC Local Oscillator (in 100 Hz)	M (read) if ODULIST is	7.2

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
					not supported (write)	
14	…dvbRcsSystemIduPep	OBJECT IDENTIFIER	N	Generic PEP control	M	See [6]
14.1	…dvbRcsSystemIduPep.dvbRcsTcpPep	INTEGER	R W	Status and control of embedded TCP PEP: 0 - disabled or not implemented 1 - enabled	M	See [6]
14.2	…dvbRcsSystemIduPep.dvbRcsHttpPep	INTEGER	R W	Status and control of embedded HTTP PEP: 0 - disabled or not implemented 1 - enabled	M	See [6]
15	…dvbRcsOduTx	OBJECT IDENTIFIER	N	Generic ODU structural entities	ODULIST	
16	…dvbRcsOduRx	OBJECT IDENTIFIER	N	Generic ODU structural entities	ODULIST	
17	…dvbRcsOduAntenna	OBJECT IDENTIFIER	N	Generic ODU structural entities	ODULIST	
15.1	…dvbRcsOduTxTypeTable	SEQUENCE OF DvbRcsOduTxTypeEntry	N	This table contains the identification of each well-known BUC type supported by the IDU and provides its associated index.	ODULIST	
15.1.1	…dvbRcsOduTxTypeEntry	DvbRcsOduTxTypeEntry SEQUENCE	N	An entry in the BUC type table.	ODULIST	

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
15.1.1.1	…dvbRcsOduTxTypeIndex	Unsigned32 (1..32)	N	Index for the BUC type.	ODULIST	
15.1.1.2	…dvbRcsOduTxTypeDescription	SnmpAdminString	R	Text-based identification of a BUC type.	ODULIST	
15.1.1.3	…dvbRcsOduTxType	Unsigned32	R W	Index of the selected BUC type.	ODULIST	
16.1	…dvbRcsOduRxTypeTable	SEQUENCE OF DvbRcsOduRxTypeEntry	N	This table contains the identification of each well-known LNB type supported by the IDU and provides its associated index.	ODULIST	
16.1.1	…dvbRcsOduRxTypeEntry	DvbRcsOduRxTypeEntry SEQUENCE	N	An entry in the LNB type table.	ODULIST	
16.1.1.1	…dvbRcsOduRxTypeIndex	Unsigned32 (1..32)	N	Index for the LNB type.	ODULIST	
16.1.1.2	…dvbRcsOduRxTypeDescription	SnmpAdminString	R	Text-based identification of an LNB type.	ODULIST	
16.1.1.3	…dvbRcsOduRxType	Unsigned32	R W	Index of the selected LNB type.	ODULIST	
17.1	…dvbRcsOduAntennaTypeTable	SEQUENCE OF DvbRcsOduAntennaTypeEntry	N	This table contains the identification of each well-known antenna type supported by the IDU and provides its associated index.	ODULIST	
17.1.	…dvbRcsOduAntennaTypeEn	DvbRcsOduAntenna	N	An entry in the antenna type table.	ODULIST	

OID	Name dvbRcsRcstSystem…	Syntax	Access	Description	Satlabs Applicability	Section
1	try	TypeEntry SEQUENCE				
17.1.1.1	…dvbRcsOduAntennaTypeIndex	Unsigned32 (1..32)	N	Index for the antenna type.	ODULIST	
17.1.1.2	…dvbRcsOduAntennaTypeDescription	SnmpAdminString	R	Text-based identification of an antenna type.	ODULIST	
17.1.1.3	…dvbRcsOduAntennaType	Unsigned32	R W	Index of the selected antenna type.	ODULIST	

17.2.3.1.2 dvbRcsRcstNetwork subgroup (2)

This subgroup contains all the MIB objects related to network parameters. The different MIB objects are defined in Table 17-6.

In this subgroup, two objects have been defined in order to differentiate between control and user traffic and associate them with a physical interface. Both *dvbRcsNetworkLanInetAddress* (Traffic) and *dvbRcsNetworkOamInetAddress* (OAM) provide the value of the IP address of, respectively, the user traffic and the control flow.

Table 17-6 – *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstNetwork* subgroup definition

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
1	…dvbRcsNetworkOamInetAddressTyp e	InetAddress Type ipv4(1)	R W	The type of Internet address of *dvbRcsNetworkOamInetAddress*. If the terminal OAM Internet address is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	M	5

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
2	...dvbRcsNetworkOamInetAddress	InetAddress SIZE(4)	R W	OAM IP Address of the RCST. This object used with both *ip* and *interfaces* MIB-II subgroups. It uniquely determines the interface through which OAM traffic passes. The OAM IP address may be statically or dynamically assigned. It is system dependent whether the OAM IP address and the Traffic IP address are the same address. If the terminal has no OAM Internet address assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsNetworkOamInetAddressType* object. An implementation is only required to support IPv4 addresses.	M	5
3	...dvbRcsNetworkOamInetAddressPrefixLength	InetAddress PrefixLength	R W	Prefix length for the OAM IP Address. If this address prefix is unknown or does not apply, the value is zero.	M	5
4	...dvbRcsNetworkOamInetAddressAssign	INTEGER	R	Identifies whether the OAM IP address is statically (1) or dynamically (2) assigned. (1) oamInetAddressStatic (2) oamInetAddressDynamic	EXTNETWORK	5

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
5	…dvbRcsNetworkLanInetAddressType	InetAddress Type ipv4(1)	R W	The type of Internet address of *dvbRcsNetworkLanInetAddress*. If the terminal Internet address on the LAN interface is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	M	5
6	…dvbRcsNetworkLanInetAddress	InetAddress SIZE(4)	R W	IP address of the LAN interface of the terminal. If the terminal has no Internet address assigned on the LAN interface or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsNetworkLanInetAddressType* object. An implementation is only required to support IPv4 addresses.	M	5
7	…dvbRcsNetworkLanInetAddressPrefix Length	InetAddress PrefixLength	R W	Prefix length for the LAN IP Address of the terminal. If this address prefix is unknown or does not apply, the value is zero.	M	5

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
8	…dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressType	InetAddressType ipv4(1)	R W	The type of Internet address of *dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress*. If the default gateway Internet address is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	EXTNETWORK ipv4(1)	5
9	…dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress	InetAddress SIZE(4)	R W	IP address of the default gateway for the air interface. If the terminal has no default gateway assigned on the air interface or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressTyp e* object. An implementation is only required to support IPv4 addresses.	EXTNETWORK	5
10	…dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressPrefixLength	InetAddress PrefixLength	R W	Prefix length for the IP address of the default gateway for the air interface. If this address prefix is unknown or does not apply, the value is zero.	EXTNETWORK	5
11	…dvbRcsNetworkDnsServers	OBJECT IDENTIFIER	N	IP addresses of the DNS servers in the NCC.	DNS	5.4

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
11.1	…dvbRcsNetworkDnsServers.dvbRcsPrimaryDnsServerInetAddressType	InetAddress Type ipv4(1)	R W	The type of Internet address of *dvbRcsPrimaryDnsServerInetAddress*. If the primary DNS server Internet address is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	DNS	5.4
11.2	…dvbRcsNetworkDnsServers.dvbRcsPrimaryDnsServerInetAddress	InetAddress SIZE(4)	R W	IP address of the primary DNS server in the NCC. If the terminal has no primary DNS server assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsPrimaryDnsServerInetAddressType* object. An implementation is only required to support IPv4 addresses.	DNS	5.4
11.3	…dvbRcsNetworkDnsServers.dvbRcsPrimaryDnsServerInetAddressPrefixLength	InetAddress PrefixLength	R W	Prefix length for the IP address of the primary DNS server in the NCC. If this address prefix is unknown or does not apply, the value is zero.	DNS	5.4

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
11.4	…dvbRcsNetworkDnsServers.dvbRcsSecondaryDnsServerInetAddressType	InetAddressType ipv4(1)	R W	The type of Internet address of *dvbRcsSecondaryDnsServerInetAddress*. If the secondary DNS server Internet address is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	DNS ipv4(1)	5.4
11.5	…dvbRcsNetworkDnsServers.dvbRcsSecondaryDnsServerInetAddress	InetAddress SIZE(4)	R W	IP address of the secondary DNS server in the NCC. If the terminal has no secondary DNS server assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsSecondaryDnsServerInetAddressType* object. An implementation is only required to support IPv4 addresses.	DNS	5.4
11.6	…dvbRcsNetworkDnsServers.dvbRcsSecondaryDnsServerInetAddressPrefixLength	InetAddress PrefixLength	R W	Prefix length for the IP address of the secondary DNS server in the NCC. If this address prefix is unknown or does not apply, the value is zero.	DNS	5.4

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
12	…dvbRcsNetworkNccMgtInetAddressType	InetAddressType ipv4(1)	R W	The type of Internet address of dvbRcsNetworkNccMgtInetAddress. If the management server Internet address is unassigned or unknown, then the value of this object is unknown(0). An implementation is only required to support IPv4 addresses.	EXTNETWORK ipv4(1)	5
13	… dvbRcsNetworkNccMgtInetAddress	InetAddress SIZE(4)	R W	IP address of the management server in the NCC. If the terminal has no management server assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsNetworkNccMgtInetAddressType* object. An implementation is only required to support IPv4 addresses.	EXTNETWORK	5
14	dvbRcsNetworkNccMgtInetAddressPrefixLength	InetAddress PrefixLength	R W	Prefix length for the IP address of the management server in the NCC. If this address prefix is unknown or does not apply, the value is zero.	EXTNETWORK	5

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
15	…dvbRcsNetworkConfigFileDownloadUrl	Uri (SIZE(0..65535))	R W	Full path name for the configuration file download It includes the protocol type (TFTP or FTP) and the associated server IP address or hostname. Hostname can only be used if DNS is supported by the RCST. The format of this parameter follows RFC 3986.	M	7.3
16	…dvbRcsNetworkInstallLogFileDownloadUrl	Uri (SIZE(0..65535))	R W	Full path of the installation log file to download. It includes the protocol type (TFTP or FTP) and the associated server IP address or hostname. Hostname can only be used if DNS is supported by the RCST. The installation log file can be created on the installer's computer and downloaded to the RCST. The format of this parameter follows RFC 3986.	INSTALL_LOG	7.2
17	…dvbRcsNetworkConfigFileUploadUrl	Uri (SIZE(0..65535))	R W	Full path name for the configuration file upload. It includes the protocol type (TFTP or FTP) and the associated server IP address or hostname. Hostname can only be used if DNS is supported by the RCST. The format of this parameter follows RFC 3986.	M	7.6

OID	Name dvbRcsRcstNetwork…	Syntax	Access	Description	Satlabs Applicability	Section
18	…dvbRcsNetworkLogFileUploadUrl	Uri (SIZE(0..65535))	R W	Full path of the event log file. It includes the protocol type (TFTP or FTP) and the associated server IP address or hostname. Hostname can only be used if DNS is supported by the RCST. The format of this parameter follows RFC 3986.	M	9.2
13	…dvbRcsNetworkInstallLogFileUpload Url	Uri (SIZE(0..65535))	R W	Full path of the installation log file. It includes the protocol type (TFTP or FTP) and the associated server IP address or hostname. Hostname can only be used if DNS is supported by the RCST. The installation log file can be retrieved from the RCST by the NCC or by the installer via the LAN. The format of this parameter follows RFC 3986.	INSTALL_LOG	7.2

17.2.3.1.3 dvbRcsRcstInstall subgroup (3)

This subgroup contains all the information related to the RCST installation and commissioning. Many parameters are believed to stay unchanged once it has been defined during installation. Modification of hardware equipment, maintenance operations and geographical re-location may require an update of those MIB objects. They are defined in Table 17-7.

