NewNet Mobile Messaging HUB R04.10.08

Operator Manual

Release 16.8 Revision A

March 2018

Table of Contents

List of Figures

Chapter 1

Introduction

Topics:

1.1 About this Document

This document contains all relevant details required for the operation and administration of the NewNet Mobile Messaging HUB component.

The HUB is a component in the NewNet Mobile Messaging product family of SS7 message routing and network querying products.

This document contains a description of the general operations and maintenance aspects of the HUB. Because the available functions are licensed and depend on the specific NewNet Mobile Messaging implementation, not all functions and/or applications contained in this document may be relevant or applicable to the NewNet Mobile Messaging system you will be working with. Actual screen presentations may differ from the screens presented in this document, depending on the software version or browser configurations.

1.2 Scope

This document discusses the functionality of the NewNet Mobile Messaging HUB component.

1.3 Intended Audience

This document is meant for all those interested in how NewNet Mobile Messaging can best be used, but mainly for:

- Implementation Engineers who are responsible for the pre-installation, on-site installation and configuration of NewNet Mobile Messaging products in the end-user environment.
- Maintenance and Support Engineers who are responsible for maintaining the total system environment of which NewNet Mobile Messaging is a part or just the NewNet Mobile Messaging devices.
- Network Operators who are in charge of the daily operation of the NewNet Mobile Messaging systems and infrastructure.

1.4 Documentation Conventions

Typeface or Symbol	Meaning	Example
Bold	Refers to part of a graphical user interface.	Click **Cancel**.
`Courier`	Refers to a directory name, file name, command, or output.	The `billing` directory contains...

Typeface or Symbol	Meaning	Example
`<pointed brackets>`	Serves as a placeholder for text that the user will replace, as appropriate in context.	The file is called `MGRdata.xml.<ip>.gz`, where `<ip>` is the server's IP address.
`[square brackets]`	Indicates an optional command.	`[--validateonly]`
Note:	Indicates information alongside normal text, requiring extra attention.	**Note:** Ensure that the configuration...
\ (Unix)	Denotes line continuation; the character should be ignored as the user types the example, and ENTER should only be pressed after the last line.	`% grep searchkey \` `data/*.dat`

1.5 Locate Product Documentation on the Customer Support Site

Access to NewNet's Customer Support site is restricted to current NewNet customers only. This section describes how to log into the NewNet Customer Support site and locate a document. Viewing the document requires Adobe Acrobat Reader, which can be downloaded at www.adobe.com.

1. Log into the NewNet Customer Support site.

   Note: If you have not registered for this new site, click the **Register Here** link. Have your customer number available. The response time for registration requests is 24 to 48 hours.

2. Click the **Product Support** tab.
3. Use the Search field to locate a document by its part number, release number, document name, or document type. The Search field accepts both full and partial entries.
4. Click a subject folder to browse through a list of related files.
5. To download a file to your location, right-click the file name and select **Save Target As**.

Chapter 2

System Overview

Topics:

2.1 Introduction

The HUB allows you to route SMS traffic from any SMS application to SMS Routers (RTRs), SMS gateways, or SMSCs over various application protocols. The SMS application traffic is routed using load distribution over a configured set of nodes (RTRs or SMSCs). The HUB enables routing SMS traffic directly to mobile recipients in combination with the RTR and can forward the SMS messages to the SMSC for subsequent delivery attempts when required. Additionally, the HUB can relay messages destined to an SMS application connected to the HUB coming from RTRs or SMSCs.

The HUB, in combination with RTRs and optionally the Active Message Store (AMS), offer a reliable optimization for SMS networks.

Note: The HUB requires a connection to at least one RTR in order to function. When the HUB attempts to start, it will search for available RTRs. If no RTRs are found, the HUB will not start.

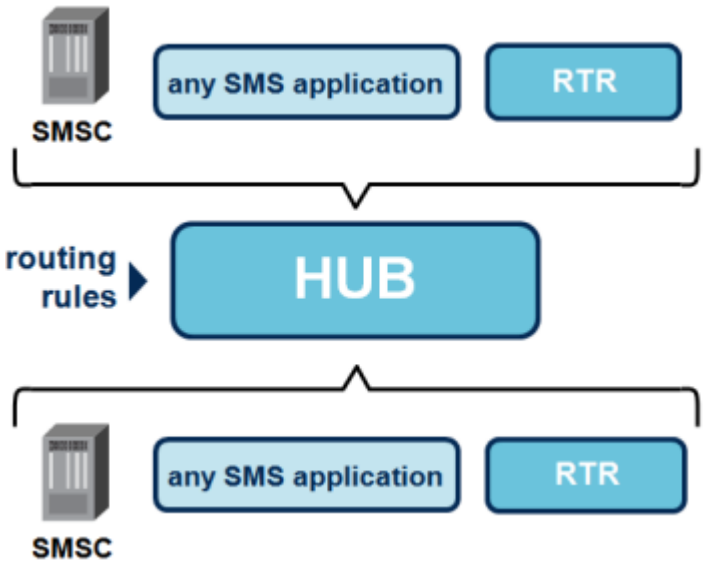


Figure 1: HUB system overview

The HUB routes application-originated (AO) messages to RTRs or SMSCs and routes application-terminated (AT) traffic to traffic destined for interactive/voting applications directly to the related application. This functionality reduces SMS bottlenecks and improves overall quality of service of SMS services and enables optimized use of the existing SMS network infrastructure.

Examples of how the HUB can be used are:

- Routing application-originated (AO) SMS traffic to various SMSCs and RTRs using load distribution and throughput control.
- Routing application-terminated (AT) SMS traffic coming from RTRs or SMSCs to applications, offering a controlled connect point for SMS applications.
- Controlling application-originated (AO) messages and allowing message traffic to use a configured set of RTRs, SMSCs, or a combination of RTRs and SMSCs or other termination points for AO messages (such as SMS gateways).

The HUB, in conjunction with the RTR, can provide security, scalability and maintainability benefits for SMS networks.

2.2 System Context

The system context diagram below shows the context spaces and their functional interfaces of the HUB. The high-level NewNet Mobile Messaging overview contains Applications that connect to the HUB to send AO messages to and receive AT messages from mobile network components (using SMPP, UCP or CIMD2 over TCP/IP).

The diagram shows the logical system interface context for the HUB (the physical interfaces are described in later chapters).

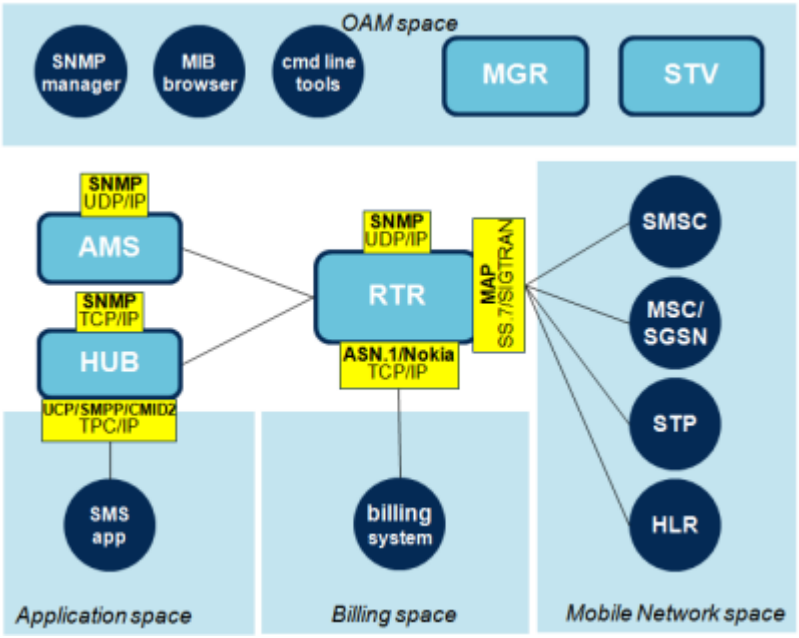


Figure 2: HUB system context

2.2.1 Application Space

The following components may be part of the application space:

- SMS application—Application responsible for receiving the SMS traffic for one short number by using the SMPP, UCP, or CIMD2 protocol over TCP/IP.
- Application Router—Optional concentrator acting as a proxy for all sessions from the SMS applications and handling authentication and session control of all sessions towards the HUB.

Note: Not all components in the data network that are vital for end-to-end SMS traffic (such as IP routers, hubs, switches, and firewalls) are considered relevant in this manual.

2.2.2 Operations, Maintenance and Provisioning Space

This OAM space represents the interface with the operations and maintenance software, including the Manager, Statistics Viewer, and other tools using SNMP. All provisioning commands, management, and alarms pass through this interface.

The following (optional) components may be part of the OAM space:

- SNMP manager—System responsible to capture and process all SNMP alarms (represents the various SNMP-based network management tools that may be available).
- MIB browser—Application that can be used to view (and modify) the contents of the various SNMP variables as defined in the MIB file.
- Command-line tools—Miscellaneous tools and utilities that can be used on the command line of the HUB host machine.
- Manager (MGR)—Web-based management component from the NewNet Mobile Messaging Product Family, used to manage the configuration of all HUBs present in the SMS network configuration.
- Statistics Viewer (ST)—Component from the NewNet Mobile Messaging Product Family used to store, process, and display statistical output of all available SMS Routers, for real-time and archived viewing.

The HUB host has fast Ethernet connections for data traffic (that is, SMS messages encapsulated in SMPP, UCP CIMD2 over TCP/IP packets) and for operations and management traffic (that is, configuration data and SNMP alarms).

2.2.3 Mobile Network Space

The following components may be part of the mobile network space:

- SMSC—Store-and-forward mechanism capable of handling incoming AO messages and outgoing AT messages using the SMPP, UCP or CIMD2 protocol over TCP/IP. Note that SMS gateways can behave similar to SMSCs.
- SMS Router—SMS Router that is configured with a mobile network interface (that is, SS7 or SIGTRAN) that can handle incoming AO messages and outgoing AT messages.

2.3 Rule-Based Operation

The HUB uses the RTR's rule engine and, as such, can route all SMS traffic to many different kinds of destinations. One of the main aspects of the flexible rule engine is the use of the highly configurable routing rules.

The RTR can accommodate the following HUB-related types of rules:

- AO rules—Routing rules, counting rules, and external condition rules related to incoming AO messages.
- AT rules—Routing rules, counting rules, and external condition rules related to outgoing AT messages.

For more information about NewNet Mobile Messaging routing and the use of rules, refer to the RTR Operator Manual.

2.3.1 Software Processes

The HUB engine consists of the Message Processing Unit and its supporting layers for message formatting, load balancing, and throughput regulation.

The HUB process software running on the HUB host server consists of the following executables:

- One instance of the HUB process containing the HUB kernel
- One instance of the watchdog process monitoring the HUB process and restarting and configuring it, if required

The HUB configuration is maintained in the configuration file and the HUB MIB, which can be addressed using SNMP.

Note: The HUB can run in an Application Gateway (AGW) configuration, which requires the RTR to operate as the rule engine for AO and AT traffic.

2.3.2 System Software Components

The HUB assumes that at least one RTR and all RTR tools are installed. All prerequisite system software components that apply for the RTR also apply for the HUB. For more information about prerequisites, refer to the HUB Installation Manual.

2.4 Quality Characteristics

The HUB provides carrier-grade quality behaviour. This quality behaviour is a result of the HUB architecture and design. The most important quality design aspects are:

- High system performance, ensuring efficient use of available resources
- High availability, ensuring maximum service availability without outage
- Scalability, ensuring investment protection and virtually unlimited growth
- Modularity, providing possibility to co-locate other functionality from the NewNet Mobile Messaging Product Family;
- Flexibility, easily adjustable to changing market requirements
- Reliability, ensuring no loss of data, correctness, completeness and consistency
- Security, providing access control, fraud prevention and data protection
- Manageability, providing full system control, alarming and reporting
- Interoperability, providing solutions with different hardware versions of the HUB
- Usability, providing easy to use command line access and GUI access
- Traceability and audit ability, providing possibility to diagnose all system activity
- Accuracy, allowing correct billing and revenue assurance

The quality attributes listed above are partially based on the ISO 9126 Quality attributes. These quality characteristics can be recognized when working with the HUB systems.

2.5 Operator Personnel

The Mobile Messaging system has a number of interfaces to related systems. The different type of interfaces and applications may require different operator personnel to interact with the system. The table summarizes the tasks related to the operator personnel and the type of application or tool used to perform these tasks.

Function	Carried out by...	Using...
Configuring system set-up	System administrator	Command-line tools
Configuring network elements	OAM and planning and design personnel	Command-line tools
Configuring networks, countries, and NewNet Mobile Messaging devices	OAM and planning and design personnel	MGR
Identifying and configuring NewNet Mobile Messaging SNMP alarms	OAM personnel	MIB
Configuring SMS applications	Provisioning personnel	MGR
Configuring routing rules and counting rules	Provisioning personnel	MGR
Activating routing rules and counting rules	Authorized provisioning personnel	MGR
Troubleshooting and tracing	OAM personnel	Operating system command-line and RTR command-line tools
View statistical counters	OAM and planning and design personnel	STV (if available)
View audit logs	System administrator and security personnel	Operating system command-line tools

2.6 Multi-Instance Support

Multi instance feature allows multiple NMM users (up to 10, including the existing `'textpass'` user) be created on the same node, each of whom will be able to run one instance of HUB.

Please note:

1. Multi-instance feature is supported on both RedHat Enterprise Linux and Solaris platforms.
2. A separate LICENSE is required for each NMM user.

Chapter 3

Application Routing

Topics:

3.1 Introduction

There are several ways to route SMS application messages with the HUB.

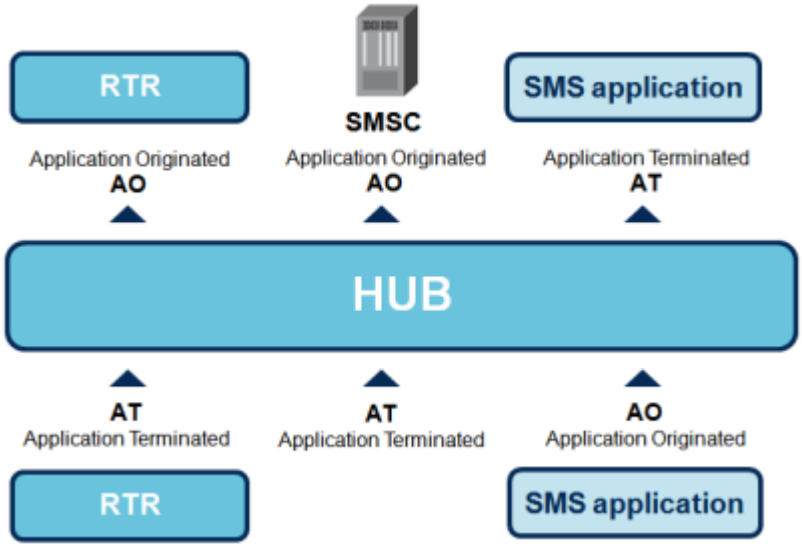


Figure 3: Routing entities and message types

These routing paths can be defined in the HUB by creating the proper application configuration. How to do this is explained in next chapters, this chapter covers the explanation of the general routing concepts of the HUB.