Table 17-7 - *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstInstall* sub-group definition

OID	Name **dvbRcsRcstInstall…**	Syntax	Access	Description	Satlabs Applicability	Section
1	…dvbRcsInstallAntennaAlignmentState	INTEGER	R W	Indicates the alignment state of the antenna: (1) antennaAlignmentStart (2) antennaAlignmentDeny (3) antennaAlignmentContinue (4) antennaAlignmentStop (5) antennaAlignmentSuccess (6) antennaAlignmentFail	M	7.2
2	…dvbRcsInstallCwFrequency	Unsigned32 UNITS “x100 Hz”	R W	Frequency of the transmitted Continuous Wave carrier (in 100 Hz). Minimum required precision is 1 kHz.	M	7.2
3	…dvbRcsInstallCwMaxDuration	Unsigned32 UNITS “seconds”	R W	Time after which the Continuous Wave carrier must be put down (in seconds).	M	7.2
4	…dvbRcsInstallCwPower	Integer32 UNITS “x0.1 dBm”	R W	IDU TX output level when the IDU is configured to send CW. The resolution is 0.1 dBm and the accuracy is ±1 dBm. Reconfiguration is applied immediately to a CW.	M	7.2
5	…dvbRcsInstallCoPolReading	Unsigned32 UNITS “x0.1 dB”	R W	Co-polarisation measured value during installation procedure (in 0.1 dB).	M	7.2

OID	Name dvbRcsRcstInstall…	Syntax	Access	Description	Satlabs Applicability	Section
6	…dvbRcsInstallXPolReading	Unsigned32 UNITS “x0.1 dB”	R W	Cross-polarisation measured value during installation procedure (in 0.1 dB).	M	7.2
7	…dvbRcsInstallCoPolTarget	Unsigned32 UNITS “x0.1 dB”	R W	Co-polarisation target value during installation procedure (in 0.1 dB)	M	7.2
8	…dvbRcsInstallXPolTarget	Unsigned32 UNITS “x0.1 dB”	R W	Cross-polarisation target value during installation procedure (in 0.1 dB)	M	7.2
9	…dvbRcsInstallStandByDuration	Unsigned32 UNITS “seconds”	R W	Time to wait in stand-by mode (in seconds).	M	7.2
10	... dvbRcsInstallTargetEsN0	Unsigned32 (0..315) UNITS “x0.1 dB” DEFVAL = 70	R W	This value describes the wanted Es/N0 value that enables operation of the return link with the required error performance. The values shall be given in tenth of dB and the initial value shall be equal to 7 dB. The range shall be from 0 dB to 31.5 dB, with a precision of 0.1 dB.	M	7.2

17.2.3.1.4 dvbRcsRcstQos subgroup (4)

This subgroup contains objects to configure the QoS of the RCST by the NCC.

The *dvbRcsPktClass* table defines the packet classification for IP layer 3 classifications. Some objects part of the IP layer 3 packet classification and layer 2 Ethernet classification (e.g. MAC address and VLAN id) are not required for short term QoS support. Each *dvbRcsPktClass* entry is mapped to a *dvbRcsPhbEntry* in *dvbRcsPhbMappingTable*.

The *dvbRcsPhbMappingTable* makes the relation between a packet classification entry, a PHB identifier and a Request class entry.

The *dvbRcsRequestClassTable* defines all the layer 2 DVB-RCS QoS parameters.

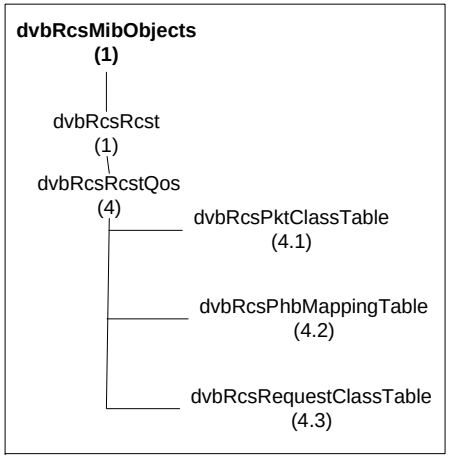

Figure 17-5 - **dvbRcsRcstQos** sub-tree structure

Table 17-8 – *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstQos* subgroup

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
1	…dvbRcsPktClassTable	SEQUENCE OF DvbRcsPktClassEntry	N	This table describes the packet classification used in the DVB-RCS terminal. The number of entries is specified by *dvbRcsPktClassIndex*.	M	See [7]
1.1	…dvbRcsPktClassEntry	dvbRcsPktClassEntry SEQUENCE	N	An entry in the packet classification table. One object type of each entry may have a value in the active range (a non-default value). The other object types are then assumed set to 'inactive'. The entry with the lowest index value takes precedence when classifying a packet.	M	See [7]
1.1.1	…dvbRcsPktClassIndex	Unsigned32 (1..64) INDEX	N	Index of the packet classification table.	M	See [7]
1.1.2	…dvbRcsPktClassDscpLow	Dscp DEFVAL = 0	R R C	This object specifies the low value of a range of DSCP values to which a packet is compared. A value of 0 is used to inactivate.	M (read) ENHCLASSIFIER (create)	See [7]
1.1.3	…dvbRcsPktClassDscpHigh	Dscp DEFVAL = 63	R R C	This object specifies the high value of a range of DSCP values to which a packet is compared. A value of 63 is used to inactivate.	M (read) ENHCLASSIFIER (create)	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
1.1.4	…dvbRcsPktClassDscpMarkValue	DscpOrAny DEFVAL = -1	R R C	This object is the DSCP value used to mark the packet. -1 indicates no DSCP marking. Possible DSCP marks values are (0..63).	M (read) ENHCLASSIFIER (create)	See [7]
1.1.5	...dvbRcsPktClassIpProtocol	Unsigned32 (0..255)	R C	This object specifies the IP protocol to which a packet is compared. A value of 255 means match all.	ENHCLASSIFIER	See [7]
1.1.6	…dvbRcsPktClassSrcInetAddressType	InetAddressType ipv4(1)	R C	The type of internet Internet address of dvbRcsPktClassSrcInetAddress. If the packet class source Internet address is unassigned or unknown, then the value of this object is unknown(0).	ENHCLASSIFIER	See [7]
1.1.7	…dvbRcsPktClassSrcInetAddress	InetAddress SIZE(4)	R C	This object specifies the IP source address to which a packet is compared. If the packet class has no source Internet address assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsPktClassSrcInetAddressType* object. An implementation is only required to support IPv4 addresses.	ENHCLASSIFIER	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
1.1.8	…dvbRcsPktClassSrcInetAddressPrefixLength	InetAddressPrefixLength DEFVAL = 0	R C	Prefix length of the IP source address that will be matched for this packet class. A value of zero indicates that the selectivity is inactive.	ENHCLASSIFIER	See [7]
1.1.9	…dvbRcsPktClassDstInetAddressType	InetAddressType ipv4(1)	R C	The type of Internet address of dvbRcsPktClassDstInetAddress. If the packet class destination Internet address is unassigned or unknown, then the value of this object is unknown(0).	ENHCLASSIFIER	See [7]
1.1.10	…dvbRcsPktClassDstInetAddress	InetAddress SIZE(4)	R C	This object specifies the IP destination address to which a packet is compared. If the packet class has no destination Internet address assigned or if this Internet address is unknown, the value of this object is the zero-length OCTET STRING. The *InetAddressType* is given by the *dvbRcsPktClassDstInetAddressType* object. An implementation is only required to support IPv4 addresses.	ENHCLASSIFIER	See [7]
1.1.11	…dvbRcsPktClassDstInetAddressPrefixLength	InetAddressPrefixLength DEFVAL = 0	R C	The type of Internet address of *dvbRcsPktClassDstInetAddress*. If the packet class destination Internet address is unassigned or unknown, then the value of this object is unknown(0).	ENHCLASSIFIER	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
1.1.12	…dvbRcsPktClassSrcPortLow	InetPortNumber DEFVAL = 0	R C	This object specifies the low range of the source port to which a packet is compared. A value of 0 indicates that the selectivity is inactive.	ENHCLASSIFIER	See [7]
1.1.13	…dvbRcsPktClassSrcPortHigh	InetPortNumber DEFVAL = 65535	R C	This object specifies the high range of the source port to which a packet is compared. A value of 0 indicates that the selectivity is inactive.	ENHCLASSIFIER	See [7]
1.1.14	…dvbRcsPktClassDstPortLow	InetPortNumber DEFVAL = 0	R C	This object specifies the low range of the destination port to which a packet is compared. A value of 0 indicates that the selectivity is inactive.	ENHCLASSIFIER	See [7]
1.1.15	…dvbRcsPktClassDstPortHigh	InetPortNumber DEFVAL = 65535	R C	This object specifies the high range of the destination port to which a packet is compared. A value of 0 indicates that the selectivity is inactive.	ENHCLASSIFIER	See [7]
1.1.16	…dvbRcsPktClassVlanUserPri	Integer32 (-1..7) DEFVAL = -1	R C	This object specifies the VLAN User Priority to which a packet is compared. A value of -1 indicates that the selectivity is inactive.	ENHCLASSIFIER Not tested	See [7]
1.1.17	…dvbRcsPktClassPhbAssociation	Unsigned32 (0..65535)	R C	Associate the filter entry to a specific PHB (refer to *dvbRcsPhbIdentifier*).	M	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
1.1.18	…dvbRcsPktClassRowStatus	RowStatus DEFVAL = active	R C	Standard SNMP row status: (1) active (2) notInService (3) notReady (4) createAndGo (5) createAndWait (6) destroy All writable objects in this row may be modified at any time.	M	See [7]
2	…dvbRcsPhbMappingTable	SEQUENCE OF DvbRcsPhbMappingEntry	N	This table is a list of PHB MIB entries. This class describes the PHB mapping with the Request Class.	M	See [7]
2.1	…dvbRcsPhbMappingEntry	DvbRcsPhbMappingEntry SEQUENCE	N	An entry in the PHB mapping table.	M	See [7]
2.1.1	…dvbRcsPhbIdentifier	Unsigned32 (0..65535) INDEX	N	Identification of the Per-Hop Behavior (PHB). It follows the unsigned 16-bit binary encoding as specified in RFC 3140 [24]. The value 0 designates the Default PHB.	M	See [7]
2.1.2	…dvbRcsPhbName	SnmpAdminString	R R C	Name of the PHB.	M (read) SNMPMISC (create)	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
2.1.3	…dvbRcsPhbRequestClassAssociation	Unsigned32 (1..16)	R R C	This object is an association of this PHB to a Request Class (by reference to a Request Class index).	M (read) SNMPMISC (create)	See [7]
2.1.4	…dvbRcsPhbMappingRowStatus	RowStatus DEFVAL = active	R R C	The status of this conceptual row. All writable objects in this row may be modified at any time.	M (read) SNMPMISC (create)	See [7]
3	…dvbRcsRequestClassTable	SEQUENCE OF DvbRcsRequestClassEntry	N	This table is a list of Request Class entries. This class describes the layer 2 QoS objects.	M	See [7]
3.1	…dvbRcsRequestClassEntry	DvbRcsRequestClassEntry SEQUENCE	N	An entry in the Request class table.	M	See [7]
3.1.1	…dvbRcsRequestClassIndex	Unsigned32 (1..16) INDEX	N	Index of the Request Class table. A total of 16 entries are supported.	M	See [7]
3.1.2	…dvbRcsRequestClassName	SnmpAdminString	R R C	Name of the Request class.	M (read) SNMPMISC (create)	See [7]
3.1.3	…dvbRcsRequestClassChanId	Unsigned32 (0..15)	R R C	Channel ID of the Request class.	M (read) SNMPMISC (create)	See [7]
3.1.4	…dvbRcsRequestClassVccVpi	Unsigned32 (0..255)	R R C	Defines VPI used for the Request class (ATM profile).	M (read) SNMPMISC (create)	See [7]
3.1.5	…dvbRcsRequestClassVccVci	Unsigned32 (0..65535)	R R C	Defines VCI used for the Request class (ATM profile).	M (read) SNMPMISC (create)	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
3.1.6	…dvbRcsRequestClassPidPoolReference	Unsigned32 (1..16)	R C	Reference to the PID pool applicable for the Request Class.	MPEG_TRF SNMPMISC (create)	See [7]
3.1.7	…dvbRcsRequestClassCra	Unsigned32 UNITS “bit/s”	R R C	Defines the CRA level for the Request class in bits per second (bit/s).	M (read) SNMPMISC (create)	See [7]
3.1.8	…dvbRcsRequestClassRbdcMax	Unsigned32 UNITS “x2 kbit/s”	R R C	Maximum RBDC that can be requested for the Request class, in number of 2 kbit/s.	M (read) SNMPMISC (create)	See [7]
3.1.9	…dvbRcsRequestClassRbdcTimeout	Unsigned32 UNITS “superframes”	R R C	Persistence of the RBDC request, expressed in superframes.	M (read) SNMPMISC (create)	See [7]
3.1.10	…dvbRcsRequestClassVbdcMax	Unsigned32 UNITS “ATM cells /MPEG packets”	R R C	Maximum VBDC that can be allocated to the Request class, in payload units (one ATM cell or one MPEG packet) per superframe.	M (read) SNMPMISC (create)	See [7]
3.1.11	…dvbRcsRequestClassVbdcTimeout	Unsigned32 UNITS “superframes”	R R C	Time after which the RCST considers that the pending requests are lost. The RCST may issue new requests for that traffic. VBDC Timeout is expressed in superframes.	M (read) SNMPMISC (create)	See [7]
3.1.12	…dvbRcsRequestClassVbdcMaxBackLog	Unsigned32 UNITS “bytes”	R R C	VBDC back log per Request class (expressed in bytes).	M (read) SNMPMISC (create)	See [7]
3.1.13	…dvbRcsRequestClassRowStatus	RowStatus DEFVAL = active	R R C	The status of this conceptual row. It is not possible to change values in a row of this table while the row is active.	M (read) SNMPMISC (create)	See [7]

OID	Name dvbRcsRcstQos…	Syntax	Access	Description	Satlabs Applicability	Section
4	…dvbRcsPidPoolTable	SEQUENCE OF DvbRcsPidPoolEntry	N	This table contains the PID pools. For the MPEG profile several Request Classes may be mapped within a pool of several PIDs to allow Section Packing across several Request Classes. A PID value may occur in more than one PID pool. Each PID value can effectively occur only once in each pool. Only one PID per PID pool is required for short term QoS support.	MPEG_TRF	See [7]
4.1	…dvbRcsPidPoolEntry	DvbRcsPidPoolEntry SEQUENCE	N	An entry in the PID pool table.	MPEG _TRF	See [7]
4.1.1	…dvbRcsPidPoolIndex	Unsigned32 (1..16) INDEX	N	Index of the PID pool in the PID pool table.	MPEG_TRF	See [7]
4.1.2	…dvbRcsPidIndex	Unsigned32 (1..16) INDEX	N	Index of the PID entry within the PID pool.	MPEG_TRF	See [7]
4.1.3	…dvbRcsPidValue	Unsigned32 (0..8191)	R C	Define one of the PIDs to be used in a PID pool of *dvbRcsPidPoolIndex*.	MPEG_TRF SNMPMISC (create)	See [7]
4.1.4	…dvbRcsPidPoolRowStatus	RowStatus DEFVAL = active	R C	The status of this conceptual row. All writable objects in this row may be modified at any time.	MPEG_TRF SNMPMISC (create)	See [7]
5	…dvbRcsQosGlobalRbdcMax	Unsigned32 UNITS “x2 kbit/s”	R W	Global maximum RBDC that can be requested for the RCST, in number of 2 kbitss.	RCST_PARA	See [7]

OID	Name **dvbRcsRcstQos…**	Syntax	Access	Description	Satlabs Applicability	Section
6	…dvbRcsQosGlobalVbdcMax	Unsigned32 UNITS "ATM cells /MPEG packets"	R W	Global maximum VBDC that can be allocated to the RCST, in payload units (one ATM cell or one MPEG packet) per superframe.	RCST_PARA	See [7]
7	…dvbRcsQosGlobalVbdcMaxBackLog	Unsigned32 UNITS "bytes"	R W	Global VBDC back log at the RCST level (expressed in bytes). It is used only if the VBDC back log is not configured in the Request Class (expressed in bytes).	RCST_PARA	See [7]
8	…dvbRcsQosChannelIdStrictDispatching	INTEGER	R W	Indicates whether the RCST will strictly follow RC association when signaled through Channel_ID in the TBTP: (0) notStrict = No strict association (1) strict = Strict association	CHID_STRICT	See [7]

17.2.3.1.5 dvbRcsRcstControl group (5)

This MIB group contains objects a network manager can use to invoke actions and tests supported by the RCST agent and to retrieve the action/test results.

Table 17-9 – *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstControl* subgroup

OID	Name **dvbRcsRcstControl…**	Syntax	Access	Description	Satlabs **Applicability**	Section
1	...dvbRcsCtrlRebootCommand	INTEGER DEFVAL = 1	R W	This variable shall force the RCST to reboot: (1) idle (2) normal = normal reboot (from current software load) (3) alternate = reboot from alternate load (swap to alternate load before reboot)	M	6.2
2	…dvbRcsCtrlRcstTxDisable	INTEGER DEFVAL = 1	R W	This variable shall force the RCST to stop transmission (transmit disabled as defined in [1]): (1) idle (2) disable = initiate Tx Disabled	EXTCONTROL	6.2
3	…dvbRcsCtrlUserTrafficDisable	INTEGER DEFVAL = 1	R W	This variable shall disable user traffic (only RCST management traffic can be transmitted): (1) idle (2) disable = disable user traffic	M	6.2

OID	Name **dvbRcsRcstControl…**	Syntax	Access	Description	Satlabs **Applicability**	Section
4	…dvbRcsCtrlCwEnable	INTEGER DEFVAL = 1	R W	This variable will force the RCST to start transmission of CW, if the RCST is first set to the installation state and is properly configured for CW transmission: (1) off (2) on	M	7.2
5	…dvbRcsCtrlOduTxReferenceEnable	INTEGER DEFVAL = 2	R W	Enables activation and deactivation of 10 MHz reference clock in the Tx IFL cable: (1) off (2) on	EXTCONTROL	7.2
6	…dvbRcsCtrlOduTxDCEnable	INTEGER DEFVAL = 2	R W	Enables activation and deactivation of DC in the Tx IFL cable: (1) off (2) on	EXTCONTROL	7.2
7	…dvbRcsCtrlOduRxDCEnable	INTEGER DEFVAL = 2	R W	Enables activation and deactivation of DC in the Rx IFL cable: (1) off (2) on	EXTCONTROL	7.2

OID	Name dvbRcsRcstControl…	Syntax	Access	Description	Satlabs Applicability	Section
8	...dvbRcsCtrlDownloadFileCommand	INTEGER DEFVAL = 1	R W	This variable will initiate an RCST configuration file download process: (1) idle (2) config = download RCST configuration file from TFTP/FTP server (3) installationLog = Download RCST installation log file from TFTP/FTP server.	M (3) for INSTALL_LOG	6.2, 7.3
9	...dvbRcsCtrlUploadFileCommand	INTEGER DEFVAL = 1	R W	This variable will initiate an RCST upload process: (1) idle (2) config = upload RCST configuration file to TFTP/FTP server (3) eventAlarm = upload RCST event/alarm log file to TFTP/FTP server (4) installationLog = upload RCST installation log file to TFTP/FTP server.	M (4) for INSTALL_LOG	7.6
10	…dvbRcsCtrlActivateConfigFileCommand	INTEGER DEFVAL = 1	R W	Triggers the RCST to use the configuration file and update its parameters accordingly. Some RCST implementations may require a reboot for the parameters to take effect (vendor specific). (1) idle (2) activate	M	7.3

OID	Name dvbRcsRcstControl…	Syntax	Access	Description	Satlabs Applicability	Section
11	...dvbRcsCtrlRcstLogonCommand	INTEGER DEFVAL = 1	R W	This variable will initiate an RCST logon: (1) idle (2) logon = initiate RCST logon	EXTCONTROL	6.2
12	...dvbRcsCtrlRcstLogoffCommand	INTEGER DEFVAL = 1	R W	This variable will initiate an RCST logoff: (1) idle (2) logon = initiate RCST logoff	EXTCONTROL	6.2
13	…dvbRcsCtrlRcstRxReacquire	INTEGER DEFVAL = 1	R W	This variable will force the RCST to acquire the forward link and start receiving: (1) idle (2) reacquireForwardLink	M	

17.2.3.1.6 dvbRcsRcstState group (6)

This group describes the fault state, software versions and configuration file versions.

Table 17-10 – *dvbRcsMibObjects.dvbRcsRcst.dvbRcsRcstState* group definition

OID	Name dvbRcsRcstState…	Syntax	Access	Description	Satlabs Applicability	Section
1	…dvbRcsRcstMode	INTEGER	R W	Identifies the current mode of the RCST and allows the RCST to the installation mode when needed. Values for the RCST mode are: (0) installation (1) operational	M	6.1, 7.2
2	...dvbRcsRcstFaultStatus	INTEGER	R	Provides the fault status of the terminal. The fault status management is vendor specific. Values for the fault status are: (0) nofault (1) fault	M	6.1
3	…dvbRcsRcstFwdLinkStatus	INTEGER	R	Provides the status of the RCST forward link. Values for the forward link status are: (0) notAcquired (1) acquired	M	6.1

OID	Name dvbRcsRcstState…	Syntax	Access	Description	Satlabs Applicability	Section
4	…dvbRcsRcstRtnLinkStatus	INTEGER	R	Provides the status of the RCST return link. Values for the return link status are: (0) loggedOff (1) loggedOn	EXTSTATUS	6.1
5	…dvbRcsRcstLogUpdated	INTEGER	R	Indicates the existence of an updated event log file: (0) noUpdate (1) logFileUpdated The RCST should remove the 'Event log file updated' indication as the log file is fetched by the NCC.	M	9.2
6	...dvbRcsRcstCurrentSoftwareVersion	SnmpAdminString	R	Current RCST software version.	M	7.4
7	...dvbRcsRcstAlternateSoftwareVersion	SnmpAdminString	R	Alternate (backup/new) RCST software version.	M	7.4
8	…dvbRcsRcstActivatedConfigFileVersion	SnmpAdminString	R	Version of the most recently activated configuration file. Version is vendor specific	M	7.3
9	…dvbRcsRcstDownloadedConfigFileVersion	SnmpAdminString	R	Version of the most recently downloaded configuration file. Version is vendor specific. If the value is different from *dvbRcsRcstActivatedConfigFileVersion*, it is pending for activation.	M	7.3

17.2.3.2 dvbRcsFwdLink subgroup (2)

This subgroup contains parameters that enable the NCC to have access to data about the forward path.

Configuration information is kept into the *dvbRcsFwdLink.dvbRcsFwdConfig* subgroup. Status information is kept into the *dvbRcsFwdLink.dvbRcsFwdStatus* subgroup.

The information of the *dvbRcsFwdLink.dvbRcsFwdConfig.dvbRcsFwdStartTable* table are used for the first time the RCST tries to acquire the FL. All these objects values are aligned with the Satellite Delivery System Descriptor in the NIT table [3].

The objects in the *dvbRcsFwdLink.dvbRcsFwdConfig.dvbRcsFwdStatusTable* table are aligned with the satellite forward path descriptor form the RMT [1] and with the Physical Layer (PL) Header [5], which specified the MODCOD (modulation and FEC rate) and the Type (frame length short of long and the presence/absence of pilots). The Transmission Mode Support Table (TMST) is optional for DVB-S2 support, thus can not be used to update the *dvbRcsFwdStatusTable* and the TMST does not represent the MODCOD in real-time.

Figure 17-6 illustrates different relations between the *dvbRcsFwdLink* groups with the NIT, RMT, PL Frame, TMST and the DVB-RCS interfaces.

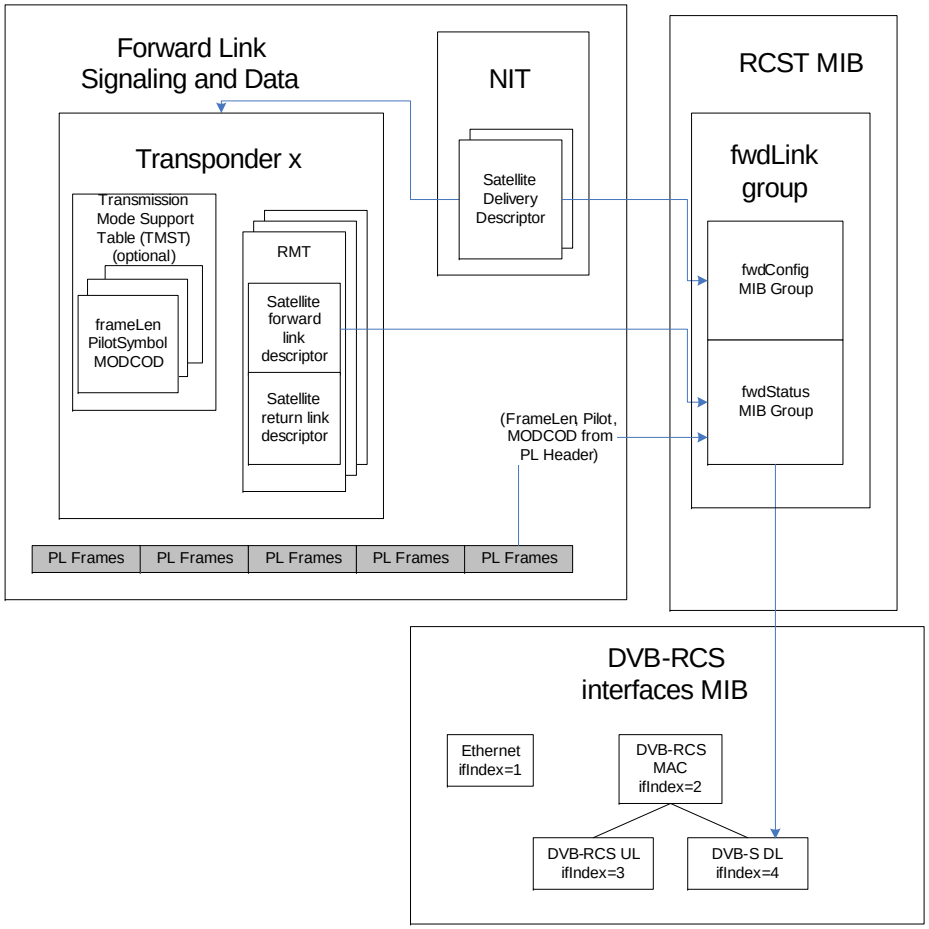

Figure 17-6 – dvbRcsFwdLink group

17.2.3.2.1 dvbRcsFwdConfig subgroup (1)