3.1.1 Terminology

The following terminology is important to understand application routing with a HUB:

- Outside session—Connection meant for SMS traffic coming from an SMS application (or ESME) or set up toward an SMS application; SMS applications are typically located outside the mobile network operator domain.
- Inside session—Connection meant for SMS traffic set up toward an SMSC or SMS RTR or coming from an SMSC or SMS RTR; SMSCs and SMS RTRs are typically located inside the mobile network operator domain.

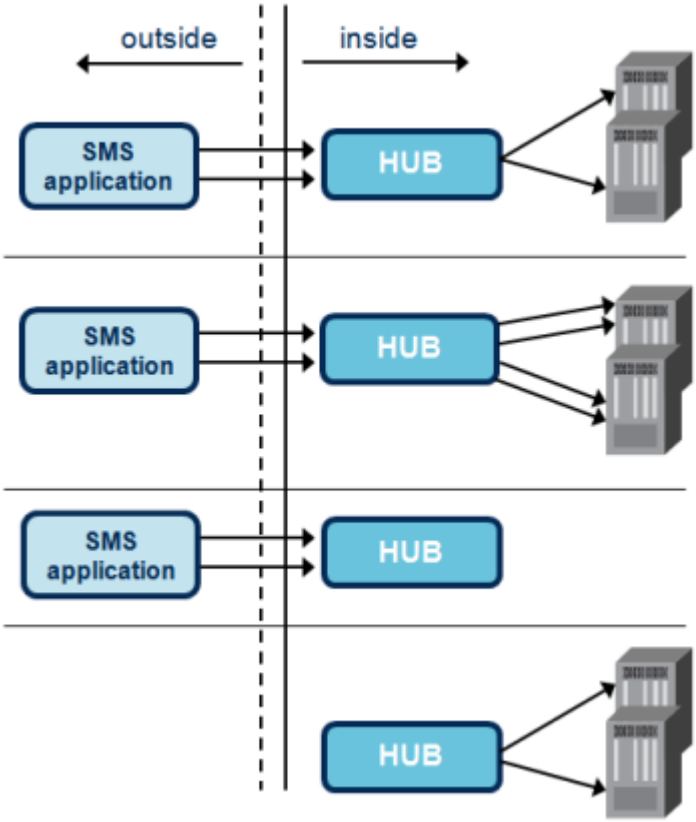


Figure 4: Inside and outside sessions

3.1.2 Application Routing Concepts

The following concepts are important to understand SMS application routing with a HUB:

- Session model—Connection type, which determines how the HUB handles incoming connections. A session model is associated with the outside listen port of the HUB. The session models are:
  - Inside only—Inbound SMS application sessions are not allowed. The HUB will act as a bridge between RTRs and service centers
  - Outside only—Inbound SMS application sessions will only be set up to RTRs. No service centers will be addressable. This session model is (typically used in combination with an AMS.
  - Replicate—Inbound SMS application sessions are replicated to each configured service center. If capacity is not sufficient, then the HUB sets up more inside sessions.
  - Distribute—Inbound SMS application sessions are distributed to each configured service center. If another inside session for this application is connected and enough capacity is available, the HUB does not set up more inside sessions.
- Service class—Specification of a group of service properties associated to an outside listener or an application. If an SMS application connects to a specific outside listener (TCP port), by default, this application will be allowed to use the service properties specified by the service class. If an application can connect to multiple outside listeners, the service class can be overruled on the application level to force the use of one specific service class.

3.1.2.1 Session Functions

The following session functions are available on the HUB:

- Authentication
  - Outside session
    - Password
    - CLI (TCP/IP)
    - Anonymous
  - Inside session
    - Password
- Session timeout
  - Inactivity timeout
- Status tracking with the RTR and other HUBs
  - Keep total count of application session across HUBs
  - RTR can alert AMS application recipient buffer at connection

3.2 CLI and Password Authentication

The HUB supports the following methods for outside session CLI (Calling Line ID) and/or password authentication:

- Password only authentication
- CLI only authentication
- CLI **or** password authentication
- Password **and** CLI authentication

The *CLI only authentication* method and the *CLI **or** password authentication* method both allow access when CLI authentication is successful (a match was found between the connecting application and a configured CLI application). However, both methods require that the source address is unique for each CLI application that uses these authentication methods.

The *password **and** CLI authentication* method does not require that the configured CLI addresses be unique because the HUB authenticates each application when it receives the application's log-in request. However, this method requires that both the password and the CLI authentication match, or the application will not be authenticated.

For each SMS application that is configured in the HUB, you can provide either:

- A single CLI source address
- A list of CLI source addresses

Refer to the MGR Operator Manual for detailed information about configuring authentication.

3.2.1 Troubleshooting CLI Authentication

DNS lookup on Cisco routers can delay the HUB's credential collector, causing CLI-authenticated application log-ins to time out with an "invalid login" error. For best results, disable DNS lookup:

1. Use `ssh` to log in to the Cisco router:

   ```
   $ ssh <user>@<cisco_ip>
   Password:
   ```

   A Cisco prompt appears:

   ```
   cisco>
   ```

2. Enter privileged mode:

   ```
   cisco>enable
   Password:
   ```

   The Cisco prompt changes:

   ```
   cisco#
   ```

3. Enter configuration mode:

   ```
   cisco#configure terminal
   ```

   The Cisco prompt changes:

   ```
   Enter configuration commands, one per line.  End with CNTL/Z.
   cisco(config)#
   ```

4. Disable IP domain lookup:

   ```
   cisco(config)#no ip domain lookup
   ```

5. Exit configuration mode:

   ```
   cisco(config)#end
   ```

6. Verify that the configuration was modified:

   ```
   cisco(config)#sh run
   ```

   Verify that the output contains:

   ```
   !
   no ip domain lookup
   !
   ```

3.3 Session Management

After the HUB accepts an outside session and the session model is *replicate* or *distribute*, the HUB calculates the number of inside sessions that should be set up. The calculation is based on the capacity of all outside sessions per application (capacity is defined as the number of sessions multiplied by the window size).

The HUB ensures that the total capacity available on the application outside sessions is equal to the total capacity available on the application inside sessions:

∑ (number of outside sessions x outside window size) per application = ∑ (number of inside sessions x inside window size) per application

There is a difference between the *distribute* session model and the *replicate* session model here:

- When using the *distribute* session model, the inside sessions are set up individually.
- When using the *replicate* session model, the inside sessions are set up symmetrical to all configured termination points.

Note:

1. The operational state of a service centre, node, or termination point reflects whether the HUB can set up sessions to it. These components may be operational, even if no sessions to them are currently set up.
2. Changes done to the window size to an active application will only be applicable when the application connects with the HUB. If the application is already connected, then changes done to the window size will have no impact on the total capacity of the session, unless and until application re-connects to HUB.
3. Outside total capacity will be calculated based on the following logic:
   a. In case the connection type is transceiver, then the maximum of the outside Transmitting or outside receiving window size will be used for the outside capacity calculation.
   b. If the bind type is Transmitter, then outside receiving window size will be used for the outside total capacity calculation.
   c. If the bind type is Receiver, then outside transmitting window size will be used for the outside total capacity calculation.

3.3.1 Example

One application connects to one HUB with a *replicate* session model to connect to five configured SMSCs. The outside window size and inside window size are both set to three. The result of this configuration for this application is:

- 1 outside session results in 5 inside sessions.
- 2 outside session results in 5 inside sessions.
- 3 outside session results in 5 inside sessions.
- 4 outside session results in 5 inside sessions.
- 5 outside session results in 5 inside sessions.
- 6 outside session results in 10 inside sessions.
- 7 outside session results in 10 inside sessions.

Therefore, for *replicate*, the symmetrical session set up will result in a step-by-step capacity increase of inside sessions.

3.3.2 Send Short Ack

If an application is configured for the Send Short Ack functionality, the AGW will establish an outside session with the application by sending an Ack. to its login/bind request as soon as any one of the SMSC(IP) has successfully set up the first inside session with the AGW. This will minimize the chances of the application timing out on a login/bind request and will allow the application to start sending traffic immediately to the AGW.

This functionality is supported by SMPP, UCP and CIMD2 protocol interfaces and it is applicable only for the following session models:

- Replicate - SC List
- Replicate - All SCs
- Distribute - SC List
- Distribute - All SCs

In case the application is configured for using the session model associated with the Service Class corresponding to the outside listener, then this functionality will be applicable only if the relevant session model is either ' Replicate - All SCs' or 'Distribute - All SCs'.

The Send Short Ack. functionality is controlled by a dynamic configuration parameter on a per-application basis, and by default it is disabled.

To enable the functionality for an application:

1. In the MGR, go to **SMS Applications ➤ Applications**.
2. Select the desired application
3. Select the Send Short Ack

Note: Use caution when enabling this parameter, because there will be a certain risk of messages being dropped by the HUB during the time-window between the outside session getting established (i.e. the application receiving the 'Short Ack' and starting to send traffic) and sufficient inside session capacity getting secured on the HUB-SMSC(IP) interfaces. The probability of messages being dropped will increase proportionately with the number of concurrent (near-simultaneous) outside sessions from a given application for which the HUB sends 'Short Ack'.

3.4 Dialout Applications

The HUB can set up a dialout connection to external applications that use the SMPP or UCP protocol. Dialout is used for:

- SMPP DELIVER_SM (AT)
- UCP 52 (deliver short message)
- UCP 53 (deliver notification)

Dialout enables the Application Gateway (AGW) to route MO-AT traffic from an SMSC to an application. Only one notification address is possible per application. This functionality requires an *inside only* application.

Refer to the MGR Operator Manual for more information about the parameters that are discussed in this section.

3.4.1 General Configuration

To implement this functionality, configure the following settings in the Manager interface, in **SMS Applications ➤ Application**:

- Outside dialout type—IP
- Outside dialout IP address—IP address to use for dialout
- Outside dialout TCP port—TCP port to use for dialout
- AT enabled—Selected

Then, in **Environment ➤ Service Class**, configure the following settings:

- Session model—Inside only - All SCs
- Dialout allowed—Selected
- Allow traffic—AO or AOAT

Note: In case of dialout applications, the IP address configured can be of IPv4 type only.

3.4.2 SMPP Applications

For SMPP dialout, also configure the following settings in **SMS Applications ➤ Application**:

- Outside SMPP password
- Dialout inactivity time
- Dialout maximum sessions
- Dialout SMPP window size

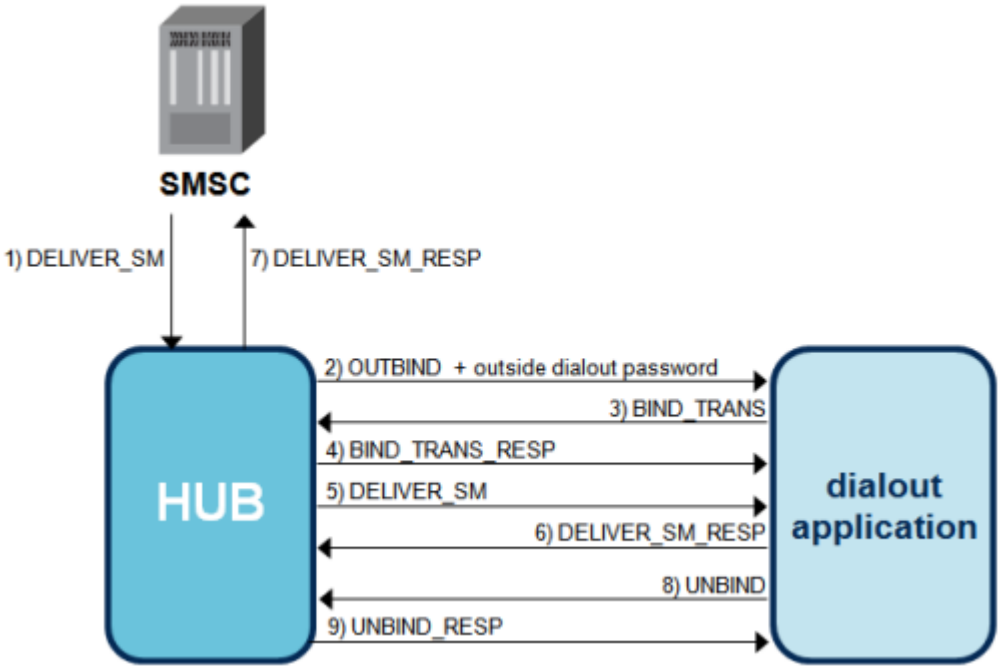


Figure 5: SMPP dialout message flow

The HUB can send an OUTBIND with an outside dialout password to the application. The functionality expects that the dialout session is established via:

- OUTBIND
- BIND_TRANSCEIVER or BIND_RECEIVER
- BIND_TRANSCEIVER_RESP or BIND_RECEIVER_RESP

When the dialout session is established, the HUB sends DELIVER_SM messages to the dialout application. Session closure is triggered by the UNBIND, and is acknowledged with an UNBIND_RESP message.

3.4.3 UCP Applications

For UCP dialout, also configure the following settings in **SMS Applications ➤ Application**:

- Dialout maximum inactivity time
- Dialout maximum sessions
- Dialout UCP window size

The dialout session is immediately set for the delivery of UCP 52 messages.

3.4.3.1 UCP Dialout Notifications

Multiple notification addresses can be used in UCP AO messages that request notification.

Notifications that come from an SMSC for earlier AO messages with individual notification addresses should be sent by the SMSC (not by the AGW).

The dialout application is configured via the notification address NADC (IP+port) in the UCP 51 message.

When the notification address is set, the following semi-static parameters determine the AMS queue in which to store the notifications:

- `amsqueuefordialoutatstatusreports`
- `enabledialoutnotificationasamsmessagetype`

Refer to the AMS Operator Manual for more information about these parameters.

In the case of an AGW with a CMSC, the CSMSC is responsible for sending notifications (UCP 53) to the multiple notification addresses.

3.4.4 Configurable Session Thresholds

The HUB includes alarms (traps) that notify the operator when the total number of inside or outside sessions falls below a configurable threshold limit per application. The threshold limits for inside and outside sessions are configured separately.

3.4.4.1 Requirements

The HUB will only send minimum session traps when the following conditions are met:

- The device state is operating
- The application operational state is operating
- The threshold limit is not set to 0 (zero)

3.4.4.2 Configuring the Minimum Number of Sessions

Use the MGR to set the minimum number of inside sessions and the minimum number of outside sessions per application.

3.4.4.3 Configuring the Maximum Number of Sessions

The threshold limit is expressed as a percentage of the maximum number of sessions, which is configurable:

- For **inside** sessions, the `applicationNrSmscOnlySessions` property determines the maximum number of sessions.
- For **outside** sessions, the property that determines the maximum number of sessions depends on the session type:

- UCP—`applicationOutsideUcpMaxSessions`
- CIMD—`applicationOutsideCimdMaxSessions`
- SMPP RX—`applicationOutsideSmppMaxReceivers`
- SMPP TX—`applicationOutsideSmppMaxTransmitters`
- SMPP TRX—`applicationOutsideSmppMaxTransceivers`

3.4.4.4 Alarm State

When the current number of sessions is greater than or equal to the threshold limit, the alarm state is "reached". When the current number of sessions is less than the threshold limit, the alarm state is "not reached". The initial alarm state is "reached".

The HUB sends traps when the alarm state changes from one state to the other. The traps contain the:

- Application name
- Number of sessions
- Device type

3.4.4.5 Trap Types

The trap depends on the session type, as follows:

Protocol	Session Type	Alarm State	Trap
UCP	Inside	Not Reached	applicationInsideUcpMinimumSessionsNotReached
UCP	Inside	Reached	applicationInsideUcpMinimumSessionsReached
UCP	Outside	Not Reached	applicationOutsideUcpMinimumSessionsNotReached
UCP	Outside	Reached	applicationOutsideUcpMinimumSessionsReached
CIMD	Inside	Not Reached	applicationInsideCimdMinimumSessionsNotReached
CIMD	Inside	Reached	applicationInsideCimdMinimumSessionsReached
CIMD	Outside	Not Reached	applicationOutsideCimdMinimumSessionsNotReached
CIMD	Outside	Reached	applicationOutsideCimdMinimumSessionsReached
SMPP RX	Inside	Not Reached	applicationInsideSmppRxMinimumSessionsNotReached
SMPP RX	Inside	Reached	applicationInsideSmppRxMinimumSessionsReached
SMPP RX	Outside	Not Reached	applicationOutsideSmppRxMinimumSessionsNotReached
SMPP RX	Outside	Reached	applicationOutsideSmppRxMinimumSessionsReached
SMPP TX	Inside	Not Reached	applicationInsideSmppTxMinimumSessionsNotReached
SMPP TX	Inside	Reached	applicationInsideSmppTxMinimumSessionsReached
SMPP TX	Outside	Not Reached	applicationOutsideSmppTxMinimumSessionsNotReached
SMPP TX	Outside	Reached	applicationOutsideSmppTxMinimumSessionsReached
SMPP TRX	Inside	Not Reached	applicationInsideSmppTrxMinimumSessionsNotReached

Protocol	Session Type	Alarm State	Trap
SMPP TRX	Inside	Reached	applicationInsideSmppTrxMinimumSessionsReached
SMPP TRX	Outside	Not Reached	applicationOutsideSmppTrxMinimumSessionsNotReached
SMPP TRX	Outside	Reached	applicationOutsideSmppTrxMinimumSessionsReached

3.4.4.6 Disabling the Minimum Sessions Alarm

To disable the minimum sessions alarm functionality, set the threshold limit to 0 (zero).

3.4.5 Time-outs

The HUB actively manages inside sessions and will set up a session to SMSCs and/or RTRs on behalf of SMS applications by issuing UCP31 as keep-alive messages.

If more time-outs than configured occur on an inside session (such as due to a "hanging" session), the HUB disconnects the inside session and attempts to re-establish it.

The semi-static configuration parameters for time-outs are:

- `hubcloseoutsideafterxtimeouts` for **outside** sessions
- `hubcloseinsideafterxtimeouts` for **inside** sessions

The action that results from exceeding the time-out parameter(s) depend on the specific circumstances:

- In case of an **inside** hanging session:
  - If the time-out pertains to the last inside session, the HUB will also close the corresponding outside session;
  - If there are more inside sessions left, the corresponding outside sessions will remain open and the HUB will try to establish a new inside session.
- In case of an **outside** hanging session:
  - The HUB will recalculate the required number of inside sessions;
  - If applicable, the surplus of inside sessions will be closed.

Note: The default configuration causes the HUB to not disconnect due to detected time-outs.

Refer to the AMS Operator Manual for more information about the role of storage in routing.

Refer to the RTR Operator Manual for more information about application-related messaging originated from the RTR.

3.5 TCP Keep-Alive Functionality

When an authenticated TCP/IP application session disappears without sending the appropriate TCP disconnect to the HUB, the session remains open on the HUB (as the HUB is not informed of any change). The HUB will only disconnect the session when the inactivity timeout period expires. This

can result in situations in which a client that unexpectedly restarted cannot reconnect to the HUB because the HUB views the previously disconnected session as still being open.

TCP keep-alive (`SO_KEEPALIVE`) functionality can enable the HUB to differentiate between a disconnected session and an application that is slow to respond. TCP keep-alive sends a "life check" message to determine whether a connected TCP socket is still connected or if the connection is broken.

Note: The HUB sends TCP keep-alive packets every x seconds after the prior TCP packet, where x is the TCP keep-alive interval configured in the operating system. Therefore, if there are other TCP packets (such as ACKs or data packets) being sent, then they will impact the exact time that the HUB sends the TCP keep-alive packet.

3.5.1 TCP Keep-Alive Handling

TCP keep-alive handling can be applied to the following types of sessions:

- Outside sessions of applications for which the functionality is enabled
- Inside sessions of applications for which the functionality is enabled
- Inside sessions toward service centre termination points for which the functionality is enabled

The TCP keep-alive functionality operates on an "or" condition. When you enable TCP keep-alive handling for an application, the socket that the HUB uses for the connection will use keep-alive, even if the inside sessions are started toward a termination point on which keep-alive is disabled. When you enable TCP keep-alive handling for a termination point, the inside sessions that are started toward it will use keep-alive, even if the sessions are for an application on which keep-alive is disabled.

3.5.2 Configuring TCP Keep-Alive

To configure TCP keep-alive:

1. In the semi-static configuration file, set the `tpconfig` attribute `hubenabletcpkeepalive` to "true".
2. In the MGR interface, enable the TCP keep-alive option for each application and termination point for which the HUB should perform TCP keep-alive checks.

   Refer to the MGR Operator Manual for more information.

Enabling TCP keep-alive functionality only impacts newly created sessions; existing sessions are not impacted.

3.5.3 Configuring Solaris 10

In Solaris 10, the following parameters control TCP keep-alive:

Parameter	Description	Range	Default
tcp_keepalive_interval	The time (in milliseconds) after which the first keep-alive probe is sent to an endpoint.	10 seconds - 10 days	2 hours
tcp_ip_abort_interval	The total transmission timeout (in milliseconds) for a TCP connection. If TCP has	500 milliseconds - 1193 hours	8 minutes

Parameter	Description	Range	Default
	been re-transmitting to an endpoint for this period of time and has not received an acknowledgment from the other endpoint, TCP closes the connection to that endpoint.		

For more information about these parameters, refer to the Solaris Tunable Parameters Reference Manual at *http://download.oracle.com/docs/cd/E19253-01/817-0404/*.

Note: Use caution when changing these parameters, as they can have a significant impact on traffic and binds in the system.

3.5.3.1 Setting Parameters in the System File

Set the parameters in the `etc/system` file of each system on which you want to adjust TCP keep-alive. You must reboot the system for the settings to take effect.

3.5.3.2 Setting Parameters on the Fly

You can set the parameters on the fly. For example, to set the keep-alive interval to 50 seconds:

```
ndd -set /dev/tcp tcp_keepalive_interval 50000
```

To set the TCP/IP abort interval to 10 seconds:

```
ndd -set /dev/tcp tcp_ip_abort_interval 10000
```

Note: If you set parameters on the fly, they will be reset to their default values upon the next reboot.

3.5.4 Configuring Red Hat Enterprise Linux 6

In Red Hat Enterprise Linux 7.3 (RHEL7), the following parameters control TCP keep-alive:

Parameter	Description	Default
tcp_keepalive_time	The time after which the first keep-alive probe is sent to an endpoint.	2 hours
tcp_keepalive_intvl	The time between TCP keep-alive probes.	75 seconds
tcp_keepalive_probes	The maximum number of probes to send to an endpoint before considering it unavailable and closing the connection to the endpoint.	9

For more information about these parameters, refer to the `TCP(7)` man page by executing:

```
man -s 7 tcp
```

Note: Use caution when changing these parameters, as they can have a significant impact on traffic and binds in the system.

3.5.4.1 Setting Parameters in the System File

Set the parameters in the `/etc/sysctl.conf` file of each system on which you want to adjust TCP keep-alive. For example:

```
# Enable TCP keep-alive and set them to 10 minutes
net.ipv4.tcp_keepalive_time = 600
# Set the probe interval to 6 seconds
net.ipv4.tcp_keepalive_intvl = 6
# Set the number of probes to 3
net.ipv4.tcp_keepalive_probes = 3
```

3.5.4.2 Setting Parameters on the Fly

You can set the parameters on the fly using the echo command:

```
echo "xxxx" > /proc/sys/net/ipv4/tcp_keepalive_intvl
echo "xxxx" > /proc/sys/net/ipv4/tcp_keepalive_probes
echo "xxxx" > /proc/sys/net/ipv4/tcp_keepalive_time
```

Where `xxxx` is the value that you want to set for the parameter. Set each parameter on each system on which you want to adjust TCP keep-alive.

Note: If you set parameters on the fly, they will be reset to their default values upon the next reboot.

3.5.4.3 Viewing Current Parameter Values

You can view the current parameter values using the `cat` command:

```
cat /proc/sys/net/ipv4/tcp_keepalive_time
cat /proc/sys/net/ipv4/tcp_keepalive_intvl
cat /proc/sys/net/ipv4/tcp_keepalive_probes
```

3.6 Routing Paths Overview

The routing paths defined in the HUB handle all incoming application traffic (AO and AT). Routing AO or AT traffic through the HUB achieves:

- Traffic distribution and load balancing over RTRs and SMSCs.
- Throughput regulation of AO traffic.
- Optimized SMS delivery, while maintaining quality of service, using the RTR for the first delivery.

3.6.1 HUB Role in MO Routing

The HUB also plays an important role in AO and AT routing for MO messages coming from the RTR. The routing paths that apply to messages that are routed by the HUB, in conjunction with the RTR, are:

- MO-AT routing (direct delivery to an application)
- MO-AO routing (forward MO message as an AO message to an SMSC or SMS Gateway)
- MO-MT-AO routing (forward MO message, after a failed delivery attempt, as an AO message to an SMSC or SMS Gateway)

3.7 AO Routing

Application-originating (AO) routing is the generic name for the processing of incoming AO messages. The HUB works in conjunction with the RTR and, in the case of routing with storage, the AMS to route AO messages to their destinations.

3.7.1 AO Routing Paths

Supported AO routing paths are:

- AO-MT (route to mobile station)

  Note: AO-MT messages are subject to AO and MTO routing rules.

- AO-AT (route to application)
- AO-AO (route to SMSC)
- AO-MT-AO (route to mobile station, fallback to SMSC)
- AO-MT-Store (route to mobile station, fallback to storage)
- AO-AT-Store (route to application, fallback to storage)
- AO-AO-Store (route to SMSC, fallback to storage)
- AO-Store-MT (store for delivery to mobile station)
- AO-Store-AT (store for delivery to application)
- AO-Store-AO (store for forwarding as AO)
- AO-Discard:
  - Discard with NACK
  - Discard with ACK

3.7.2 AO-MT, AO-MT-Store and AO-Store-MT Routing

The HUB can support AO-MT routing (route to mobile station) . The HUB provides the functionality to process incoming AO messages and the RTR performs an optimised MT delivery. AO-MT routing provides the following features:

- Optimised direct delivery of MT to mobile
- Throughput control
- Notification generation, if configured
- CDR generation, if configured

Figure 6: AO-MT routing

AO-MT routing can provide SMSC offload and SMSC overload protection. The direct MT delivery of AO-MT routing can use optimised routing on a per-application basis.

When the Mobile Messaging system includes the AMS, the HUB can provide AO-MT-Store (route to mobile station, fallback to storage) and/or AO-Store-MT (store for delivery to mobile station) routing.

Note: The AO-MT messages use the SMSC address that is configured in the `commonaddress` attribute as the MAP SMSC address in the MT message.

Important: In case of AO-MT-Store (route to mobile station, fallback to storage), the message is sent by the RTR to Store (AMS) before even making a FDA to the MS. Here the message is not stored in the AMS database but it is placed in the appropriate message queue; now the AMS immediately sends an Ack. to the RTR and then sends the message back to the RTR for performing the FDA. If the FDA is successful, the RTR generates a billing record (CDR); otherwise the routing result is returned to the AMS for storing it in the database for later retries. In the AO routing rule counters on the RTR, when the message is sent to the AMS, it is always counted as the primary destination and never as a fallback.

Note: In the AO-MT scenario, if early HLR query for AO/SM is configured (see **Routing-➤ Properties Early SRI-SM for AO/SM Whitelist** and **Early SRI-SM for AO/SM** in MGR), then during AO Rule evaluation, early SRISM is sent irrespective of Recipient Mobile Network Domain. MT delivery is also attempted in SS7 domain irrespective of Recipient Mobile Network Domain.

Note: The AO Rule condition (**Terminating MSC/SGSN** [cond]) is evaluated against the MSC or SGSN address as returned by the HLR. If the HLR returns both addresses, the rule set is evaluated against either the MSC or the SGSN address, as selected by the semi-static attribute `preferredmtdestination`. If the Network configuration is available according to MSC and/or SGSN (i.e. received in the HLR query), the **Preferred MT Destination** in the Network configuration overrides the semi static attribute `preferredmtdestination` for the rules evaluation.

Note: During the AT delivery via store, the configured value of the "outside transmit window size" is used as the `maxtransactions` in the store request. For the AT delivery "outside transmit window size" of the configured application is used for load balancing the traffic towards the HUB.

Important: In case of AO-MT, AO-MT-ST and AO-ST-MT billing if Source address of AO message is an alphanumeric address and reflecting the service provider name or address then the content vendors should send an SMPP message where an optional field TLV should reflect the real short code. The SMSC should retrieve this field and put it in the FCDR's OrigAddress fields.

The RTR will receive an `originatorAddressForBillingOverride` in receivedSubmitSmRequest from HUB over the MXP interface and will retrieve the short code belonging to the optional TLV in range 0x1400 to 0x3fff (5120 to 16383).

The following sub-fields will be affected in the FCDR's OrigAddress:

- `ton` will be short (4)
- `npi` will be private (5)
- `msisdn` will be an integer value (up to 9 digit) received in TLV 0x1403
- `msisdnUTF8` will be an integer value (up to 9 digit) received in TLV 0x1403.