Table 17-11 – *dvbRcsMibObjects.dvbRcsFwdLink.dvbRcsFwdConfig* subgroup definition

OID	Name **dvbRcsFwdLink.dvbRcsFwdConfig…**	Syntax	Access	Description	Satlabs Applicability	Section
1	…dvbRcsFwdStartTable	SEQUENCE OF dvbRcsFwdStartEntry	N	Lists forward link attachment points (e.g., different for installation and operation). The table describes the forward link parameters used for the start-up stream with the NCC.	M	
1.1	…dvbRcsFwdStartEntry	dvbRcsFwdStartEntry SEQUENCE	N	An entry in the FL Start Configuration table.	M	
1.1.1	…dvbRcsFwdStartIndex	Unsigned32 (1..8) INDEX	N	Index of the FL Start Configuration table. An RCST SHALL have at least one entry.	M	
1.1.2	...dvbRcsFwdStartPopId	Integer32 (-1..65535)	R C	Population identifier associated with the start-up forward link: (-1) any (auto) (0-n) specific StartPopId If ‘any’ is set, the RCST will assume membership of any announced population ID and will commence with logon in accordance with this assumption.	M	7.2
1.1.3	... dvbRcsFwdStartFrequency	Unsigned32	R C	Frequency of the start transponder carrying a Network	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdConfig…	Syntax	Access	Description	Satlabs Applicability	Section
		UNITS "x100 kHz"		Information Table to which any RCST shall trigger to acquire forward link. Its value shall be given in multiples of 100 kHz.		
1.1.4	…dvbRcsFwdStartPolar	INTEGER	R C	2-bit field giving the polarization of the start transponder carrying a Network Information Table to which any RCST shall trigger to acquire forward link: (0) linearHorizontal (1) linearVertical (2) circularLeft (3) circularRight	M	7.2
1.1.5	...dvbRcsFwdStartFormat	INTEGER	R C	Specifies the transmission format standard applied for the startup stream. The startup stream carries a Network Information Table that the RCST uses for acquiring the forward link signaling. Supported values are: (-1) auto = unspecified (automatic format acquisition is assumed) (0) dvbs = DVB-S (support of this value is mandatory if DVB-S support is claimed) (1) dvbs2ccm = DVB-S2 with CCM support (support of this value is mandatory if DVB-S2 CCM	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdConfig…	Syntax	Access	Description	Satlabs Applicability	Section
				support is claimed) (2) dvbs2acm = DVB-S2 with VCM or ACM support (support of this value is mandatory if DVB-S2 ACM support is claimed) This allows the RCST to discriminate between CCM and VCM/ACM when selecting FL. The support of automatic format selection is optional. One or several of the other format selections must be supported, according to the claimed SatLabs profile support.		
1.1.6	…dvbRcsFwdStartRolloff	INTEGER	R C	Specifies the receive filter roll-off applied on the start transponder. The start transponder carries a Network Information Table that the RCST uses for acquiring the forward link signaling. Supported values are: (0) autoRolloff = any (1) rolloff020 = 0.20 (2) rolloff025 = 0.25 (3) rolloff035 = 0.35	M	7.2
1.1.7	...dvbRcsFwdStartSymbolRate	Unsigned32 UNITS “x100 symbols/s”	R C	Specifies the symbol rate on the start transponder carrying a Network Information Table to which any RCST shall trigger to acquire forward link. Its value	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdConfig…	Syntax	Access	Description	Satlabs Applicability	Section
				shall be given in multiples of 100 symbol/s.		
1.1.8	...dvbRcsFwdStartInnerFec	INTEGER	R C	Specifies the inner Forward Error Correction used on the start transponder carrying a NIT to which any RCST shall trigger to acquire forward link. Supported values are: (-1) autoFec (0) fecRate12 = fecRate 1/2 (1) fecRate23 = fecRate 2/3 (2) fecRate34 = fecRate 3/4 (3) fecRate56 = fecRate 5/6 (4) fecRate78 = fecRate 7/8 (5) fecRate89 = fecRate 8/9 (6) fecRate35 = fecRate 3/5 (7) fecRate45 =fecRate 4/5 (8) fecRate910 = fecRate 9/10 (9) fecRate25 = fecRate 2/5 (10) fecRate13 = fecRate 1/3 (11) fecRate14 = fecRate 1/4 (12) noInnerCode The support of autoFec is optional.	M	7.2
1.1.9	…dvbRcsFwdStartRowStatus	RowStatus	R C	The status of this conceptual row. It is not possible to	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdConfig…	Syntax	Access	Description	Satlabs Applicability	Section
				change values in a row of this table while the row is active.		

17.2.3.2.2 dvbRcsFwdStatus subgroup (2)

Table 17-12 - *dvbRcsMibObjects.dvbRcsFwdLink.dvbRcsFwdStatus* subgroup definition

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
1	…dvbRcsFwdStatusPopId	Unsigned32 (0..65535)	R	Population identifier applied at log-on: 0-65535: specific StartPopId If the RCST was allowed to logon with any population, the RCST will report the base number of the announced population ID indicated by the RMT linkage descriptor used at logon.	M	7.2
2	…dvbRcsFwdStatusTable	SEQUENCE OF DvbRcsFwdStatusEntry	N	This table describes the current status of FL interfaces.	M	
2.1	…dvbRcsFwdStatusEntry	DvbRcsFwdStatusEntry SEQUENCE	N	An entry in the FL status table. Each entry is associated with a physical interface.	M	

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
				An RCST SHALL support at least one entry.		
2.1.1	…dvbRcsFwdStatusIndex	Unsigned32 (1..8) INDEX	N	Index of the FL status table.	M	
2.1.2	…dvbRcsFwdStatusIfReference	Unsigned32 (1..8)	R	Cross reference to the interface table.	M	17.2.4.3.2
2.1.3	... dvbRcsFwdStatusNetId	Unsigned32	R	Interactive network identifier of the forward link (from RMT).	M	7.2
2.1.4	...dvbRcsFwdStatusNetName	SnmpAdminString	R	The name of the interactive network of the forward link (from RMT).	M	
2.1.5	...dvbRcsFwdStatusFormat	INTEGER	R	Specifies the transmission format applied on the forward link. Supported values are (from RMT): (0) dvbs = DVB-S (1) dvbs2ccm = DVB-S2 using CCM (2) dvbs2acm = DVB-S2 using VCM or ACM (3) reservedFormat	M	7.2
2.1.6	...dvbRcsFwdStatusFrequency	Unsigned32 UNITS “x100 kHz”	R	An estimate of the frequency of the forward link. Its value shall be given in multiples of 100 kHz.	M	7.2
2.1.7	…dvbRcsFwdStatusPolar	INTEGER	R	2-bit field giving the polarization of the forward link (from RMT): (0) linearHorizontal	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
				(1) linearVertical (2) circularLeft (3) circularRight		
2.1.8	...dvbRcsFwdStatusInnerFec	INTEGER	R	Specifies the inner Forward Error Correction of the forward link for transmission to the RCST. Supported values are: (-1) unknown (0) fecRate12 = fecRate 1/2 (1) fecRate23 = fecRate 2/3 (2) fecRate34 = fecRate 3/4 (3) fecRate56 = fecRate 5/6 (4) fecRate78 = fecRate 7/8 (5) fecRate89 = fecRate 8/9 (6) fecRate35 = fecRate 3/5 (7) fecRate45 = fecRate 4/5 (8) fecRate910 = fecRate 9/10 (9) fecRate25 = fecRate 2/5 (10) fecRate13 = fecRate 1/3 (11) fecRate14 = fecRate 1/4 (12) noInnerCode	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
				The RCST will report a value that has been used for transmission to the RCST within the most recent 60 seconds. If this is not relevant, the RCST will report 'unknown'.		
2.1.9	...dvbRcsFwdStatusSymbolRate	Unsigned32 UNITS “x100 symbol/s”	R	An estimate of the symbol rate of the forward link. Its value shall be given in multiple of 100 symbols/s.	M	7.2
2.1.10	…dvbRcsFwdStatusRolloff	INTEGER	R	An estimate of the roll-off applied on the forward link. Supported values are: (0) undefRolloff = undefined (1) rolloff020 = 0.20 (2) rolloff025 = 0.25 (3) rolloff035 = 0.35	M	7.2
2.1.11	…dvbRcsFwdStatusModulation	INTEGER	R	Indicates the modulation on the forward link used for transmission to the RCST: (0) unknown (1) mBPSK = BPSK (2) mQPSK = QPSK (3) m8PSK = 8PSK (4) m16APSK = 16APSK (5) m32APSK = 32APSK	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
				The RCST will report a value that has been used for transmission to the RCST within the most recent 60 seconds. If this is not relevant, the RCST will report 'unknown'.		
2.1.12	…dvbRcsFwdStatusFecFrame	INTEGER	R	Indicates the frame length used on the forward link for transmission to the RCST. Supported values are: (0) unknown (1) shortframe = Short frame (2) longframe = Normal frame The RCST will report a value that has been used for transmission to the RCST within the most recent 60 seconds. If this is not relevant, the RCST will report 'unknown'.	M	7.2
2.1.13	…dvbRcsFwdStatusPilot	INTEGER	R	Indicates whether pilots are used on the forward link for transmission to the RCST. Supported values are: (0) unknown (1) pilotNotused = Pilots are not used (2) pilotUsed = Pilots are used The RCST will report a value that has been used for transmission to the RCST within the most recent 60	M	7.2

OID	Name dvbRcsFwdLink.dvbRcsFwdStatus…	Syntax	Access	Description	Satlabs applicability	Section
				seconds. If this is not relevant, the RCST will report 'unknown'.		
2.1.14	…dvbRcsFwdStatusBer	Integer32 UNITS “exponent of 10”	R	Provides the RCST BER on the forward link in $\log_{10}$ units.	M	7.2
2.1.15	…dvbRcsFwdStatusCnr	Integer32 UNITS “x0.1 dB”	R	Provides the RCST CNR on the forward link in 0.1 dB units.	M	7.2
2.1.16	…dvbRcsFwdStatusRxPower	Integer32 UNITS “x0.1 dBm” DEFVAL = -500	R	Provides the power level of the forward link as received at the IDU, in 0.1 dBm units.	M	7.2

17.2.3.3 dvbRcsRtnLink subgroup (3)

This subgroup contains parameters that enable the NCC to have access to data about the return path.

Up to now, the RCST is only able to deal with one return link at a time. Hence, there is no need to define a table to collect the different SNMP objects, as it is done for the forward.

17.2.3.3.1 dvbRcsRtnConfig subgroup (1)

Table 17-13 - *dvbRcsMibObjects.dvbRcsRtnLink.dvbRcsRtnConfig* subgroup definition

OID	Name **dvbRcsRtnLink.dvbRcsRtnConfig…**	Syntax	Access	Description	Satlabs Applicability	Section
1	...dvbRcsRtnConfigMaxEirp	Integer32 UNITS "x0.1 dBm"	R W	Maximum EIRP of the RCST, given in resolution of 0.1 dBm and applied when the IDU can, itself, set the necessary IDU TX output level e.g., when using a BUC that has a power level detector and that provides sufficient feedback to the IDU.	EXTCONFIG	7.2
2	... dvbRcsRtnConfigDefIfLevel	Integer32 UNITS "x0.1 dBm"	R W	IDU TX output level applied in case the *dvbRcsRtnConfigMaxEirp* cannot be used. The resolution is 0.1 dBm and the accuracy is ±1 dBm.	M	7.2

17.2.3.3.2 dvbRcsRtnStatus subgroup (2)

Table 17-14 - *dvbRcsMibObjects.dvbRcsRtnLink.dvbRcsRtnStatus* subgroup definition

OID	Name **dvbRcsRtnLink.dvbRcsRtnStatus…**	Syntax	Access	Description	Satlabs Applicability	Section
1	...dvbRcsRtnStatusEbN0	Integer32 UNITS “x0.1 dB”	R	The EbN0 value reported for the return link, referenced to the regular SYNC burst transmission, in 0.1 dB units.	EXTSTATUS	7.2
2	…dvbRcsRtnStatusSFDuration	Unsigned32 UNITS “x0.1 ms” (250..7500)	R	The duration of the currently applied return link superframe structure, in tenths of milliseconds.	EXTSTATUS	7.2
3	... dvbRcsRtnStatusPayloadUnit	INTEGER	R	Indicates if the payload unit used for the return link is ATM or MPEG. Supported values are: (0) unitATM = ATM payload unit (1) unitMPEG = MPEG payload unit	M	7.2

17.2.4 MIB-II

17.2.4.1 Supported MIB-II Groups

The following MIB-II Groups are applicable to the management of the RCST and shall be supported by the RCST. The supported Objects are specified in the following clauses:

system Group

interfaces Group

17.2.4.2 Objects Not Supported

The MIB-I and MIB-II specifications dictate that for an implementation to claim support for a group, it must support all of the objects in a group. It is certainly permissible for an agent to support only some of the objects in a group, but in that case the vendor cannot claim that the group is supported.

The correct way to handle this situation is for the agent to return the error code *noSuchName* and for the vendor to admit that this particular group is not supported.

17.2.4.3 MIB-II Groups Specifications

Objects of the following MIB-II groups shall be supported as specified.

ID and *Name* identify objects contained in the definition tables. The *ID* specifies the object location in the MIB hierarchy under the group branch; e.g. object *sysContact* in group system has ID 4, and this corresponds to the last digit in the OID string 1.3.6.1.2.1.1.4 specifying the object location.

Individual objects can be accessed in one of the following ways, as defined in the tables' Access column: Read Only (R); Read-Write (RW); Write Only (W); or Not Accessible (N). Write-access in this context is understood to mean from an SNMP Manager or Agent.

17.2.4.3.1 system Group (1)

The system group is under the iso.org.dod.internet.mgnt.mib-2 branch. Some objects of the MIB-II system group shall be supported by the RCST. These objects contain basic system information and are described in Table 17-15.

Table 17-15 - MIB-II *system* Group Objects

OID	Name	Syntax	Access	Description	Satlabs Applicability
1	sysDescr	DisplayString	R	A text description, which should include some very generic information about the device: type of system (RCST), manufacturer, and date.	M
2	sysObjectID	Object Identifier	R	An authoritative identifier assigned to this product by its vendor.	M
3	sysUpTime	TimeTicks	R	The time (in 1/100 th of a second) since the network management portion of the system was last re-initialized.	M
4	sysContact	DisplayString	R W	A person responsible for the node, along with information such as phone number. Defined at installation. In the scope of the RCS system, this object shall provide all necessary information about the local RCST administrator – Super User.	M
5	sysName	DisplayString	R W	An administratively assigned name (usually the TCP/IP domain name). Defined at installation.	M
6	sysLocation	DisplayString	R W	The physical location of the IDU, including street	M

OID	Name	Syntax	Access	Description	Satlabs Applicability
				address and GPS co-ordinates expressed as Longitude, Latitude and Altitude. Co-ordinates are made of 31 characters in the following format: <xxxx.xxx>,<a>,<yyyyy.yyy>,<b>,<zzzz.z>,M where x, y and z represents digits, a=N or S, b=E or W, Reading the digits from left to right: 'x' – 7 latitude digits; x digits 1-2 contain the degrees, x digits 3-7 contain the minutes in decimal; 'y' – 8 longitude digits; y digits 1-3 contain the degrees, y digits 4-8 contain the minutes in decimal; 'z' 5 altitude digits; meters above sea level in decimal; '.' is the decimal point; ',' is the field separator; 'M' is the indicator for altitude meters. This format is a modified subset of the NMEA 0183 format for Global Positioning System Fix Data.	

OID	Name	Syntax	Access	Description	Satlabs Applicability
				Note: The *dvbRcsRcstSystem.dvbRcsSystemLocation* object of the RCST MIB provides physical location of the ODU antenna and is used for synchronization purpose.	

17.2.4.3.2 interfaces Group (2)

The interface group is under the iso.org.dod.internet.mgnt.mib-2 branch. Almost all objects in the MIB-II interfaces group RFC 2863 shall be supported by the RCST. These objects contain configuration, status and performance data for interfaces and are described in Table 17-16.

The interface types supported are ethernetCsmacd, dvbRcsMacLayer, dvbTdm, and dvbRcsTdma.

Table 17-16 - MIB-II *interfaces* Group Objects

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
1	ifNumber	Integer	R	Total number of network interfaces in the system..	M	9.1
2	ifTable	Sequence of ifEntry	N	List of interface entries. The number of entries is specified by *ifNumber*.	M	9.1
2.1	ifEntry	Sequence	N	An interface entry containing objects at the sub-network layer and below for a particular interface.	M	9.1
2.1.1	ifIndex	Integer	R	A unique value for each interface.	M	9.1
2.1.2	ifDescr	DisplayString Size(0..255)	R	A unique description for each interface.	M Description is vendor specific	9.1
2.1.3	ifType	IANAifType	R	The type of interface: *dvbRcsMacLayer* (239), *dvbTdm* (240), *dvbRcsTdma* (241) or *ethernetCsmacd* (6). Optionally the ATM (80) and MPEG (214) interface can be supported as well.	M	9.1

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
2.1.4	ifMtu	Integer	R	The size (in octets) of the largest protocol data unit that can be sent or received on the interface. For Ethernet, *dvbRcsMacLayer*, *dvbTdm* and *dvbRcsTdma* the MTU is 1500.	M	6.1
2.1.5	ifSpeed	Gauge	R	For the *dvbTdm* interface, the Speed value is the raw bandwidth in bits/s of this interface. This is the symbol rate multiplied with the number of bits per symbol for the highest modulation profile supported by the terminal. For the *dvbRcsTdma* (UL) interface, the speed value is the raw bandwidth in bits/s of this interface, regarding the highest modulation profile that is defined for the MF-TDMA super-frame structure and supported by the device. This is the symbol rate multiplied with the number of bits per symbol.	M	6.1
2.1.6	ifPhysAddress	PhysAddress	R	This physical address can be the same value for all DVB-RCS interfaces, as well for the Ethernet interface.	M	6.1
2.1.7	ifAdminStatus	Integer	R W	The state of the interface. Supported values are: Up (1); Down (2); Testing (3)	M	6.1

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
2.1.8	ifOperStatus	Integer	R	The current operational state of the interface. Supported values are: Up (1); Down (2); Testing (3); Unknown (4); Dormant (5); notPresent (6); lowerlayerDown (7);	M	6.1
2.1.9	ifLastChange	TimeTicks	R	The value of *sysUpTime*, in 1/100th of seconds, when the interface entered its current operational state. If the operational state of the interface has not changed since power up, the value is 0.	M	6.1
2.1.10	ifInOctets	Counter	R	The total number of octets received on the interface, including framing octets.	M	9.1
2.1.11	ifInUcastPkts	Counter	R	The number of subnetwork unicast packets delivered to a higher layer protocol.	M	9.1
2.1.13	ifInDiscards	Counter	R	The number of inbound packets discarded although no errors were found. This is due to a lack of buffer memory.	M	6.1

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
2.1.14	ifInErrors	Counter	R	The number of inbound packets discarded because they contain errors.	M	6.1
2.1.15	ifUnknownProtos	Counter	R	The number of inbound packets discarded because of an unknown or unsupported protocol.	M	6.1
2.1.16	ifOutOctets	Counter	R	The total number of octets transmitted out of the interface including framing octets.	M	9.1
2.1.17	ifOutUcastPkts	Counter	R	The total number of unicast packet whose transmission to a single address was requested.	M	9.1
2.1.19	ifOutDiscards	Counter	R	The number of outbound packets that were free of errors but discarded. (i.e. packets that were filtered out, e.g. to free up memory).	M	6.1
2.1.20	ifOutErrors	Counter	R	The number of outbound packets discarded because of errors.	M	6.1

17.2.4.3.3 ifMIBObjects Group (31.1)

The ifMIBObjects group is under the iso.org.dod.internet.mgnt.mib-2.ifMIB branch. Some objects from the ifMIBObjects group RFC 2863 [23] shall be supported by the RCST. These objects contain the *ifStackTable* and the performance counters for multicast and broadcast.

Table 17-17 - MIB-II *ifMIB. ifMIBObjects* Group Objects

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
1	ifXTable	Sequence of ifXEntry	N	List of *ifMIB* entries. The number of entries is specified by *ifXNumber*.	M	9.1
1.1	ifXEntry	Sequence	N	An interface entry containing objects at the sub-network layer and below for a particular interface.	M	6.1
1.1.1	ifName	DisplayString Size(0..255)	R	A unique name for each interface.	M Name is vendor specific	6.1
1.1.2	ifInMulticastPkts	Counter32	R	The total number of multicast Pkts received on the interface.	M	6.1
1.1.3	ifInBroadcastPkts	Counter32	R	The total number of broadcast Pkts received on the interface.	M	6.1
1.1.4	ifOutMulticastPkts	Counter32	R	The total number of multicast Pkts received on the interface.	M	6.1
1.1.5	ifOutBroadcastPkts	Counter32	R	The total number of multicast Pkts received on the interface.	M	6.1

OID	Name	Syntax	Access	Description	Satlabs Applicability	Section
2	ifStackTable	Sequence of ifStackEntry	N	This table contains information on the relationships between the multiple sub-layers of network interfaces.	M	
2.1	ifStackEntry	Sequence	N	Information on a particular relationship between two sub-layers, specifying that one sub-layer runs on 'top' of the other sub-layer.	M	
2.1.1	ifStackHigherLayer	Interface index	N	The value of ifIndex corresponding to the higher sub-layer of the relationship. (value can be 0 if there is no higher sub-layer)	M	
2.1.2	ifStackLowerLayer	Interface index	N	The value of ifIndex corresponding to the lower sub-layer of the relationship. (value can be 0 if there is no lower sub-layer)	M	
2.1.3	ifStackStatus	Row Status	R	The status of the relationship between two sub-layers. (Values are active, notInService or destroy)	M	
6	ifStacklastChange	Time Ticks	R	The value of sysUpTime at the time of the last change of the interface stack. (contains 0 if there is no changes since the last re-initialization of the local network management subsystem)	M	

17.2.5 SNMP Response Code

The SNMP Response-PDU error status codes shall be issued in compliance with [1].

18 Appendix H – Configuration file format

The configuration file, introduced in section 7.3, shall be formatted as an XML document with a DTD (Document Type Definition). The DTD is to define the legal building blocks of the XML document. It defines the document structure with a list of legal elements. The DTD can be declared inline in your XML document, or as an external reference.

Both an XML document example and the DTD are expanded below.