Note:

1. It should be applicable for the submitted, rejected, Delivery FCDRs,
2. there should not be any impact on Routing rules, log and events.
3. Following routing path can be impacted:
   a. AO-MT
   b. AO-Store-MT
   c. AO-MT-Store

3.7.2.1 Setting SRI-SM Priority for Application

SRI-SM Priority configuration for an Application controls the setting of the priority field in the SRI-SM Request (to be sent by the RTR) for messages originated from the application, prior to their MT delivery attempts.. If this parameter set to 'High' or 'Low', the value of the sm-RP-PRI in the SRI-SM Request will be set to 'TRUE' (1) or 'FALSE' (0), respectively. In case of 'Use message priority', the value of the sm-RP-PRI will be set as per the value of the Priority field in individual messages, if that field is present; in case no Priority field is present in a particular AO message, the value of the sm-RP-PRI in the SRI-SM will be set to 'FALSE'.

By default this parameter is configured as 'High'.

Note: Although this parameter is relevant for AO-MT, AO-MT-ST and AO-ST-MT routing, in case a message gets stored in the AMS and retried later (i.e. after a failed FDA) then this SRI-SM Priority setting may get superseded by the corresponding configuration setting for the AMS queue; refer to the AMS Operator Manual for more details.

In order to configure the **SRI-SM Priority** setting in the Manager interface:

1. Go to **SMS Applications ➤ Application**.
2. Select the desired application.
3. Select the desired value for **SRI-SM Priority**.

3.7.3 AO-AT, AO-AT-Store, and AO-Store-AT Routing

With a license upgrade, the HUB can support AO-AT routing. The HUB provides the functionality to process incoming AO messages and directly perform an AT delivery. AO-AT routing provides the following features:

- Throughput control
- Notification generation, if configured
- CDR generation, if configured

Figure 7: AO-AT routing

When the Mobile Messaging system includes the AMS, the HUB can provide AO-AT-Store (route to application, fallback to storage) and/or AO-Store-AT (store for delivery to application) routing.

For AO-AT routing and AO-Store-AT routing, the HUB supports protocol conversion from UCP to SMPP.

Important: In case of AO-MT, AO-MT-ST and AO-ST-MT billing if the source address of AO message is an alphanumeric address and reflecting the service provider name or address, then the content vendors should send an SMPP message where an optional field TLV should reflect the real short code. The SMSC should retrieve this field and put it in the FCDR's OrigAddress fields.

The following sub-fields will be affected in the FCDR's OrigAddress fields when MGR's **Outside SMPP Originator Addr Tag** is provisioned with a valid value above 0x13ff (0x1400 to 0x3fff) and the

received submit_sm or data_sm message contains the optional TLV with a TAG value same as **Outside SMPP Originator Addr Tag**:

- `ton` will be short (4)
- `npi` will be private (5)
- `msisdn` will be an integer value (up to 9 digit) received in TLV 0x1403
- `msisdnUTF8` will be an integer value (up to 9 digit) received in TLV 0x1403.

When HUB receives message with a configured Tag that is not in the range 0x1400 to 0x3fff, then the HUB should not encode the new optional MXP parameter `originatorAddressForBillingOverride` in receivedSubmitSmRequest.

Hub should follow the existing behavior. As per current behavior, HUB parses few TLVs (such as message_payload, qos_time_to_live, more_messages_to_send, etc) and use their value to encode different fields of Message context. The TLV fields that HUB does not parse, are copied directly to "smppTlvs" message field in receivedSubmitSmRequest.

Example: When MGR's **Outside SMPP Originator Addr Tag** is provisioned with a decimal value of 0x1403 and the received submit_sm or data_sm message contains the optional TLV 0x1403, the HUB should retrieve the short code value from submit_sm or data_sm message. Refer the following table for optional TLV 0x1403 description.

Field	Size in octets	Type	Description
Parameter Tag	2	Integer	value equal to **Outside SMPP Originator Addr Tag**, present in submit_sm or data_sm. The most likely value is 0x1403.
Length	2	Integer	Length of the value.
Value	4	Integer	Unsigned value up to 9 digits long.

A new field, `originatorAddressForBillingOverride`, was introduced in the MXP interface to send the short code value received in the optional TLV to RTR for FCDR processing.

If the value received in TLV with a tag in the range 0x1400 (5120) - 0x3fff (16383) is not an integer, then the value received in that TLV would be ignored and proceed with existing behavior. The maximum value received in the TLV should be 99999999.

Note:

1. Above SMPP TLV option is applicable for version 3.4 or above.
2. In MGR, the corresponding decimal format of hex-value for **Outside SMPP Originator Addr. Tag** should be provisioned.
3. If HUB receives a message with a configured tag that does not lie in the range of 0x1400 (5120) - 0x3fff (16383), then HUB shall not encode the new optional MXP parameter `originatorAddressForBillingOverride` and shall follow the existing behavior.
4. If HUB receives a message with a configured tag that lies in the range of 0x1400 (5120) - 0x3fff (16383) [including 0x1400 and 0x3fff], the SMSC rejects the messages and logs the error for TLV in `/var/log/messages`, in the following conditions:
   a. When the SMPP TLV tag will be received for SMPP 3.3.
   b. When the SMPP TLV value received in the TLV will be more than 9 digits.

3.7.4 AO-AO, AO-AO-Store, and AO-Store-AO Routing

The AO-AO path (route to SMSC as AO) is mainly used to monitor and control incoming application traffic from applications. The HUB offers a single access point that handles routing and session load-balancing toward to all connected RTRs and SMSCs. The load balancing algorithm attempts to relay AO messages for the same recipient to the same SMSC. The HUB also supports throughput control to block or throttle AO traffic.

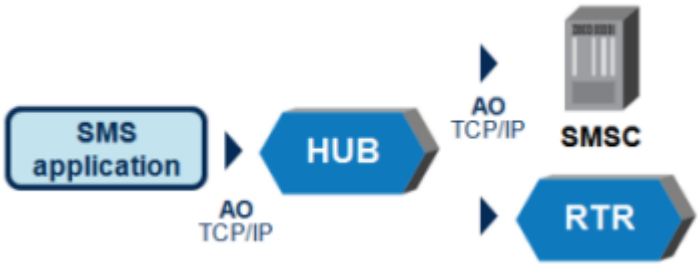


Figure 8: AO-AO routing

When the Mobile Messaging system includes the AMS, the HUB can provide AO-AO-Store (route to SMSC, fallback to storage) and/or AO-Store-AO (store for forwarding as AO) routing.

3.7.4.1 Example Detailed AO-AO Message Flow

In this example, two types of SMS services are defined with different quality of service (QoS) levels:

- Normal SMS traffic forwarded to the SMSC (QoS: Bulk SMS)
- Premium SMS traffic, forwarded to a premium SMS gateway (QoS: Premium SMS)

A typical message flow for AO-AO routing in this example is:

1. The HUB accepts a connect request from a Bulk SMSapplication on one of the configured AO-AO ports for Bulk SMS.
2. The HUB receives a UCP60 log-in request from the SMS application on the established session.

   To verify if the UCP60 log-in is valid, the HUB:

   a) Establishes a connection to the first specified SMSC listener
   b) Logs in to the SMSC on behalf of the SMS application (that is, forwards the UCP60 transparently to the SMSC)
   c) Receives the response on the UCP60 from the SMSC
   d) If the UCP60 response is positive, dynamically configure the SMS application in the HUB
   e) Forward the UCP60 response to the SMS application.

   Note: This sequence is repeated for all sessions that the application opens.

3. The HUB receives a UCP51 submit message from the SMS application.
4. The HUB screens the properties of the UCP51 message to determine one of the following required actions:

   a) If the SMSC listener has bandwidth available in the second in which the UCP51 is received, the UCP51 is forwarded to the SMSC listener
   b) If the UCP51 contains a destination number that matches a defined country or network number range, the appropriate counters are increased

c) The general UCP51 request counter is increased
d) After the ACK on the UCP51 is received, the ACK is forwarded to the application.

Note: If the received UCP51 exceeds the specified bandwidth, the message is temporarily buffered until the bandwidth is available or until a time-out occurs.

5. If the HUB receives a valid UCP message (not UCP60 or UCP51), the HUB immediately forwards the message transparently to the SMSC.
6. If the HUB receives a disconnect request from the SMS application, it disconnects the corresponding session to the SMSC and the RTR.

3.7.5 AO-MT-AO Routing

AO-MT-AO routing provides the functionality to process incoming AO messages where the HUB will route the AO message to the RTR for a first delivery attempt (FDA). The HUB will only forward the AO message to an SMSC if the FDA fails. AO-MT-AO routing provides the following features:

- Traffic distribution and load balancing traffic over RTRs and SMSCs
- Throughput regulation of the AO traffic
- Optimised direct delivery of MT to mobile
- Throughput control
- Notification generation, if delivered successfully and if configured
- CDR generation, if configured
- Forward to SMSC in case of an unsuccessful delivery attempt

To relieve the SMSC of excessive application traffic, the HUB can also provide the first delivery attempt (MT) for application-originated (AO) traffic.

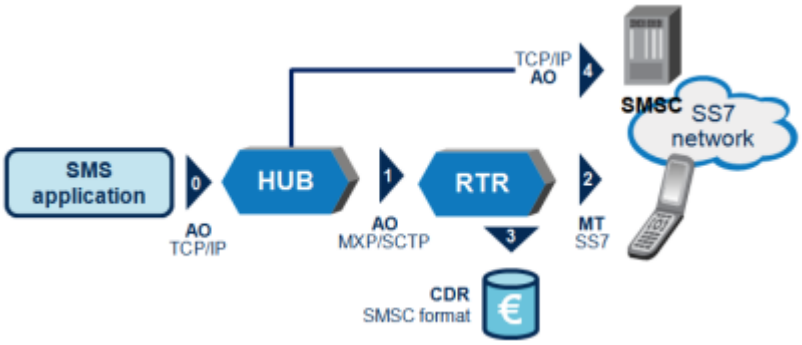


Figure 9: AO-MT-AO routing

The application submits an AO message (via SMPP, UCP, or CIMD2) to the HUB (flow 0). When the routing path AO-MT-AO is defined for the incoming SMS message, the HUB routes the AO message to the RTR (flow 1). Immediately, the RTR performs one delivery attempt to the recipient (flow 2), which in this example is a mobile phone. If the delivery attempt is successful, a billing record is generated (flow 3); if the delivery failed, the message is forwarded as an AO submit message to the appropriate SMSC (flow 4) for further delivery. Now the SMSC will take care of the delivery (retries) of the message and will generate a billing record when the message is delivered.

3.7.5.1 AO-MT-AO Message Flow

A typical message flow for regular AO-MT-AO routing is:

1. The HUB accepts a connect request from an SMS application on one of the configured AO-MT-AO ports.
2. The HUB receives a UCP60 log-in request from the SMS application on the established session.

   If the UCP60 log-in is valid, the HUB:

   a) Forwards the log-in request to the RTR configured
   b) Establishes a connection to the SMSC configured in the service class
   c) Logs in to the SMSC on behalf of the SMS application
   d) Responds to the UCP60 log-in request to the SMS application
   e) Logs in to the RTR on behalf of the SMS application
3. The HUB receives a UCP51 submit message from the SMS application.
4. The HUB investigates the properties of the UCP51 message to determine one of the following required actions:

   a) If the message can be delivered by the RTR, the HUB forwards the message to the RTR
   b) If the RTR's delivery attempt is not successful, the message is forwarded to the SMSC for further retries
5. If the HUB receives a valid UCP message (not UCP60, UCP51, UCP54), the HUB forwards the message to the SMSC.
6. If the HUB receives a disconnect request from the SMS application, it disconnects the corresponding session to the SMSC and forwards the disconnect request to the RTR.

The HUB actively manages the session set up to the SMSCs on behalf of the SMS applications by issuing UCP31 as keep-alive messages.

Note: Forwarding to the SMSC requires that the system wait on the response from the SMSC to return a response to the application, due to the identifier that the SMSC assigns.

3.7.5.2 AO-MT-AO Failure Scenarios

This section describes some AO-MT-AO failure scenarios.

Scenario	Result
There is not an SMSC connection when the message is submitted (UCP51 message is received)	The AO-MT-AO rule will not match the message.
All SMSC connections are overloaded (throughput limitation or window size is exceeded)	The AO-MT-AO rule will not match the message.
When the fallback AO message is submitted, the SMSC returns an error	If an acknowledgement has already been returned to the application, the Mobile Messaging system can only notify the application of this failure if it requested a delivery notification. The delivery status in the notification would indicate failure. If the application did not request delivery notification and logging, the log record will show that the message was dropped. However, the application sees the message as having been delivered successfully.

Scenario	Result
The SRI-SM or MT operation performed by the RTR returns a permanent error	If an acknowledgement has already been returned to the application, the Mobile Messaging system can only notify the application of this failure if it requested a delivery notification. The delivery status in the notification would indicate failure. If the application did not request delivery notification and logging, the log record will show that the message was dropped. However, the application sees the message as having been delivered successfully.
The connection to the SMSC fails after the RTR performs a delivery attempt (MtFwdSm)	If an acknowledgement has already been returned to the application, the Mobile Messaging system can only notify the application of this failure if it requested a delivery notification. The delivery status in the notification would indicate failure. If the application did not request delivery notification and logging, the log record will show that the message was dropped. However, the application sees the message as having been delivered successfully.

3.7.5.3 Configuring the Application Entity for AO-MT-AO Routing

In the case of AO-MT-AO routing, RTR routing properties must be adapted determine when to generate the ACK/NACK response for a submit operation. These properties can be configured on an individual RTR application basis.

In the MGR interface, the SMS application entity has the following attributes:

- **AO-MT-AO Enabled**—When selected, the RTR always generate an ACK/NACK response that reflects the result of the delivery attempt. (This functionality requires the license for the AO-MT-AO routing path.)
- **Respond After Delivery**—When selected, the RTR will generate an ACK/NACK response that reflects the result of the delivery attempt. This attribute is mandatory for AO-MT-AO routing, but it can also be applied to the AO-MT path.

3.7.6 MO-AO Routing

The HUB, in conjunction with the RTR, can support MO-AO routing (*route to* SMSC *Group as* AO or *route to SMSC Application as AO* option in the Manager). In this routing path, mobile-originated (MO) messages are forwarded as application-originated (AO) messages to a service centre or to a message gateway that accepts AO messages.

This functionality can be useful when:

- The SS7 connection to the SMSC should be spared
- There is no SS7 connection to the SMSC or message gateway

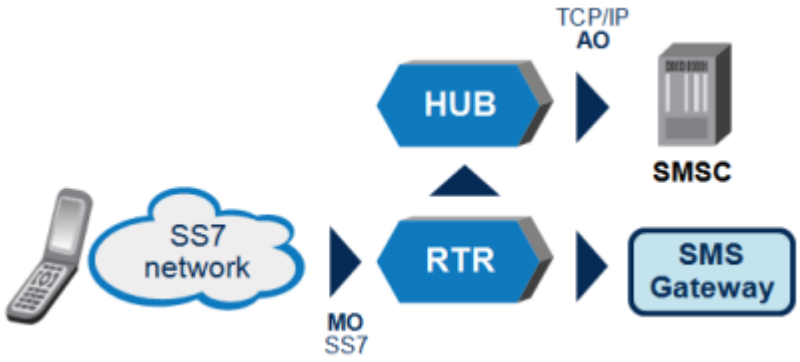


Figure 10: MO-AO routing

When applying MO-AO routing, the system (RTR and HUB) mimics the behaviour of an application toward the receiver of the AO message: the service centre sees the system as an application issuing short messages.

On the system, a dedicated application must be configured for this purpose, and the MO-AO routing rule must refer to that application. The session model of this application must be *insideOnlyAllScs* or *insideOnlyScList*.

The system only sends an acknowledgement to the MO message originator only after the system knows the result of the AO. Therefore, if the service centre or message gateway rejected the message, the mobile originator will see "message not sent" and the RTR will not generate a CDR.

If the AO message is accepted, CDRs for the submission to the RTR and for the successful delivery to the service centre or message gateway are generated (depending on the rule's provisioned parameters).

3.7.7 MO-MT-AO Routing

The HUB, in conjunction with the RTR, can support MO-MT-AO routing (*route to* MS *fallback to SMSC Group as AO* or *route to MS fallback to SMSC Application as AO* option in the Manager). In this routing path, the SMS RTR executes the first delivery attempt (FDA).

If this delivery attempt fails, the HUB enables routing the message to the SMSC as an application-originated (AO) message.

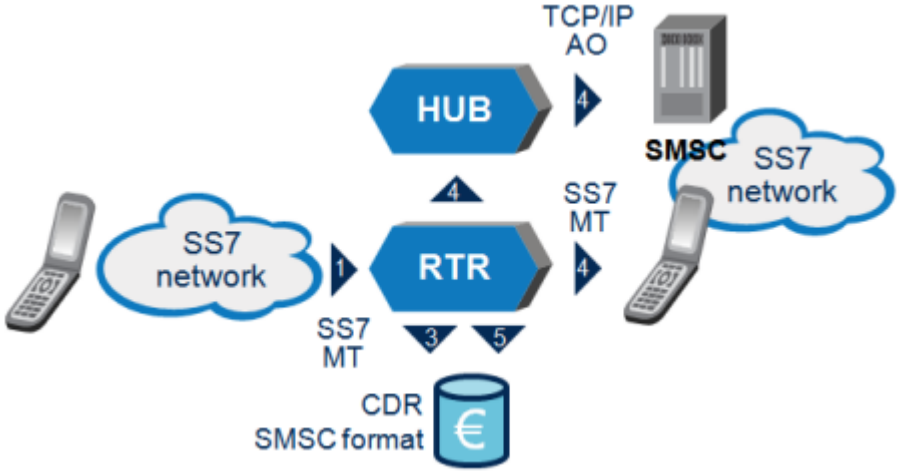


Figure 11: MO-MT-AO routing

The sender sends an SMS message, which arrives in the RTR (flow 1). Immediately, the RTR performs one delivery attempt to the recipient (flow 2), which in this example is a mobile phone. When this delivery attempt is successful, optionally a billing record is generated (flow 3). If the attempt fails, the HUB forwards the message as an AO message to the SMSC over TCP/IP (flow 4). Optionally, a billing record can be generated when the message is delivered to the SMSC (flow 5).

3.8 AT Routing

3.8.1 Introduction

Application-terminated (AT) routing is the generic name for the processing of incoming (ATI) and outgoing (ATO) application-terminated messages.

The RTR and HUB work in conjunction to route AT messages to their destinations. In the case of AT routing with storage, the AMS acts as a message store. The AMS can also provide a transaction database (called the Intermediate Cache, or Icache) that stores message state and parameters while the message itself is stored in an external SMSC. For more information about the Icache, refer to the AMS Operator Manual.

Note: MIB items that are related to ATO rules use the acronym AT. MIB items that are related to ATI rules use the acronym ATI.

Up to 500 ATIR rules and up to 500 ATOR rules can be defined.

Incoming AT Messages

When an incoming AT message arrives at the RTR, it is evaluated by the following rules, in the following order:

1. Incoming AT external condition (ATIX) rules
2. Incoming AT routing (ATIR) rules
3. Incoming AT counting (ATIC) rules

Outgoing AT Messages

When outgoing AT message is ready to leave the RTR, it is evaluated by the following rules, in the following order:

1. Outgoing AT external condition (ATOX) rules
2. Outgoing AT routing (ATOR) rules
3. Outgoing AT counting (ATOC) rules

3.8.2 AT Routing Paths

The available AT routing paths are:

- AT-AT (route to application)
- AT-AT-Store (route to application, fallback to storage)
- AT-Store-AT (store for delivery to application)
- AT-AO (route to SMSC as AO)

- AT-AO-Store (route to SMSC as AO, fallback to storage)
- AT-Store-AO (store for delivery to SMSC as AO)
- AT-Discard:
  - Discard with ACK
  - Discard with temporary error
  - Discard with permanent message error
  - Discard with permanent recipient error

Note: During the AT delivery via store, the configured value of the "outside transmit window size" is used as the `maxtransactions` in the store request. For the AT delivery "outside transmit window size" of the configured application is used for load balancing the traffic towards the HUB.

3.8.3 AT-AT Routing

AT-AT routing allows for monitoring and controlling traffic toward applications. In AT-AT routing, the RTR receives incoming AT messages from a service centre and routes them to an application. AT-AT messages are first evaluated by the incoming AT routing (ATIR) rules, then by the outgoing AT (ATOR) routing rules.

Figure 12: AT-AT routing

The **Destination Application** parameter in the ATIR rule determines the application to which the RTR routes the message:

- The application that sent the message
- The recipient application that is specified in the message
- An application that the user specifies in the ATIR rule
- An application that is determined by the Intermediate Cache (Icache)

Note: During the AT delivery via store, the configured value of the "outside transmit window size" is used as the `maxtransactions` in the store request. For the AT delivery "outside transmit window size" of the configured application is used for load balancing the traffic towards the HUB.

Note: For more information about the Icache, refer to the AMS Operator Manual.

3.8.4 AT-AT-Store Routing

In AT-AT-Store routing, if the application rejects a message with a temporary error, the message can fall back to storage in the AMS. The AMS delivery scheme determines when and how often message delivery is retried.

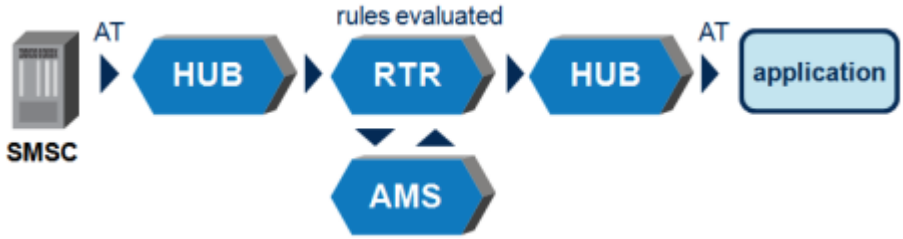


Figure 13: AT-AT-Store routing

The **AMS Queue** parameter in the ATIR rule determines the queue in which the message is stored (and, therefore, the delivery scheme that applies to the message).

Note: During the AT delivery via store, the configured value of the "outside transmit window size" is used as the `maxtransactions` in the store request. For the AT delivery "outside transmit window size" of the configured application is used for load balancing the traffic towards the HUB.

3.8.5 AT-Store-AT Routing

In AT-Store-AT routing, the RTR does not perform a first delivery attempt of the message. Instead, it immediately sends the message to the AMS for storage. The AMS delivery scheme determines when and how often message delivery is tried.

Note: During the AT delivery via store, the configured value of the "outside transmit window size" is used as the `maxtransactions` in the store request. For the AT delivery "outside transmit window size" of the configured application is used for load balancing the traffic towards the HUB.

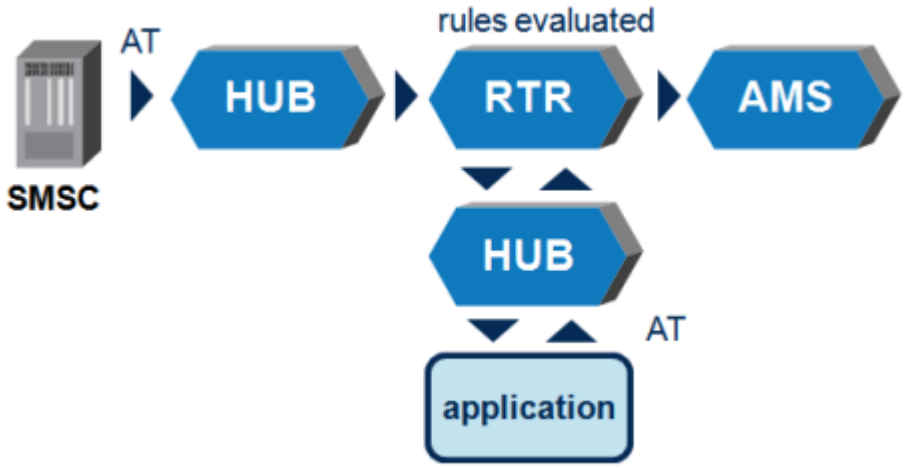


Figure 14: AT-Store-AT

3.8.6 AT-AO Routing

AT-AO routing allows a carrier to receive inter-carrier peer-to-peer AT messages, convert them to AO messages, and route them to the appropriate home network SMSC. AT-AO can also enable intra-carrier SMSCs to route AT messages that arrive at the wrong SMSC to the correct SMSC as AO messages.

AT-AO routing requires the AT-AO license.

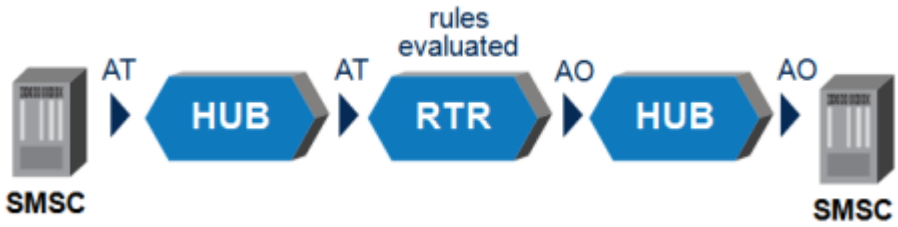


Figure 15: AT-AO

AT-AO routing flows as follows:

1. An inbound AT message arrives at the HUB
2. The HUB sends the AT message to the RTR for rule evaluation
3. The RTR evaluates the ATIR rules for the message, and it matches an AT-AO rule
   a. If the protocol of the incoming AT message does not match the protocol that is accepted by the destination application, the RTR blocks the message with a permanent error and processing of the message stops
   b. If the protocol of the incoming AT message matches the protocol that is accepted by the destination application, the RTR converts the AT message to AO and sends the AO message to the HUB
4. The HUB sends the AO message to the SMSC
5. The SMSC acknowledges (ACKs) or negatively acknowledges (NACKs) the AO message to the HUB
6. The HUB relays the SMSC's response to the RTR
   a. If the AO message was ACKed, the RTR sends an ACK for the AT message to the HUB
   b. If the AO message was NACKed, the RTR sends a NACK for the AT message to the HUB
7. The HUB relays the RTR's response to the AT originator

Note: This routing path is not supported for CIMD messages.

3.8.7 AT-AO-Store Routing

In AT-AO-Store routing, if the SMSC NACKs the AO message with a temporary error, the message can fall back to storage in the AMS. The AMS delivery scheme determines when and how often message delivery is retried.

AT-AO-Store routing requires the following licenses:

- AT-AO
- AT-Store
- Store-AO

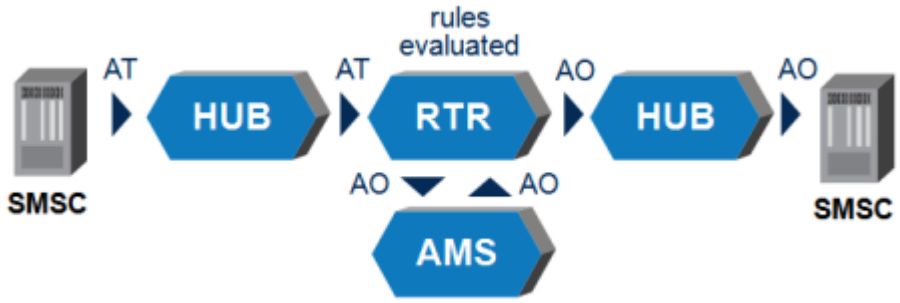


Figure 16: AT-AO-Store

AT-AO-Store routing flows as follows:

1. An inbound AT message arrives at the HUB
2. The HUB sends the AT message to the RTR for rule evaluation
3. The RTR evaluates the ATIR rules for the message, and it matches an AT-AO-Store rule
   a. If the protocol of the incoming AT message does not match the protocol that is accepted by the destination application, the RTR blocks the message with a permanent error and processing of the message stops
   b. If the protocol of the incoming AT message matches the protocol that is accepted by the destination application, the RTR converts the AT message to AO and sends the AO message to the HUB
4. The HUB sends the AO message to the SMSC
5. The SMSC ACKs or NACKs the AO message to the HUB
6. The HUB relays the SMSC's response to the RTR
   a. If the AO message was ACKed, the RTR sends an ACK for the AT message to the HUB (this is the AT-AO routing path) and steps 7, 9, and 10 are omitted from the flow
   b. If the AO message was NACKed with a temporary error, the RTR attempts to store the AO message in the AMS
7. The AMS ACKs or NACKs the storage request
   a. If the storage request was ACKed, the RTR sends an ACK for the AT message to the HUB
   b. If the storage request was NACKed, the RTR sends a NACK for the AT message to the HUB
8. The HUB relays the RTR's response to the AT originator
9. When the delivery scheme indicates that the message should be delivered, the AMS notifies the RTR
10. The RTR sends the message to the SMSC
    a. If the SMSC ACKs the message, the RTR notifies the AMS, which then deletes its internal copy of the message
    b. If the SMSC NACKs the message with a temporary error, the RTR notifies the AMS, which continues to store the message until the next scheduled delivery attempt

The **AMS Queue** parameter in the ATIR rule determines the queue in which the message is stored (and, therefore, the delivery scheme that applies to the message).

Note: This routing path is not supported for CIMD messages.