DTD file "configScheme.dtd":

```
<!-- This dtd contains a sample set of MIB parameters that a configuration file can have. -->
<!-- Parameters that the config file MUST have (according to SSR M&C 1.1) are all part of this set. -->
<!-- In addition, some optional parameters have been added to illustrate the usage of the dtd.   -->
<!-- See the configFileParameters-list below whether parameters are mandatory or not. -->
<!-- The list should eventually contain all parameters described in dvbrcs mib                                                    -->
<!-- Legend in configFileParameters:  -->
<!--                                  No postfix: parameter MUST occure once in configfile [1..1]
                -->
<!--                                  ?: parameter  CAN occure once in configfile          [0..1]
                                     -->
<!--                                  +: parameter MUST occure at least once in configfile [1..n]
                -->
<!--                                  *: parameter CAN occure once or more times in configfile [0..n]
        -->
<!ELEMENT configFileParameters (configFileIdentifier,snmpWriteCommunity,snmpReadCommunity,dvbRcsSystemMibRevision?,
dvbRcsSystemSatLabsOptionsDeclaration?,dvbRcsSystemLocation?,dvbRcsSystemOduAntennaSize?,dvbRcsSystemOduAntennaGai
n?,dvbRcsSystemOduSspa?,
dvbRcsSystemOduTxType?,dvbRcsSystemOduRxType?,dvbRcsSystemOduRxBand?,dvbRcsSystemOduRxLO?,dvbRcsSystemOduTx
LO?,dvbRcsNetworkOamInetAddr?,
dvbRcsNetworkOamInetAddressPrefixLength?,dvbRcsNetworkOamInetAddrAssign?,dvbRcsNetworkLanInetAddress,dvbRcsNetworkLa
nInetAddressPrefixLength,
dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress?,dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressPrefixLength?,
dvbRcsNetworkNccMgtInetAddress?,dvbRcsNetworkNccMgtInetAddressPrefixLength?,dvbRcsNetworkConfigFileDownloadUrl?,
dvbRcsNetworkInstallLogFileDownloadUrl?,dvbRcsNetworkConfigFileUploadUrl?,
dvbRcsNetworkLogFileUploadUrl?,dvbRcsNetworkInstallLogFileUploadUrl?,dvbRcsInstallAntennaAlignmentState?,dvbRcsInstallCwFre
quency?,
dvbRcsInstallCwMaxDuration?,dvbRcsInstallCwPower?,dvbRcsInstallCoPolReading?,dvbRcsInstallXPolReading?,dvbRcsInstallCoPolT
arget?,
dvbRcsInstallXPolTarget?,dvbRcsInstallStandByDuration?,dvbRcsInstallTargetEsN0?,dvbRcsPktClassDscpLow+,dvbRcsPktClassDscp
High+,
dvbRcsPktClassDscpMarkValue+,dvbRcsPktClassIPProtocol+,dvbRcsPktClassSrcInetAddressType+,dvbRcsPktClassSrcInetAddress+,
dvbRcsPktClassSrcInetAddressPrefixLength+,dvbRcsPktClassDstInetAddressType+,dvbRcsPktClassDstInetAddress+,
dvbRcsPktClassDstInetAddressPrefixLength+,dvbRcsPktClassSrcPortLow+,dvbRcsPktClassSrcPortHigh+,dvbRcsPktClassDstPortLow
+,dvbRcsPktClassDstPortHigh+,
dvbRcsPktClassVlanUserPri+,dvbRcsPktClassPhbAssociation+,dvbRcsPktClassRowStatus+,dvbRcsPhbName+,
dvbRcsPhbRequestClassAssociation+,dvbRcsPhbMappingRowStatus+,dvbRcsRequestClassName+,dvbRcsRequestClassChanId+,dv
bRcsRequestClassVccVpi+,
dvbRcsRequestClassVccVci+,dvbRcsRequestClassPidPoolReference*,dvbRcsRequestClassCra+,dvbRcsRequestClassRbdcMax+,dvb
RcsRequestClassRbdcTimeout+,
dvbRcsRequestClassVbdcMax+,dvbRcsRequestClassVbdcTimeout+,dvbRcsRequestClassVbdcMaxBackLog+,
dvbRcsRequestClassRowStatus+,dvbRcsPidIndex*,dvbRcsPidValue*,dvbRcsPidPoolRowStatus*,dvbRcsQosGlobalRbdcMax?,dvbRcs
QosGlobalVbdcMax?,
dvbRcsQosGlobalVbdcMaxBackLog?,dvbRcsQosChannelIdStrictDispatching?,dvbRcsCtrlRebootCommand?,
dvbRcsCtrlRcstTxDisable?,dvbRcsCtrlUserTrafficDisable?,dvbRcsCtrlCwEnable?,dvbRcsCtrlOduTxReferenceEnable?,dvbRcsCtrlOduT
xDCEnable?,
dvbRcsCtrlOduRxDCEnable?,dvbRcsCtrlDownloadFileCommand?,dvbRcsCtrlUploadFileCommand?,dvbRcsCtrlActivateConfigFileCom
mand?,
dvbRcsCtrlRcstLogonCommand?,dvbRcsCtrlRcstLogoffCommand?,dvbRcsRcstMode?,dvbRcsFwdStartPopId+,dvbRcsFwdStartFreque
ncy+,dvbRcsFwdStartPolar+,
dvbRcsFwdStartFormat+,dvbRcsFwdStartRolloff+,dvbRcsFwdStartSymbolRate+,dvbRcsFwdStartInnerFec+,dvbRcsFwdStartRowStatu
s+,dvbRcsRtnConfigMaxEirp?,
dvbRcsRtnConfigDefIfLevel?, dvbrcsVendorspecificMIB?)>
<!--  The following defines all allowed elements: -->
```

<!ELEMENT configFileIdentifier (Value)>
<!ELEMENT snmpWriteCommunity ANY>
<!ELEMENT snmpReadCommunity ANY>
<!ELEMENT dvbRcsSystemMibRevision ANY>
<!ELEMENT dvbRcsSystemSatLabsOptionsDeclaration ANY>
<!ELEMENT dvbRcsSystemLocation ANY>
<!ELEMENT dvbRcsSystemOduAntennaSize ANY>
<!ELEMENT dvbRcsSystemOduAntennaGain ANY>
<!ELEMENT dvbRcsSystemOduSspa ANY>
<!ELEMENT dvbRcsSystemOduTxType ANY>
<!ELEMENT dvbRcsSystemOduRxType ANY>
<!ELEMENT dvbRcsSystemOduRxBand ANY>
<!ELEMENT dvbRcsSystemOduRxLO ANY>
<!ELEMENT dvbRcsSystemOduTxLO ANY>
<!ELEMENT dvbRcsNetworkOamInetAddr ANY>
<!ELEMENT dvbRcsNetworkOamInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsNetworkOamInetAddrAssign ANY>
<!ELEMENT dvbRcsNetworkLanInetAddress ANY>
<!ELEMENT dvbRcsNetworkLanInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress ANY>
<!ELEMENT dvbRcsNetworkAirInterfaceDefaultGatewayInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsNetworkNccMgtInetAddress ANY>
<!ELEMENT dvbRcsNetworkNccMgtInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsNetworkConfigFileDownloadUrl ANY>
<!ELEMENT dvbRcsNetworkInstallLogFileDownloadUrl ANY>
<!ELEMENT dvbRcsNetworkConfigFileUploadUrl ANY>
<!ELEMENT dvbRcsNetworkLogFileUploadUrl ANY>
<!ELEMENT dvbRcsNetworkInstallLogFileUploadUrl ANY>
<!ELEMENT dvbRcsInstallAntennaAlignmentState ANY>
<!ELEMENT dvbRcsInstallCwFrequency ANY>
<!ELEMENT dvbRcsInstallCwMaxDuration ANY>
<!ELEMENT dvbRcsInstallCwPower ANY>
<!ELEMENT dvbRcsInstallCoPolReading ANY>
<!ELEMENT dvbRcsInstallXPolReading ANY>
<!ELEMENT dvbRcsInstallCoPolTarget ANY>
<!ELEMENT dvbRcsInstallXPolTarget ANY>
<!ELEMENT dvbRcsInstallStandByDuration ANY>
<!ELEMENT dvbRcsInstallTargetEsN0 ANY>
<!ELEMENT dvbRcsPktClassDscpLow ANY>
<!ELEMENT dvbRcsPktClassDscpHigh ANY>
<!ELEMENT dvbRcsPktClassDscpMarkValue ANY>
<!ELEMENT dvbRcsPktClassIPProtocol ANY>
<!ELEMENT dvbRcsPktClassSrcInetAddressType ANY>
<!ELEMENT dvbRcsPktClassSrcInetAddress ANY>
<!ELEMENT dvbRcsPktClassSrcInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsPktClassDstInetAddressType ANY>
<!ELEMENT dvbRcsPktClassDstInetAddress ANY>
<!ELEMENT dvbRcsPktClassDstInetAddressPrefixLength ANY>
<!ELEMENT dvbRcsPktClassSrcPortLow ANY>
<!ELEMENT dvbRcsPktClassSrcPortHigh ANY>
<!ELEMENT dvbRcsPktClassDstPortLow ANY>
<!ELEMENT dvbRcsPktClassDstPortHigh ANY>
<!ELEMENT dvbRcsPktClassVlanUserPri ANY>
<!ELEMENT dvbRcsPktClassPhbAssociation ANY>
<!ELEMENT dvbRcsPktClassRowStatus ANY>
<!ELEMENT dvbRcsPhbName ANY>
<!ELEMENT dvbRcsPhbRequestClassAssociation ANY>
<!ELEMENT dvbRcsPhbMappingRowStatus ANY>
<!ELEMENT dvbRcsRequestClassName ANY>
<!ELEMENT dvbRcsRequestClassChanId ANY>
<!ELEMENT dvbRcsRequestClassVccVpi ANY>
<!ELEMENT dvbRcsRequestClassVccVci ANY>
<!ELEMENT dvbRcsRequestClassPidPoolReference ANY>
<!ELEMENT dvbRcsRequestClassCra ANY>
<!ELEMENT dvbRcsRequestClassRbdcMax ANY>
<!ELEMENT dvbRcsRequestClassRbdcTimeout ANY>
<!ELEMENT dvbRcsRequestClassVbdcMax ANY>
<!ELEMENT dvbRcsRequestClassVbdcTimeout ANY>
<!ELEMENT dvbRcsRequestClassVbdcMaxBackLog ANY>
<!ELEMENT dvbRcsRequestClassRowStatus ANY>

```
<!ELEMENT dvbRcsPidIndex ANY>
<!ELEMENT dvbRcsPidValue ANY>
<!ELEMENT dvbRcsPidPoolRowStatus ANY>
<!ELEMENT dvbRcsQosGlobalRbdcMax ANY>
<!ELEMENT dvbRcsQosGlobalVbdcMax ANY>
<!ELEMENT dvbRcsQosGlobalVbdcMaxBackLog ANY>
<!ELEMENT dvbRcsQosChannelIdStrictDispatching ANY>
<!ELEMENT dvbRcsCtrlRebootCommand ANY>
<!ELEMENT dvbRcsCtrlRcstTxDisable ANY>
<!ELEMENT dvbRcsCtrlUserTrafficDisable ANY>
<!ELEMENT dvbRcsCtrlCwEnable ANY>
<!ELEMENT dvbRcsCtrlOduTxReferenceEnable ANY>
<!ELEMENT dvbRcsCtrlOduTxDCEnable ANY>
<!ELEMENT dvbRcsCtrlOduRxDCEnable ANY>
<!ELEMENT dvbRcsCtrlDownloadFileCommand ANY>
<!ELEMENT dvbRcsCtrlUploadFileCommand ANY>
<!ELEMENT dvbRcsCtrlActivateConfigFileCommand ANY>
<!ELEMENT dvbRcsCtrlRcstLogonCommand ANY>
<!ELEMENT dvbRcsCtrlRcstLogoffCommand ANY>
<!ELEMENT dvbRcsRcstMode ANY>
<!ELEMENT dvbRcsFwdStartPopId ANY>
<!ELEMENT dvbRcsFwdStartFrequency ANY>
<!ELEMENT dvbRcsFwdStartPolar ANY>
<!ELEMENT dvbRcsFwdStartFormat ANY>
<!ELEMENT dvbRcsFwdStartRolloff ANY>
<!ELEMENT dvbRcsFwdStartSymbolRate ANY>
<!ELEMENT dvbRcsFwdStartInnerFec ANY>
<!ELEMENT dvbRcsFwdStartRowStatus ANY>
<!ELEMENT dvbRcsRtnConfigMaxEirp ANY>
<!ELEMENT dvbRcsRtnConfigDefIfLevel ANY>
<!ELEMENT dvbrcsVendorspecificMIB (ObjectID*)>
<!ELEMENT Value ANY>
<!ELEMENT ObjectID ANY>
<!--  The following defines expected OID attributes of the scalar MIB values : -->
<!ATTLIST dvbRcsSystemMibRevision
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.1.0">
<!ATTLIST dvbRcsSystemSatLabsOptionsDeclaration
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.3.0">
<!ATTLIST dvbRcsSystemLocation
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.5.0">
<!ATTLIST dvbRcsSystemOduAntennaSize
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.6.0">
<!ATTLIST dvbRcsSystemOduAntennaGain
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.7.0">
<!ATTLIST dvbRcsSystemOduSspa
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.8.0">
<!ATTLIST dvbRcsSystemOduTxType
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.9.0">
<!ATTLIST dvbRcsSystemOduRxType
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.10.0">
<!ATTLIST dvbRcsSystemOduRxBand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.11.0">
<!ATTLIST dvbRcsSystemOduRxLO
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.12.0">
<!ATTLIST dvbRcsSystemOduTxLO
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.1.13.0">
<!ATTLIST dvbRcsNetworkOamInetAddress
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.2.0">
<!ATTLIST dvbRcsNetworkOamInetAddressPrefixLength
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.3.0">
<!ATTLIST dvbRcsNetworkOamInetAddrAssign
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.4.0">
<!ATTLIST dvbRcsNetworkLanInetAddress
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.6.0">
<!ATTLIST dvbRcsNetworkLanInetAddressPrefixLength
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.7.0">
<!ATTLIST dvbRcsNetworkAirInterfaceDefaultGatewayInetAddress
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.9.0">
<!ATTLIST dvbRcsNetworkNccMgtInetAddress
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.13.0">
```

```
<!ATTLIST dvbRcsNetworkNccMgtInetAddressPrefixLength
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.14.0">
<!ATTLIST dvbRcsNetworkConfigFileDownloadUrl
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.15.0">
<!ATTLIST dvbRcsNetworkInstallLogFileDownloadUrl
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.16.0">
<!ATTLIST dvbRcsNetworkConfigFileUploadUrl
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.17.0">
<!ATTLIST dvbRcsNetworkLogFileUploadUrl
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.18.0">
<!ATTLIST dvbRcsNetworkInstallLogFileUploadUrl
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.2.19.0">
<!ATTLIST dvbRcsInstallAntennaAlignmentState
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.1.0">
<!ATTLIST dvbRcsInstallCwFrequency
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.2.0">
<!ATTLIST dvbRcsInstallCwMaxDuration
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.3.0">
<!ATTLIST dvbRcsInstallCwPower
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.4.0">
<!ATTLIST dvbRcsInstallCoPolReading
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.5.0">
<!ATTLIST dvbRcsInstallXPolReading
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.6.0">
<!ATTLIST dvbRcsInstallCoPolTarget
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.7.0">
<!ATTLIST dvbRcsInstallXPolTarget
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.8.0">
<!ATTLIST dvbRcsInstallStandByDuration
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.9.0">
<!ATTLIST dvbRcsInstallTargetEsN0
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.3.10.0">
<!--  The following defines attributes of the QoS table MIB values (OIDs not checked): -->
<!ATTLIST dvbRcsPktClassDscpLow
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDscpHigh
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDscpMarkValue
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassIPProtocol
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassSrcInetAddressType
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassSrcInetAddress
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassSrcInetAddressPrefixLength
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDstInetAddressType
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDstInetAddress
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDstInetAddressPrefixLength
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassSrcPortLow
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassSrcPortHigh
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDstPortLow
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassDstPortHigh
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassVlanUserPri
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassPhbAssociation
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPktClassRowStatus
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPhbName
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPhbRequestClassAssociation
```

```
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPhbMappingRowStatus
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassName
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassChanId
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassVccVpi
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassVccVci
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassPidPoolReference
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassCra
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassRbdcMax
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassRbdcTimeout
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassVbdcMax
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassVbdcTimeout
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassVbdcMaxBackLog
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsRequestClassRowStatus
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPidIndex ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPidValue ID CDATA #IMPLIED>
<!ATTLIST dvbRcsPidPoolRowStatus ID CDATA #IMPLIED>
<!--  The following defines expected OID attributes of the scalar MIB values : -->
<!ATTLIST dvbRcsQosGlobalRbdcMax
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.4.5.0">
<!ATTLIST dvbRcsQosGlobalVbdcMax
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.4.6.0">
<!ATTLIST dvbRcsQosGlobalVbdcMaxBackLog
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.4.7.0">
<!ATTLIST dvbRcsQosChannelIdStrictDispatching
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.4.8.0">
<!ATTLIST dvbRcsCtrlRebootCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.1.0">
<!ATTLIST dvbRcsCtrlRcstTxDisable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.2.0">
<!ATTLIST dvbRcsCtrlUserTrafficDisable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.3.0">
<!ATTLIST dvbRcsCtrlCwEnable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.4.0">
<!ATTLIST dvbRcsCtrlOduTxReferenceEnable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.5.0">
<!ATTLIST dvbRcsCtrlOduTxDCEnable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.6.0">
<!ATTLIST dvbRcsCtrlOduRxDCEnable
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.7.0">
<!ATTLIST dvbRcsCtrlDownloadFileCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.8.0">
<!ATTLIST dvbRcsCtrlUploadFileCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.9.0">
<!ATTLIST dvbRcsCtrlActivateConfigFileCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.10.0">
<!ATTLIST dvbRcsCtrlRcstLogonCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.11.0">
<!ATTLIST dvbRcsCtrlRcstLogoffCommand
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.5.12.0">
<!ATTLIST dvbRcsRcstMode
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.1.6.1.0">
<!--  The following defines attributes of the Forward link start table MIB values (OIDs not checked): -->
<!ATTLIST dvbRcsFwdStartPopId
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartFrequency
ID CDATA #IMPLIED>
```

```
<!ATTLIST dvbRcsFwdStartPolar
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartFormat
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartRolloff
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartSymbolRate
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartInnerFec
ID CDATA #IMPLIED>
<!ATTLIST dvbRcsFwdStartRowStatus
ID CDATA #IMPLIED>
<!--  The following defines expected OID attributes of the scalar MIB values : -->
<!ATTLIST dvbRcsRtnConfigMaxEirp
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.3.1.1.0">
<!ATTLIST dvbRcsRtnConfigDefIfLevel
ID CDATA #FIXED "1.3.6.1.2.1.10.239.1.3.1.2.0">
<!ATTLIST ObjectID
ID CDATA #IMPLIED>
```