3.8.8 AT-Store-AO Routing

In AT-Store-AO routing, the RTR does not perform a first submission attempt of the message. Instead, it immediately sends the message to the AMS for storage. The AMS delivery scheme determines when and how often message delivery is tried.

AT-Store-AO routing requires the following licenses:

- AT-Store
- Store-AO

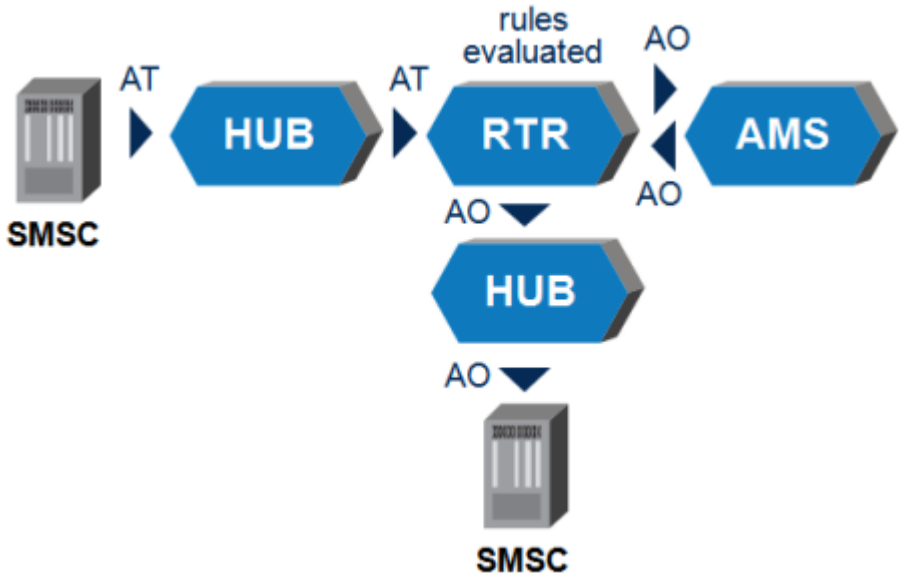


Figure 17: AT-Store-AO

AT-Store-AO routing flows as follows:

1. An inbound AT message arrives at the HUB
2. The HUB sends the AT message to the RTR for rule evaluation
3. The RTR evaluates the ATIR rules for the message, and it matches an AT-Store-AO rule
   a. If the protocol of the incoming AT message does not match the protocol that is accepted by the destination application, the RTR blocks the message with a permanent error and processing of the message stops
   b. If the protocol of the incoming AT message matches the protocol that is accepted by the destination application, the RTR converts the AT message to AO and attempts to store it in the AMS
4. The AMS ACKs or NACKs the storage request
   a. If the storage request was ACKed, the RTR sends an ACK for the AT message to the HUB
   b. If the storage request was NACKed, the RTR sends a NACK for the AT message to the HUB
5. The HUB relays the RTR's response to the AT originator
6. When the delivery scheme indicates that the message should be delivered, the AMS notifies the RTR

7. The RTR sends the message to the SMSC
   a. If the SMSC ACKs the message, the RTR notifies the AMS, which then deletes its internal copy of the message
   b. If the SMSC NACKs the message with a temporary error, the RTR notifies the AMS, which continues to store the message until the next scheduled delivery attempt

The **AMS Queue** parameter in the ATIR rule determines the queue in which the message is stored (and, therefore, the delivery scheme that applies to the message).

Note: This routing path is not supported for CIMD messages.

3.9 Routing Rule Conditions

Each routing rule can have conditions based on the message fields that are available, using the following rule condition structure:

```
([NOT] Condition1) AND ([NOT] Condition2) AND ([NOT] Condition3) AND...
```

The `NOT` operator is optional. Each rule condition can be preceded by the `NOT` operator, which complements the Boolean result of a rule condition evaluation.

3.9.1 AO Rule Conditions

The following table details the conditions that are available for AO rules.

Condition	Values	Description
Time Schedule	• Always • Schedule	Condition on the evaluation time of the message: • Always: The condition is always true. • Schedule: The condition is true if the current time falls within the schedule as specified by the selected routing schedule, defined in **Routing ➤ Schedules**.
Application	• None • Application	Index of the application originating the message.
Application Group	• None • Bit string	Index of the application group associated with the application originating the message.
Application Category	• None • Bit string	Index of the application category associated with the application originating the message.
Service Class	• None • Bitstring	Index of the service class associated with the application originating the message.

Condition	Values	Description

Protocol	• None
	• Bitstring
	  • UCP
	  • SMPP
	  • CIMD
		Protocol used by the originating application to submit the message.

Operation	• None
	• Bit string
	  • 0: UCP call input operation (01)
	  • 1: UCP multiple address call input operation (02)
	  • 2: UCP call input with supplementary services operation (03)
	  • 3: UCP MS message transfer operation (30)
	  • 4: UCP submit short message operation (51)
	  • 9: UCP operation other than above
	  • 10: SMPP SubmitSm
	  • 12: SMPP SubmitMulti
	  • 13: SMPP DataSm
	  • 19: SMPP operation other than above
	  • 20: CIMD SubmitMessage
	  • 29: CIMD operation other than above
	  • 30: Other operations (from another protocol than UCP, SMPP, or CIMD)
		Operation used by the originating application to submit the SM. The condition pertains to the following protocol specific fields:
		• UCP: Command ID
		• SMPP: Command ID
		• CIMD: Operation Code

Condition	Values	Description
Originator	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List	Originator that is specified in the SM. The condition pertains to the following protocol specific fields: • UCP: OAdC / OTOA • SMPP: source_addr • CIMD: [Alphanumeric] Originating address

Originator TON	• None
	• Bit string
	  • 0: Unknown
	  • 1: International number
	  • 2: National number
	  • 3: Network specific number
	  • 4: Subscriber number
	  • 5: Alphanumeric
	  • 6: Abbreviated number
	  • 7: Reserved
		Type of number (TON) specified for the originator of the SM. The condition pertains to the following protocol specific fields:
		• UCP: OAdC / OTOA
		• SMPP: source_addr_ton
		• CIMD: [Alphanumeric] Originating address

Originator NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN Telephony
	  • 2: Data
	  • 3: Telex
	  • 4: Reserved
	  • 5: Reserved
	  • 6: Reserved
	  • 7: Reserved
	  • 8: National numbering plan
	  • 9: Private numbering plan
	  • 10: ERMES numbering plan
		Numbering plan identification (NPI) specified for the originator of the SM. The condition pertains to the following protocol specific fields:
		• UCP: OTOA
		• SMPP: source_addr_npi
		• CIMD: [Alphanumeric] Originating address

Condition	Values	Description
	• 11: Reserved • 12: Reserved • 13: Reserved • 14: Reserved • 15: Reserved	
Recipient	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List • Single IMSI • IMSI range • IMSI prefix	Recipient that is specified in the SM. The condition pertains to the following protocol specific fields : • UCP: ADC • SMPP: destination_addr • CIMD: Destination address **Note:** When an IMSI-related condition is used but the recipient IMSI is not known at the moment of rule evaluation, the condition will evaluate to FALSE if not negated or TRUE if negated. This condition requires the recipient number to be an MSISDN and the HLR query to be performed before the rule evaluation (to obtain the recipient IMSI). The **Early SRI-SM for AO/SM** attribute in the MGR (**Routing > Properties**) controls when the HLR query is performed.

Recipient TON	• None
	• Bit string
	  • 0: Unknown
	  • 1: International number
	  • 2: National number
	  • 3: Network specific number
	  • 4: Subscriber number
	  • 5: Alphanumeric
	  • 6: Abbreviated number
	  • 7: Reserved
		Type of number (TON) specified for the recipient of the SM. The condition pertains to the following protocol specific fields:
		• UCP: ADC
		• SMPP: destination_addr_ton
		• CIMD: Destination address

Recipient NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN telephony
	  • 2: Data
	  • 3: Telex
		Numbering plan identification (NPI) specified for the recipient of the SM. The condition pertains to the following protocol specific fields:
		• UCP: (always isdnTelephony)
		• SMPP: destination_addr_npi
		• CIMD: (always isdnTelephony)

Condition	Values	Description
	• 4: Reserved • 5: Reserved • 6: Reserved • 7: Reserved • 8: National numbering plan • 9: Private numbering plan • 10: ERMES numbering plan • 11: Reserved • 12: Reserved • 13: Reserved • 14: Reserved • 15: Reserved	
Terminating MSC/SGSN	• Single MSISDN • MSISDN range • MSISDN prefix • List • Country • Network	Destination MSC and/or SGSN. This condition requires the recipient number to be an MSISDN, and requires the HLR query to be performed before AO rule evaluation (to obtain the MSC and/or SGSN). The **Early SRI-SM for AO/SM** attribute in the MGR (**Routing > Properties**) controls when the HLR query is performed. If both the MSC and SGSN are present, the `preferredmtdestination` attribute determines which will be used for rules evaluation. If the HLR query fails, the condition will evaluate to "false", whether it is a negative or positive condition.
User Data	• None • Full text • Text tag • Subtext (contains) • Text length	The text that is specified in the user data of the SM. The condition pertains to the following protocol specific fields: • UCP: Amsg • SMPP: message_payload • CIMD: User data [binary] Content of the message: • Full text: Compares the entire message content with a defined string. Only evaluates positively if there is an exact match with the user data (message content). • Text tag: Compares the first part of the message content with a defined tag-string. Only evaluates positively if the message starts exactly with the defined tag-string. Useful to define a rule on keywords (e.g. `*LONG#` for CMG SMSCs).

Condition	Values	Description
		• Subtext (contains): Compares if the message content contains a defined tag-string. To optimise the search, a start position (default 1) and an end position (default 160) must be specified in which the string is to be found. **Note:** All message content scanning is case-insensitive.

User Data Header Indication	• None
	• Bit value
	  • 0
	  • 1
		User data header indication that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: XSER (GSM UDH information)
		• SMPP: esm_class (UDHI) or UDH information received in concatenation related TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`)
		• CIMD: User data header

User Data Header	• None • Byte value	Value specified in one of the IEIs of the user data header (UDH) of the SM. The condition pertains to the following protocol specific fields: • UCP: XSER (GSM UDH information) • SMPP: short_message (prefix) or UDH information received in concatenation related TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`) • CIMD: User data header Only evaluates positively if there is an exact match with the information element identifier (00-FF, hexadecimal) of the UDH. Refer to technical specification 3GPP 23.040 for more information. Most common information element identifier (IEI) values: • 00: Concatenated short message • 01: Special SMS message indication • 04: Application port addressing scheme, 8-bit address • 05: Application port addressing scheme, 16-bit address • 06: SMSC control parameters • 07: UDH Source Indicator
Reply Path	• None • Bit value • 0: Off	Reply path indication that is specified in the SM. The condition pertains to the following protocol specific fields: • UCP: RPI

Condition	Values	Description
	• 1: On	• SMPP: esm_class (RPI) • CIMD: Reply Path
Notification Address	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List • Single TCP/IP address • Single X121 address	Notification address that is specified in the SM. The condition pertains to the following protocol specific fields: • UCP: NAdC

Notification Request	• None
	• Bit value
	  • 0
	  • 1
		Notification request indication that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: Nrq
		• SMPP: registered_delivery
		• CIMD: Status report request

Notification Type	• None
	• Bit string
	  • Delivered
	  • Not delivered
	  • Buffered
		Notification type that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: Nrq
		• SMPP: registered_delivery
		• CIMD: Status report request
		The following bit string values apply:
		• Delivered: Delivery Notification (DN)
		• Not delivered: Non-delivery notification (ND)
		• Buffered: Buffered message notification (BN)
		0 default value, 1 = DN, 2 = ND, 3 = DN+ND, 4 = BN, 5 = BN+DN, 6 = BN+ND, 7 = all.

Single Shot Indicator	• None
	• Bit value
	  • 0
	  • 1
		Single-shot indicator that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: XSER (single Shot Indication)

Condition	Values	Description

More Messages To Send	• None
	• Bit value
	  • 0
	  • 1
		More-messages-to-send indicator that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: MMS
		• SMPP: more_messages_to_send
		• CIMD: More messages to send

Priority	• None
	• Bit string
	  • 0: UCP no priority requested
	  • 1: UCP priority requested
	  • 10: SMPP no priority requested
	  • 11: SMPP priority requested
	  • 20: CIMD priority level 1 (most urgent)
	  • 21: CIMD priority level 2
	  • 22: CIMD priority level 3
	  • 23: CIMD priority level 4
	  • 24: CIMD priority level 5
	  • 25: CIMD priority level 6
	  • 26: CIMD priority level 7
	  • 27: CIMD priority level 8
	  • 28: CIMD priority level 9
		Priority level that is specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: PR
		• SMPP: priority_flag
		• CIMD: Priority

Protocol Identifier (PID)	• None • Byte value (between 00 and FF hexadecimal)	Protocol ID (PID) specified in the SM. The condition pertains to the following protocol specific fields: • UCP: RPID • SMPP: protocol_id • CIMD: Protocol identifier

Condition	Values	Description
Data Coding Scheme (DCS)	• None • Byte value (between 00 and FF hexadecimal)	Data coding scheme (DCS)specified in the SM. The condition pertains to the following protocol specific fields: • UCP: XSER (DCS) • SMPP: data_coding • CIMD: Data coding schema **Note:** If the value of the HUB parameter `hubPropDcsCharCodingConversion` and `rtrdcscharcodingconversion` are set to '`japan`' and the `data_coding` value in the incoming SMPP AO message is 0x02, then this rule shall be applied on the converted `data_coding` 0x04.

Concat. Msg. Segments	• None
	• Bit String
	  • First Segment
	  • Last Segment
	  • Not First Nor Last
		Segment sequence number of a concatenated SM. The condition pertains to the following protocol specific fields:
		• UCP: XSER (GSM UDH information)
		• SMPP: short_message (prefix) or UDH information received in concatenation related TLVs (`sar_segment_seqnum`)
		• CIMD: User data header

UCP Authentication Code	• None • Digit string	Authentication code specified in the SM. The condition pertains to the following protocol specific fields: • UCP: AC

Notification PID	• None
	• Bit string
	  • 0: Mobile station
	  • 1: Fax group 3
	  • 2: Menu over PSTN
	  • 3: PC application over PSTN
	  • 4: PC application over ISDN
	  • 5: PC application over TCP/IP
	  • 6: PC application over X25
	  • 7: Unknown
		The Notification PID specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: NPID / Nrq
		Note: To have the NPID matched, the UCP fields Notification Request (NRq) must be set, Notification Type (NT) should be larger then 0, and Notification Address (NAdC) should be filled in as well in the message.