XML file example:

```
  <?xml version="1.0" encoding="UTF-8" ?>
  <!DOCTYPE configFileParameters (View Source for full doctype...)>
- <configFileParameters>
- <configFileIdentifier>
  <Value>5.1</Value>
  </configFileIdentifier>
- <snmpWriteCommunity>
  <Value>public</Value>
  </snmpWriteCommunity>
- <snmpReadCommunity>
  <Value>public</Value>
  </snmpReadCommunity>
- <!--  dvbRcsRcstSystem params  -->
- <dvbRcsSystemLocation ID="1.3.6.1.2.1.10.239.1.1.1.5.0">
  <Value>4916.466,N,12311.122,W,545.4,M</Value>
  </dvbRcsSystemLocation>
- <dvbRcsSystemOduRxType ID="1.3.6.1.2.1.10.239.1.1.1.10.0">
  <Value>quad</Value>
  </dvbRcsSystemOduRxType>
- <!--  dvbRcsRcstNetwork params -->
- <dvbRcsNetworkLanInetAddress ID="1.3.6.1.2.1.10.239.1.1.2.6.0">
  <Value>172.16.195.1</Value>
  </dvbRcsNetworkLanInetAddress>
- <dvbRcsNetworkLanInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.2.7.0">
  <Value>0</Value>
  </dvbRcsNetworkLanInetAddressPrefixLength>
- <!--  dvbRcsRcstInstall params -->
- <!--  dvbRcsRcstQoS params -->
- <!--  Packet classification table -->
- <dvbRcsPktClassDscpLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.2.1">
  <Value>0</Value>
  </dvbRcsPktClassDscpLow>
- <dvbRcsPktClassDscpLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.2.2">
  <Value>30</Value>
  </dvbRcsPktClassDscpLow>
- <dvbRcsPktClassDscpLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.2.3">
  <Value>46</Value>
  </dvbRcsPktClassDscpLow>
- <dvbRcsPktClassDscpHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.3.1">
  <Value>0</Value>
  </dvbRcsPktClassDscpHigh>
- <dvbRcsPktClassDscpHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.3.2">
  <Value>30</Value>
  </dvbRcsPktClassDscpHigh>
- <dvbRcsPktClassDscpHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.3.3">
```