Condition	Values	Description
Last Resort Address	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List • Single TCP/IP address • Single X121 address	Last Resort Address (LRAd) specified in the SM. The condition pertains to the following protocol specific fields: • UCP: LRAD

Last Resort Request	• None
	• Bit value
	  • 0
	  • 1
		Last Resort Address Request indicator specified in the SM. The condition pertains to the following protocol specific fields:
		• UCP: LRAd
		The following bit string values apply:
		• 0 = LRAd not used
		• 1 = LRAd used

Last Resort PID	• None
	• Bit string
	  • 0: Mobile station
	  • 1: Fax group 3
	  • 2: Menu over PSTN
	  • 3: PC application over PSTN
	  • 4: PC application over ISDN
	  • 5: PC application over TCP/IP
	  • 6: PC application over X25
	  • 7: Unknown
		Last Resort Address PID specified SM. The condition pertains to the following protocol specific fields:
		• UCP: LPID

CIMD Tariff Class	• None • Bit string	Index of the CIMD tariff class specified in the SM. The condition pertains to the following protocol specific fields: • CIMD: Tariff class

Condition	Values	Description
CIMD Service Description	• None • Bit string	Index of the CIMD service description specified in the SM. The condition pertains to the following protocol specific fields: • CIMD: Service description
Ext Att	• None • External attributes	A set of 32 attributes, the value of which can be controlled by external condition (EC) applications.
Validity Period	• None • Relative Time	Validity period specified in the SM. The condition pertains to the following protocol specific fields: • UCP: VP • SMPP: validity_period • CIMD: Validity period absolute/relative
Deferred Delivery Time	• None • Relative Time	Deferred delivery time specified in the SM. The condition pertains to the following protocol specific fields: • UCP: DDT • SMPP: scheduled_delivery_time • CIMD: First delivery time absolute/relative
Recipient RN Group	• None • SNMP index	Routing number (RN) group to which the RN in the recipient address belongs
Originator SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual originator subscriber services (defined in **SPF Services ➤ SPF Services**). The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the condition would only be satisfied if both of these tests are true.
Recipient SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual recipient subscriber services. The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the condition would only be satisfied if both of these tests are true.

3.9.2 ATI Rule Conditions

The following table details the conditions that are available for ATIR rules.

Condition	Values	Description
Time Schedule	• Always • Schedule	Condition on the evaluation time of the message: • Always: The condition is always true. • Schedule: The condition is true if the current time falls within the schedule as specified by the selected routing schedule, defined in **Routing ➤ Schedules**.
Application	• None • Application	Index of the application originating the message.
Application Group	• None • Bit string	Index of the application group associated with the application originating the message.
Application Category	• None • Bit string	Index of the application category associated with the application (with inside session) sending the inbound AT message.
Service Class	• None • Bit string	Index of the service class associated with the application originating the message.

Message Type	• None
	• Bit string
	  • 0: Normal AT message
	  • 1: Delivery notification
		Indicates which message type applies (each bit corresponds to a message type). If the bit is 1, the message type applies. If the bit is 0, the message type does not apply.
		Note: Bit 0 is the least significant (last) bit, while bit 2 is the most significant (first) bit.

Protocol	• None
	• Bit string
	  • UCP
	  • SMPP
	  • CIMD
		The protocol used by the originating application to submit the message.

Operation	• None
	• Bit string
	  • 0: CP call input operation (01)
	  • 1: UCP multiple address call input operation (02)
		The operation used by the originating application to submit the message.

Condition	Values	Description
	• 2: UCP call input with supplementary services operation (03) • 3: UCP MS message transfer operation (30) • 4: UCP submit short message operation (51) • 9: UCP operation other than above • 10: SMPP SubmitSm • 12: SMPP SubmitMulti • 13: SMPP DataSm • 19: SMPP operation other than above • 20: CIMD SubmitMessage • 29: CIMD operation other than above • 30: Operation from another protocol than UCP, SMPP, or CIMD	
Originator	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List	The originator specified in the message.
Originator TON	• None • Bit string • 0: Unknown	The type of number (TON) specified in the originator address of the message.

Condition	Values	Description
	• 1: International number • 2: National number • 3: Network specific number • 4: Subscriber number • 5: Alphanumeric • 6: Abbreviated number • 7: Reserved	

Originator NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN telephony
	  • 2: Data
	  • 3: Telex
	  • 4: Reserved
	  • 5: Reserved
	  • 6: Reserved
	  • 7: Reserved
	  • 8: National numbering plan
	  • 9: Private numbering plan
	  • 10: ERMES numbering plan
	  • 11: Reserved
	  • 12: Reserved
	  • 13: Reserved
	  • 14: Reserved
	  • 15: Reserved
		The numbering plan identification (NPI) specified for the recipient of the message.

Recipient	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix	The recipient specified in the message.

Condition	Values	Description
	• Application • List	

Recipient TON	• None
	• Bit string
	  • 0: Unknown
	  • 1: International number
	  • 2: National number
	  • 3: Network specific number
	  • 4: Subscriber number
	  • 5: Alphanumeric
	  • 6: Abbreviated number
	  • 7: Reserved
		The type of number (TON) specified in the recipient address of the message.

Recipient NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN telephony
	  • 2: Data
	  • 3: Telex
	  • 4: Reserved
	  • 5: Reserved
	  • 6: Reserved
	  • 7: Reserved
	  • 8: National numbering plan
	  • 9: Private numbering plan
	  • 10: ERMES numbering plan
	  • 11: Reserved
	  • 12: Reserved
	  • 13: Reserved
	  • 14: Reserved
	  • 15: Reserved
		The number plan identifier (NPI) specified in the recipient address of the message.

Priority	• None	The priority level specified in the message.

Condition	Values	Description

	• Bit string
	  • 0: UCP no priority requested
	  • 1: UCP priority requested
	  • 10: SMPP no priority requested
	  • 11: SMPP priority requested
	  • 20: CIMD priority level 1 (most urgent)
	  • 21: CIMD priority level 2
	  • 22: CIMD priority level 3
	  • 23: CIMD priority level 4
	  • 24: CIMD priority level 5
	  • 25: CIMD priority level 6
	  • 26: CIMD priority level 7
	  • 27: CIMD priority level 8
	  • 28: CIMD priority level 9	

PID	• None • Byte value (value between 00 and FF, hexadecimal)	The protocol ID specified in the message.
Service Centre	• None • Bit string	Index of the service centre (each bit in the string corresponds to a service centre). If the bit is 1, the service centre (SC) applies. If the bit is 0, the service centre does not apply. Up to 250 service centres can be defined. Therefore, the bit string consists of 250 bits with the least significant (last) bit indicating whether the service centre with SNMP index 1 applies and the most significant (first) bit indicating whether the service centre with SNMP index 250 applies.

Condition	Values	Description
SC Timestamp	• None • Relative time	The SC timestamp specified in the message.

Segment Type	• None
	• Bit string
	  • 0: First segment of a segmented message
	  • 1: Last segment of a segmented message
	  • 2: Neither the first nor the last segment of a segmented message
		Indicates which concatenated message segments apply (each bit corresponds to a segment of a concatenated message). If the bit is 1, the segment applies. If the bit is 0, the segment does not apply.
		Note:
		• Bit 0 is the least significant (last) bit, while bit 2 is the most significant (first) bit.
		• Concatenation information received in SMPP SAR TLV (`sar_segment_seqnum`) is also considered while matching this condition.

Data Coding Scheme (DCS)	• None • Byte value (value between 00 and FF, hexadecimal)	The data coding scheme (DCS) specified in the message.
User Data	• None • Full text • Text tag • Subtext (contains) • Text length	Content of the message: • Full text: Compares the entire message content with a defined string. Only evaluates positively if there is an exact match with the user data (message content). • Text tag: Compares the first part of the message content with a defined tag-string. Only evaluates positively if the message starts exactly with the defined tag-string. Useful to define a rule on keywords (e.g. `*LONG#` for CMG SMSCs). • Subtext (contains): Compares if the message content contains a defined tag-string. To optimise the search, a start position (default 1) and an end position (default 160) must be specified in which the string is to be found. **Note:** All message content scanning is case-insensitive.

User Data Header Indication	• None
	• Bit value
	  • 0
	  • 1
		The user data header (UDH) indication specified in the message.
		Note: UDH information received in concatenation related SMPP TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`) is also considered while matching this condition.

Condition	Values	Description
User Data Header	• None • Byte value	Value specified in one of the IEIs of the UDH of the message. Only evaluates positively if there is an exact match with the information element identifier (00-FF, hexadecimal) of the UDH. Refer to technical specification 3GPP 23.040 for more information. Most common information element identifier (IEI) values: • 00: Concatenated short message • 01: Special SMS message indication • 04: Application port addressing scheme, 8-bit address • 05: Application port addressing scheme, 16-bit address • 06: SMSC control parameters • 07: UDH Source Indicator **Note:** UDH information received in concatenation related SMPP TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`) is also considered while matching this condition.

More Messages To Send	• None
	• Bit value
	  • 0
	  • 1
		The more-messages-to-send indicator specified in the message.

Delivery Status	• None
	• Bit string
	  • 0 UCP: in progress
	  • 1 UCP: delivery failed
	  • 2 UCP: delivery successful
	  • 10 SMPP: no status available
	  • 11 SMPP: in progress
	  • 12 SMPP: validity period expired
	  • 13 SMPP: delivery failed
	  • 14 SMPP: delivery successful
	  • 15 SMPP: deleted
		The delivery status specified in the notification.

Condition	Values	Description
	• 16 SMPP: deleted by cancel • 17 SMPP: scheduled • 18 SMPP: accepted • 19 SMPP: rejected • 20 CIMD: no status available • 21 CIMD: in progress • 22 CIMD: validity period expired • 23 CIMD: delivery failed • 24 CIMD: delivery successful • 25 CIMD: no response • 26 CIMD: last no response • 27 CIMD: cancelled • 28 CIMD: deleted • 29 CIMD: deleted by cancel	
Delivery Timestamp	• None • Relative time	The delivery timestamp specified in the notification.
Ext Att	• None • External attributes	A set of 32 attributes, the value of which can be controlled by external applications.
Originator SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual originator subscriber services (defined in **SPF Services ➤ SPF Services**) . The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the condition would only be satisfied if both of these tests are true.
Recipient SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual recipient subscriber services. The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the

Condition	Values	Description
		condition would only be satisfied if both of these tests are true.

3.9.3 ATO Rule Conditions

The following table details the conditions that are available for ATOR rules.

Condition	Values	Description
Time Schedule	• Always • Schedule	Condition on the evaluation time of the message: • Always: The condition is always true. • Schedule: The condition is true if the current time falls within the schedule as specified by the selected routing schedule, defined in **Routing ➤ Schedules**.
Application	• None • Application	Index of the application originating the message.
Application Group	• None • Bit string	Index of the application group associated with the application originating the message.
Application Category	• None • Bit string	Index of the application category associated with the application (with outside session) receiving the outbound AT message.
Service Class	• None • Bitstring	Index of the service class associated with the application originating the message.

Protocol	• None
	• Bit string
	  • UCP
	  • SMPP
	  • CIMD
		The protocol used by the originating application to submit the message.

Operation	• None
	• Bit string
	  • 0: CP call input operation (01)
	  • 1: UCP multiple address call input operation (02)
		The operation used by the originating application to submit the message.
		Note: For AT/AT-AT routing paths the ATOC rules cannot be used with the Operation condition (they can only be used on AGW systems).

Condition	Values	Description
	• 2: UCP call input with supplementary services operation (03) • 3: UCP MS message transfer operation (30) • 4: UCP submit short message operation (51) • 9: UCP operation other than above • 10: SMPP SubmitSm • 12: SMPP SubmitMulti • 13: SMPP DataSm • 19: SMPP operation other than above • 20: CIMD SubmitMessage • 29: CIMD operation other than above • 30: Operation from another protocol than UCP, SMPP, or CIMD	
Protocol Identifier (PID)	• None • Byte value (value between 00 and FF, hexadecimal)	The protocol ID specified in the message.

More Messages To Send	• None
	• Bit value
	  • 0
	  • 1
		The more-messages-to-send indicator specified in the message.

Originator	• Single MSISDN • MSISDN range • MSISDN prefix • Country	The originator specified in the message.

Condition	Values	Description
	• Network • Single short number • Short number range • Short number prefix • Application • List	

Originator TON	• None
	• Bit string
	  • 0: Unknown
	  • 1: International number
	  • 2: National number
	  • 3: Network specific number
	  • 4: Subscriber number
	  • 5: Alphanumeric
	  • 6: Abbreviated number
	  • 7: Reserved
		The type of number (TON) specified in the originator address of the message.

Originator NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN Telephony
	  • 2: Data
	  • 3: Telex
	  • 4: Reserved
	  • 5: Reserved
	  • 6: Reserved
	  • 7: Reserved
	  • 8: National numbering plan
	  • 9: Private numbering plan
	  • 10: ERMES numbering plan
	  • 11: Reserved
	  • 12: Reserved
	  • 13: Reserved
	  • 14: Reserved
		The numbering plan identification (NPI) specified for the recipient of the message.

Condition	Values	Description
	• 15: Reserved	
Originator IMSI	• None • Single IMSI • IMSI range • County • Network • List	The IMSI specified for the originator of the message.
Recipient	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • Single short number • Short number range • Short number prefix • Application • List	The recipient specified in the message.

Recipient TON	• None
	• Bit string
	  • 0: Unknown
	  • 1: International number
	  • 2: National number
	  • 3: Network specific number
	  • 4: Subscriber number
	  • 5: Alphanumeric
	  • 6: Abbreviated number
	  • 7: Reserved
		The type of number (TON) specified in the recipient address of the message.

Recipient NPI	• None
	• Bit string
	  • 0: Unknown
	  • 1: ISDN Telephony
	  • 2: Data
	  • 3: Telex
		The number plan identifier (NPI) specified in the recipient address of the message.

Condition	Values	Description
	• 4: Reserved • 5: Reserved • 6: Reserved • 7: Reserved • 8: National numbering plan • 9: Private numbering plan • 10: ERMES numbering plan • 11: Reserved • 12: Reserved • 13: Reserved • 14: Reserved • 15: Reserved	
Orig. MSC/SGSN	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • List	The MSC or SGSN serving the originator specified in the message.
SMSC Address	• Single MSISDN • MSISDN range • MSISDN prefix • Country • Network • List	The SMSC specified in the message.
User Data	• None • Full text • Text tag • Subtext (contains) • Text length	Content of the message: • Full text: Compares the entire message content with a defined string. Only evaluates positively if there is an exact match with the user data (message content). • Text tag: Compares the first part of the message content with a defined tag-string. Only evaluates positively if the message starts exactly with the defined tag-string. Useful to define a rule on keywords (e.g. `*LONG#` for CMG SMSCs). • Subtext (contains): Compares if the message content contains a defined tag-string. To optimise the search, a start position (default 1) and an end

Condition	Values	Description
		position (default 160) must be specified in which the string is to be found. **Note:** All message content scanning is case-insensitive.