<Value>46</Value>
</dvbRcsPktClassDscpHigh>
- <dvbRcsPktClassDscpMarkValue ID="1.3.6.1.2.1.10.239.1.1.4.1.1.4.1">
<Value>0</Value>
</dvbRcsPktClassDscpMarkValue>
- <dvbRcsPktClassDscpMarkValue ID="1.3.6.1.2.1.10.239.1.1.4.1.1.4.2">
<Value>30</Value>
</dvbRcsPktClassDscpMarkValue>
- <dvbRcsPktClassDscpMarkValue ID="1.3.6.1.2.1.10.239.1.1.4.1.1.4.3">
<Value>46</Value>
</dvbRcsPktClassDscpMarkValue>
- <dvbRcsPktClassIPProtocol ID="1.3.6.1.2.1.10.239.1.1.4.1.1.5.1">
<Value>255</Value>
</dvbRcsPktClassIPProtocol>
- <dvbRcsPktClassIPProtocol ID="1.3.6.1.2.1.10.239.1.1.4.1.1.5.2">
<Value>255</Value>
</dvbRcsPktClassIPProtocol>
- <dvbRcsPktClassIPProtocol ID="1.3.6.1.2.1.10.239.1.1.4.1.1.5.3">
<Value>255</Value>
</dvbRcsPktClassIPProtocol>
- <dvbRcsPktClassSrcInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.6.1">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressType>
- <dvbRcsPktClassSrcInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.6.2">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressType>
- <dvbRcsPktClassSrcInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.6.3">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressType>
- <dvbRcsPktClassSrcInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.7.1">
<Value>0.0.0.0</Value>
</dvbRcsPktClassSrcInetAddress>
- <dvbRcsPktClassSrcInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.7.2">
<Value>0.0.0.0</Value>
</dvbRcsPktClassSrcInetAddress>
- <dvbRcsPktClassSrcInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.7.3">
<Value>0.0.0.0</Value>
</dvbRcsPktClassSrcInetAddress>
- <dvbRcsPktClassSrcInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.8.1">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressPrefixLength>
- <dvbRcsPktClassSrcInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.8.2">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressPrefixLength>
- <dvbRcsPktClassSrcInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.8.3">
<Value>0</Value>
</dvbRcsPktClassSrcInetAddressPrefixLength>
- <dvbRcsPktClassDstInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.9.1">
<Value>0</Value>
</dvbRcsPktClassDstInetAddressType>
- <dvbRcsPktClassDstInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.9.2">
<Value>0</Value>
</dvbRcsPktClassDstInetAddressType>
- <dvbRcsPktClassDstInetAddressType ID="1.3.6.1.2.1.10.239.1.1.4.1.1.9.3">
<Value>0</Value>
</dvbRcsPktClassDstInetAddressType>
- <dvbRcsPktClassDstInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.10.1">
<Value>0.0.0.0</Value>
</dvbRcsPktClassDstInetAddress>
- <dvbRcsPktClassDstInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.10.2">
<Value>0.0.0.0</Value>
</dvbRcsPktClassDstInetAddress>
- <dvbRcsPktClassDstInetAddress ID="1.3.6.1.2.1.10.239.1.1.4.1.1.10.3">
<Value>0.0.0.0</Value>
</dvbRcsPktClassDstInetAddress>
- <dvbRcsPktClassDstInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.11.1">
<Value>0</Value>
</dvbRcsPktClassDstInetAddressPrefixLength>
- <dvbRcsPktClassDstInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.11.2">
<Value>0</Value>

```
  </dvbRcsPktClassDstInetAddressPrefixLength>
- <dvbRcsPktClassDstInetAddressPrefixLength ID="1.3.6.1.2.1.10.239.1.1.4.1.1.11.3">
  <Value>0</Value>
  </dvbRcsPktClassDstInetAddressPrefixLength>
- <dvbRcsPktClassSrcPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.12.1">
  <Value>0</Value>
  </dvbRcsPktClassSrcPortLow>
- <dvbRcsPktClassSrcPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.12.2">
  <Value>0</Value>
  </dvbRcsPktClassSrcPortLow>
- <dvbRcsPktClassSrcPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.12.3">
  <Value>0</Value>
  </dvbRcsPktClassSrcPortLow>
- <dvbRcsPktClassSrcPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.13.1">
  <Value>65535</Value>
  </dvbRcsPktClassSrcPortHigh>
- <dvbRcsPktClassSrcPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.13.2">
  <Value>65535</Value>
  </dvbRcsPktClassSrcPortHigh>
- <dvbRcsPktClassSrcPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.13.3">
  <Value>65535</Value>
  </dvbRcsPktClassSrcPortHigh>
- <dvbRcsPktClassDstPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.14.1">
  <Value>0</Value>
  </dvbRcsPktClassDstPortLow>
- <dvbRcsPktClassDstPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.14.2">
  <Value>0</Value>
  </dvbRcsPktClassDstPortLow>
- <dvbRcsPktClassDstPortLow ID="1.3.6.1.2.1.10.239.1.1.4.1.1.14.3">
  <Value>0</Value>
  </dvbRcsPktClassDstPortLow>
- <dvbRcsPktClassDstPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.15.1">
  <Value>65535</Value>
  </dvbRcsPktClassDstPortHigh>
- <dvbRcsPktClassDstPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.15.2">
  <Value>65535</Value>
  </dvbRcsPktClassDstPortHigh>
- <dvbRcsPktClassDstPortHigh ID="1.3.6.1.2.1.10.239.1.1.4.1.1.15.3">
  <Value>65535</Value>
  </dvbRcsPktClassDstPortHigh>
- <dvbRcsPktClassVlanUserPri ID="1.3.6.1.2.1.10.239.1.1.4.1.1.16.1">
  <Value>-1</Value>
  </dvbRcsPktClassVlanUserPri>
- <dvbRcsPktClassVlanUserPri ID="1.3.6.1.2.1.10.239.1.1.4.1.1.16.2">
  <Value>-1</Value>
  </dvbRcsPktClassVlanUserPri>
- <dvbRcsPktClassVlanUserPri ID="1.3.6.1.2.1.10.239.1.1.4.1.1.16.3">
  <Value>-1</Value>
  </dvbRcsPktClassVlanUserPri>
- <dvbRcsPktClassPhbAssociation ID="1.3.6.1.2.1.10.239.1.1.4.1.1.17.1">
  <Value>1</Value>
  </dvbRcsPktClassPhbAssociation>
- <dvbRcsPktClassPhbAssociation ID="1.3.6.1.2.1.10.239.1.1.4.1.1.17.2">
  <Value>2</Value>
  </dvbRcsPktClassPhbAssociation>
- <dvbRcsPktClassPhbAssociation ID="1.3.6.1.2.1.10.239.1.1.4.1.1.17.3">
  <Value>3</Value>
  </dvbRcsPktClassPhbAssociation>
- <dvbRcsPktClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.1.1.18.1">
  <Value>4</Value>
  </dvbRcsPktClassRowStatus>
- <dvbRcsPktClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.1.1.18.2">
  <Value>4</Value>
  </dvbRcsPktClassRowStatus>
- <dvbRcsPktClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.1.1.18.3">
  <Value>4</Value>
  </dvbRcsPktClassRowStatus>
- <!--  PHB mapping table -->
- <dvbRcsPhbName ID="1.3.6.1.2.1.10.239.1.1.4.2.1.2.1">
  <Value>BE</Value>
```

```
  </dvbRcsPhbName>
- <dvbRcsPhbName ID="1.3.6.1.2.1.10.239.1.1.4.2.1.2.2">
  <Value>AF</Value>
  </dvbRcsPhbName>
- <dvbRcsPhbName ID="1.3.6.1.2.1.10.239.1.1.4.2.1.2.3">
  <Value>EF</Value>
  </dvbRcsPhbName>
- <dvbRcsPhbRequestClassAssociation ID="1.3.6.1.2.1.10.239.1.1.4.2.1.3.1">
  <Value>1</Value>
  </dvbRcsPhbRequestClassAssociation>
- <dvbRcsPhbRequestClassAssociation ID="1.3.6.1.2.1.10.239.1.1.4.2.1.3.2">
  <Value>2</Value>
  </dvbRcsPhbRequestClassAssociation>
- <dvbRcsPhbRequestClassAssociation ID="1.3.6.1.2.1.10.239.1.1.4.2.1.3.3">
  <Value>3</Value>
  </dvbRcsPhbRequestClassAssociation>
- <dvbRcsPhbMappingRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.2.1.4.1">
  <Value>4</Value>
  </dvbRcsPhbMappingRowStatus>
- <dvbRcsPhbMappingRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.2.1.4.2">
  <Value>4</Value>
  </dvbRcsPhbMappingRowStatus>
- <dvbRcsPhbMappingRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.2.1.4.3">
  <Value>4</Value>
  </dvbRcsPhbMappingRowStatus>
- <!--  Request class table -->
- <dvbRcsRequestClassName ID="1.3.6.1.2.1.10.239.1.1.4.3.1.2.1">
  <Value>BE</Value>
  </dvbRcsRequestClassName>
- <dvbRcsRequestClassName ID="1.3.6.1.2.1.10.239.1.1.4.3.1.2.2">
  <Value>CD</Value>
  </dvbRcsRequestClassName>
- <dvbRcsRequestClassName ID="1.3.6.1.2.1.10.239.1.1.4.3.1.2.3">
  <Value>RT</Value>
  </dvbRcsRequestClassName>
- <dvbRcsRequestClassChanId ID="1.3.6.1.2.1.10.239.1.1.4.3.1.3.1">
  <Value>0</Value>
  </dvbRcsRequestClassChanId>
- <dvbRcsRequestClassChanId ID="1.3.6.1.2.1.10.239.1.1.4.3.1.3.2">
  <Value>2</Value>
  </dvbRcsRequestClassChanId>
- <dvbRcsRequestClassChanId ID="1.3.6.1.2.1.10.239.1.1.4.3.1.3.3">
  <Value>3</Value>
  </dvbRcsRequestClassChanId>
- <dvbRcsRequestClassVccVpi ID="1.3.6.1.2.1.10.239.1.1.4.3.1.4.1">
  <Value>0</Value>
  </dvbRcsRequestClassVccVpi>
- <dvbRcsRequestClassVccVpi ID="1.3.6.1.2.1.10.239.1.1.4.3.1.4.2">
  <Value>0</Value>
  </dvbRcsRequestClassVccVpi>
- <dvbRcsRequestClassVccVpi ID="1.3.6.1.2.1.10.239.1.1.4.3.1.4.3">
  <Value>0</Value>
  </dvbRcsRequestClassVccVpi>
- <dvbRcsRequestClassVccVci ID="1.3.6.1.2.1.10.239.1.1.4.3.1.5.1">
  <Value>0</Value>
  </dvbRcsRequestClassVccVci>
- <dvbRcsRequestClassVccVci ID="1.3.6.1.2.1.10.239.1.1.4.3.1.5.2">
  <Value>0</Value>
  </dvbRcsRequestClassVccVci>
- <dvbRcsRequestClassVccVci ID="1.3.6.1.2.1.10.239.1.1.4.3.1.5.3">
  <Value>0</Value>
  </dvbRcsRequestClassVccVci>
- <dvbRcsRequestClassPidPoolReference ID="1.3.6.1.2.1.10.239.1.1.4.3.1.6.1">
  <Value>1</Value>
  </dvbRcsRequestClassPidPoolReference>
- <dvbRcsRequestClassPidPoolReference ID="1.3.6.1.2.1.10.239.1.1.4.3.1.6.2">
  <Value>2</Value>
  </dvbRcsRequestClassPidPoolReference>
- <dvbRcsRequestClassPidPoolReference ID="1.3.6.1.2.1.10.239.1.1.4.3.1.6.3">
  <Value>3</Value>
```

```
  </dvbRcsRequestClassPidPoolReference>
- <dvbRcsRequestClassCra ID="1.3.6.1.2.1.10.239.1.1.4.3.1.7.1">
  <Value>5000</Value>
  </dvbRcsRequestClassCra>
- <dvbRcsRequestClassCra ID="1.3.6.1.2.1.10.239.1.1.4.3.1.7.2">
  <Value>5000</Value>
  </dvbRcsRequestClassCra>
- <dvbRcsRequestClassCra ID="1.3.6.1.2.1.10.239.1.1.4.3.1.7.3">
  <Value>5000</Value>
  </dvbRcsRequestClassCra>
- <dvbRcsRequestClassRbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.8.1">
  <Value>100</Value>
  </dvbRcsRequestClassRbdcMax>
- <dvbRcsRequestClassRbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.8.2">
  <Value>100</Value>
  </dvbRcsRequestClassRbdcMax>
- <dvbRcsRequestClassRbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.8.3">
  <Value>100</Value>
  </dvbRcsRequestClassRbdcMax>
- <dvbRcsRequestClassRbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.9.1">
  <Value>50</Value>
  </dvbRcsRequestClassRbdcTimeout>
- <dvbRcsRequestClassRbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.9.2">
  <Value>50</Value>
  </dvbRcsRequestClassRbdcTimeout>
- <dvbRcsRequestClassRbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.9.3">
  <Value>50</Value>
  </dvbRcsRequestClassRbdcTimeout>
- <dvbRcsRequestClassVbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.10.1">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcMax>
- <dvbRcsRequestClassVbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.10.2">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcMax>
- <dvbRcsRequestClassVbdcMax ID="1.3.6.1.2.1.10.239.1.1.4.3.1.10.3">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcMax>
- <dvbRcsRequestClassVbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.11.1">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcTimeout>
- <dvbRcsRequestClassVbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.11.2">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcTimeout>
- <dvbRcsRequestClassVbdcTimeout ID="1.3.6.1.2.1.10.239.1.1.4.3.1.11.3">
  <Value>50</Value>
  </dvbRcsRequestClassVbdcTimeout>
- <dvbRcsRequestClassVbdcMaxBackLog ID="1.3.6.1.2.1.10.239.1.1.4.3.1.12.1">
  <Value>20000</Value>
  </dvbRcsRequestClassVbdcMaxBackLog>
- <dvbRcsRequestClassVbdcMaxBackLog ID="1.3.6.1.2.1.10.239.1.1.4.3.1.12.2">
  <Value>10000</Value>
  </dvbRcsRequestClassVbdcMaxBackLog>
- <dvbRcsRequestClassVbdcMaxBackLog ID="1.3.6.1.2.1.10.239.1.1.4.3.1.12.3">
  <Value>10000</Value>
  </dvbRcsRequestClassVbdcMaxBackLog>
- <dvbRcsRequestClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.3.1.13.1">
  <Value>4</Value>
  </dvbRcsRequestClassRowStatus>
- <dvbRcsRequestClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.3.1.13.2">
  <Value>4</Value>
  </dvbRcsRequestClassRowStatus>
- <dvbRcsRequestClassRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.3.1.13.3">
  <Value>4</Value>
  </dvbRcsRequestClassRowStatus>
- <!--  Pid pool table -->
- <dvbRcsPidIndex ID="1.3.6.1.2.1.10.239.1.1.4.4.1.2.1">
  <Value>1</Value>
  </dvbRcsPidIndex>
- <dvbRcsPidIndex ID="1.3.6.1.2.1.10.239.1.1.4.4.1.2.2">
  <Value>2</Value>
```

```
  </dvbRcsPidIndex>
- <dvbRcsPidIndex ID="1.3.6.1.2.1.10.239.1.1.4.4.1.2.3">
  <Value>3</Value>
  </dvbRcsPidIndex>
- <dvbRcsPidValue ID="1.3.6.1.2.1.10.239.1.1.4.4.1.3.1">
  <Value>600</Value>
  </dvbRcsPidValue>
- <dvbRcsPidValue ID="1.3.6.1.2.1.10.239.1.1.4.4.1.3.2">
  <Value>700</Value>
  </dvbRcsPidValue>
- <dvbRcsPidValue ID="1.3.6.1.2.1.10.239.1.1.4.4.1.3.3">
  <Value>800</Value>
  </dvbRcsPidValue>
- <dvbRcsPidPoolRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.4.1.4.1">
  <Value>4</Value>
  </dvbRcsPidPoolRowStatus>
- <dvbRcsPidPoolRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.4.1.4.2">
  <Value>4</Value>
  </dvbRcsPidPoolRowStatus>
- <dvbRcsPidPoolRowStatus ID="1.3.6.1.2.1.10.239.1.1.4.4.1.4.3">
  <Value>4</Value>
  </dvbRcsPidPoolRowStatus>
- <!--  dvbRcsRcstControl params -->
- <dvbRcsCtrlCwEnable ID="1.3.6.1.2.1.10.239.1.1.5.4.0">
  <Value>1</Value>
  </dvbRcsCtrlCwEnable>
- <!--  dvbRcsRcstState params -->
- <!--  dvbRcsFwdConfig params -->
- <dvbRcsFwdStartPopId ID="1.3.6.1.2.1.10.239.1.2.1.1.1.2.1">
  <Value>22</Value>
  </dvbRcsFwdStartPopId>
- <dvbRcsFwdStartFrequency ID="1.3.6.1.2.1.10.239.1.2.1.1.1.3.1">
  <Value>114879</Value>
  </dvbRcsFwdStartFrequency>
- <dvbRcsFwdStartPolar ID="1.3.6.1.2.1.10.239.1.2.1.1.1.4.1">
  <Value>0</Value>
  </dvbRcsFwdStartPolar>
- <dvbRcsFwdStartFormat ID="1.3.6.1.2.1.10.239.1.2.1.1.1.5.1">
  <Value>0</Value>
  </dvbRcsFwdStartFormat>
- <dvbRcsFwdStartRolloff ID="1.3.6.1.2.1.10.239.1.2.1.1.1.6.1">
  <Value>2</Value>
  </dvbRcsFwdStartRolloff>
- <dvbRcsFwdStartSymbolRate ID="1.3.6.1.2.1.10.239.1.2.1.1.1.7.1">
  <Value>88888</Value>
  </dvbRcsFwdStartSymbolRate>
- <dvbRcsFwdStartInnerFec ID="1.3.6.1.2.1.10.239.1.2.1.1.1.8.1">
  <Value>1</Value>
  </dvbRcsFwdStartInnerFec>
- <dvbRcsFwdStartRowStatus ID="1.3.6.1.2.1.10.239.1.2.1.1.1.9.1">
  <Value>4</Value>
  </dvbRcsFwdStartRowStatus>
- <!--  Vendor specific params  -->
- <dvbrcsVendorspecificMIB>
- <ObjectID ID="1.3.6.1.2.1.1.2.0">
  <Value>255.255.255.0</Value>
  </ObjectID>
- <ObjectID ID="1.3.6.1.2.1.1.3.0">
  <Value>location</Value>
  </ObjectID>
  </dvbrcsVendorspecificMIB>
  </configFileParameters>
```