User Data Header Indication	• None
	• Bit value
	  • 0
	  • 1
		The user data header (UDH) indication specified in the message.
		
		Note: UDH information received in concatenation related SMPP TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`) is also considered while matching this condition.

User Data Header	• None • Byte value	Value specified in one of the IEIs of the UDH of the message. Only evaluates positively if there is an exact match with the information element identifier (00-FF, hexadecimal) of the UDH. Refer to technical specification 3GPP 23.040 for more information. Most common information element identifier (IEI) values: • 00: Concatenated short message • 01: Special SMS message indication • 04: Application port addressing scheme, 8-bit address • 05: Application port addressing scheme, 16-bit address • 06: SMSC control parameters • 07: UDH Source Indicator **Note:** UDH information received in concatenation related SMPP TLVs (`sar_msg_ref_num`, `sar_total_segments`, `sar_segment_seqnum`) is also considered while matching this condition.

Priority	• None
	• Bit string
	  • 0: UCP no priority requested
	  • 1: UCP priority requested
	  • 10: SMPP no priority requested
	  • 11: SMPP priority requested
	  • 20: CIMD priority level 1 (most urgent)
		The priority level specified in the message.

Condition	Values	Description
	• 21: CIMD priority level 2 • 22: CIMD priority level 3 • 23: CIMD priority level 4 • 24: CIMD priority level 5 • 25: CIMD priority level 6 • 26: CIMD priority level 7 • 27: CIMD priority level 8 • 28: CIMD priority level 9	

Message Type	• None
	• Bit string
	  • 0 : Normal AT Message
	  • 1 : Delivery Notification
		Condition pertaining to the message type applicable for this message.

Data Coding Scheme (DCS)	• None • Byte value (value between 00 and FF, hexadecimal)	The data coding scheme (DCS) specified in the message.

Concat. Msg. Segments	• None
	• Bit string
	  • First segment
	  • Last segment
	  • Not first nor last
		The segment sequence number of a concatenated message.
		Note: Concatenation information received in SMPP TLVs (`sar_segment_seqnum`) is also considered while matching this condition.

SC Timestamp	• None • Relative time	The service centre (SC) timestamp specified in the message.

Condition	Values	Description

Delivery Status	• None
	• Bit string
	  • 0 UCP: in progress
	  • 1 UCP: delivery failed
	  • 2 UCP: delivery successful
	  • 10 SMPP: no status available
	  • 11 SMPP: in progress
	  • 12 SMPP: validity period expired
	  • 13 SMPP: delivery failed
	  • 14 SMPP: delivery successful
	  • 15 SMPP: deleted
	  • 16 SMPP: deleted by cancel
	  • 17 SMPP: scheduled
	  • 18 SMPP: accepted
	  • 19 SMPP: rejected
	  • 20 CIMD: no status available
	  • 21 CIMD: in progress
	  • 22 CIMD: validity period expired
	  • 23 CIMD: delivery failed
	  • 24 CIMD: delivery successful
	  • 25 CIMD: no response
	  • 26 CIMD: last no response
	  • 27 CIMD: cancelled
	  • 28 CIMD: deleted
	  • 29 CIMD: deleted by cancel
		The delivery status specified in the notification.
		Note: This condition is only applicable for AT-AT scenarios and not applicable for RTR generated notifications.

Delivery Timestamp	• None • Relative time	The delivery timestamp specified in the notification.

Condition	Values	Description
Ext Att	• None • External attributes	A set of 32 attributes, the value of which can be controlled by external applications.

XS Message	• None
	• Bit string
	  • Not an XS Message
	  • Copy to Application Message
	  • Auto Reply Message
	• Copy to Email
	• Forward to Email
		The eXternal Service (XS) message type applicable for this SM.

Originator SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual originator subscriber services (defined in **SPF Services ➤ SPF Services**). The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the condition would only be satisfied if both of these tests are true.
Recipient SSI	• None • Subscriber Services	Perform a positive or negative test on one or more individual recipient subscriber services. The condition test works like a logical AND operation (similar to the external attributes). For example, if positive tests are selected for originator copy and distribution list, the condition would only be satisfied if both of these tests are true.

3.10 Counting Rules

Counting rules count the number of messages that match sets of user-defined rule conditions. The conditions that are available for each type of counting rule are the same as those that are available for the corresponding type of routing rule. However, no action is associate with counting rules, because their only action is to count.

3.11 External Condition Rules

External condition rules can process traffic and forward selected messages to a configured external condition (EC) application. Commonly used EC applications are the Prepaid Billing Controller (PBC) and Firewall Advanced Filters (FAF).

The conditions that are available for each type of external condition rule are the same as those available for the corresponding type of routing rule. External condition rules contain an additional parameter in which you specify the EC application to which to route messages that match the rule.

3.12 Character Set Conversion

The HUB support for configurable character set conversion allows message data conversion between applications and SMSCs.

It is possible to provision AO and AT conversion tables in applications and SMSCs. These tables define the mapping to and from an external format to the default GSM alphabet. The conversion is applied to text messages, and not to binary messages.

Messages from applications and SMSC are passed via the HUB to the RTR. The HUB insures that all messages from the applications or SMSCs are converted to the default GSM character set.

Note that:

- The customer character set maximum is 8 bits, which means 256 different characters.
- The HUB supports up to five different custom character sets.
- Every custom character set consists of two conversion tables:
  - Custom to default
  - Default to custom
- The content of the conversion tables is configured in the MGR.
- For every application configured in the HUB, it is possible to configure two different custom character sets:
  - One for incoming messages
  - One for outgoing messages
- For every configured SMSC, it is possible to configure two different custom character sets:
  - One for incoming messages
  - One for outgoing messages
- All incoming messages from the application or SMSC are always converted to the standard character set (GSM-character set).
- A message coming from application or SMSC is rejected if the message size exceeds 640 characters after conversion.

The MIB for character conversion is called `textpass-chr-mib.my`.

3.12.1 Character Conversion of Incoming Messages

The HUB converts all incoming messages from an application or SMSC that have a customer character set for outgoing messages configured. All custom characters are converted to default 3GPP 23.038 characters.

A single custom character can be converted to a single GSM character or a combined character `<esc> <char>`. A custom character combination `<esc> <char>` can be converted to a single GSM character or a combined character `<esc> <char>`.

3.12.2 Character Conversion of Outgoing Messages

Character conversion of outgoing messages is performed when the configured application or SMSC has specified a custom character set for incoming messages. The character conversion is done form the 3GPP 23.038 character set to a custom character set.

Character conversion can impact the size of the converted message. The HUB will reject the message from an application or SMSC if the message exceeds the maximum size of 640 characters after conversion to the 3GPP 23.038 character set.

The HUB will send the original message if the content of the message does not needed to be modified and the character set of the application and the SMSC are the same.

3.12.3 Character Conversion and Trace Points

The HUB includes a trace framework that consists of trace filters (which are created in the MGR) and a command-line trace receiver tool (`tp_trace_receiver`) that writes data to a file. The trace filters determine the points at which data is collected, so the collected data can vary depending on the trace points.

When character set conversion is used, the trace points also impact whether the characters in the collected data have not been converted (pre-conversion) or have already been converted (post-conversion).

Trace points impact trace data as follows:

Trace Point	Traffic Type	Conversion State
Outside (on the application side)	Incoming AO (from the application)	Pre-conversion
Outside (on the application side)	Outgoing AT (to the application)	Post-conversion
Inside (on the service centre side)	Outgoing AO (to the service centre)	Post-conversion
Inside (on the service centre side)	Incoming AT (from the service centre)	Pre-conversion
MXP (between the HUB and RTR)	Incoming AO-AT	Post-conversion
MXP (between the HUB and RTR)	Outgoing AO-AT	Pre-conversion

Refer to *System Management* for more information about the trace framework.

3.12.4 Character Set Conversion Example

This section provides an example of character conversion between an 8-bit ASCII SMSC and the RTR.

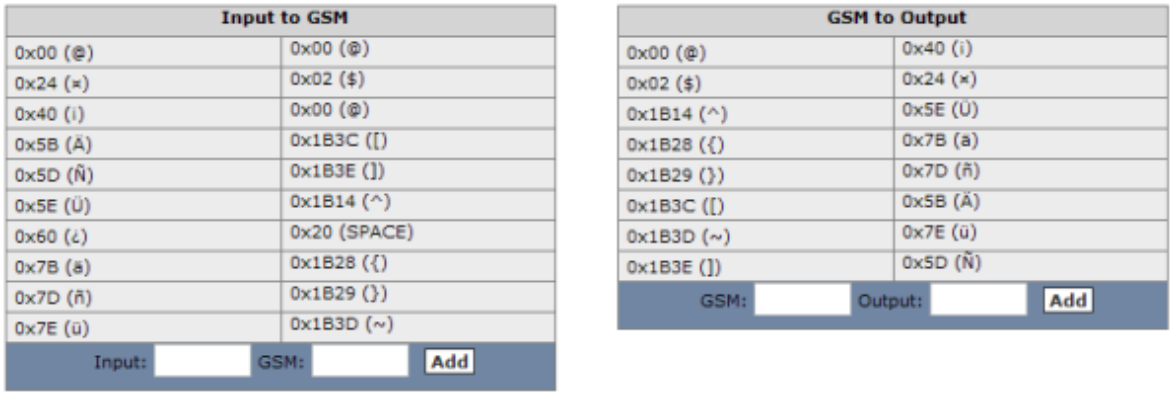


Figure 18: Character set conversion example

3.12.5 TP-DCS/data_coding Character Set Conversion for Japan Network

3.12.5.1 Inbound AO Scenarios

1. The semi-static parameters `rtrdcscharcodingconversion` and `hubdcscharcodingconversion` are used to handle the data_coding/TP-DCS character set conversion for inbound AO scenarios.

   The following table shows the semi-static configuration parameters and their combined behavior:

hubdcscharcodingconversion	rtrdcscharcodingconversion	Behavior
transparent	transparent	Backward compatibility
transparent	japan	**Not Recommended**
japan	transparent	**Not Recommended**
japan	japan	Adapt to Japan Network

   Note: Refer to section *hubdcscharcodingconversion* for the description of `hubdcscharcodingconversion`. Refer to the RTR operator manual for the description of `rtrdcscharcodingconversion`.

2. If `hubdcscharcodingconversion` and `rtrdcscharcodingconversion` semi-static parameters are set to `japan`, then the following coding and character set conversion will take place in case of inbound AO submit message:

<table>
<tr><th colspan="3">submit_sm (AO)</th><th colspan="2">MT-FSM (MT)</th></tr>
<tr><th>[data_coding] value(bit)</th><th>meaning</th><th>short_message / sm_length</th><th>[TP-DCS] value (bit)</th><th>TP-UD/TP-UDL</th></tr>
<tr><td>(00000000)</td><td>Default Alphabet</td><td rowspan="6">• short_message: 8bit ASCII data sm_length: max 140 octets
• not concatenated message max 140 octets
• concatenated message max 134 octets/segment</td><td>(00000000)</td><td rowspan="6">• converted from 8 bit ASCII data to GSM 7-bit data.
• not concatenated message max 160 septets
• concatenated message max 153 septets/segment
• custom character set will be applied if configured.</td></tr>
<tr><td>(1101xxxx)</td><td>Default Alphabet + MWI (GSM MWI control)</td><td>(1101xxxx)</td></tr>
<tr><td>(11110000)</td><td>Default Alphabet Class 0</td><td>(11110000)</td></tr>
<tr><td>(11110001)</td><td>Default Alphabet Class 1</td><td>(11110001)</td></tr>
<tr><td>(11110010)</td><td>Default Alphabet Class 2</td><td>(11110010)</td></tr>
<tr><td>(11110011)</td><td>Default Alphabet Class 3</td><td>(11110011)</td></tr>
<tr><td>(00001000)</td><td>UCS2</td><td>• short_message: UCS2(16-bit data) sm_length: max 140 octets
• not concatenated message max 140 octets
• concatenated message max 134 octets/segment</td><td>(00001000)</td><td>• no conversion (UCS2(16-bit code) to(UCS2) 16-bit code)
• not concatenated message max 140 octets
• concatenated message max 134 octets/segment</td></tr>
<tr><td>(00000010)</td><td>Octet unspecified (8-bit binary)</td><td rowspan="4">• short_message: 8bit ASCII data sm_length: max 140 octets
• not concatenated message max 140 octets
• concatenated message max</td><td>(00000100)</td><td rowspan="4">• short_message: 8bit ASCII data sm_length: max 140 octets
• not concatenated message max 140 octets
• concatenated message max</td></tr>
<tr><td>(00000100)</td><td>Octet unspecified (8-bit binary)</td><td>(00000100)</td></tr>
<tr><td>(00010001)</td><td>Reserved</td><td>(00010001)</td></tr>
<tr><td>other than above patterns (xxxxxxxx)</td><td></td><td>other than above patterns (xxxxxxxx)</td></tr>
</table>

submit_sm (AO)			MT-FSM (MT)	
[data_coding] value(bit)	**meaning**	**short_message / sm_length**	**[TP-DCS] value (bit)**	**TP-UD/TP-UDL**
		134 octets/segment		134 octets/segment

Note: x is an arbitrary digit; it can be 0 or 1.

3. The above conversion is also be applicable for the below scenarios:
   - AO-MT
   - AO-MT-Store
   - AO-Store-MT
   - AO-AO
   - AO-Store-AO
   - AO-AO-Store
   - AO-AT
   - AO-MT-AO
   - AO-Store-AT
   - AO-AT-Store
4. The converted or final user data will get reflect in logging, CDR and Rule as follows:
   - As `userData` in transactional logging.
   - As `Message_Data` in Comverse 3g CDR.
   - As `User Data[cond]` in the rule matching.

   Note: In case of `User Data[cond]` rule matching, if the data is converted in to 8-bit data, then the option **Textlength Range** will be applicable.
5. This is applicable for SMPP version 3.3, 3.4, 5.0.

3.12.5.2 MO-AT Scenarios

1. The semi-static parameters `rtrdcscharcodingconversion` and `hubdcscharcodingconversion` are used to handle the TP-DCS/data_coding character set conversion for MO-AT scenarios.

   The following table shows the semi-static configuration parameters and their combined behavior:

hubdcscharcodingconversion	rtrdcscharcodingconversion	Behavior
transparent	transparent	Backward compatibility
transparent	japan	Backward compatibility
japan	transparent	Backward compatibility
japan	japan	Adapt to Japan Network

Note: Refer to section *hubdcscharcodingconversion* for the description of `hubdcscharcodingconversion`. Refer to the RTR operator manual for the description of `rtrdcscharcodingconversion`.

2. If `hubdcscharcodingconversion` and `rtrdcscharcodingconversion` semi-static parameters are set to `japan`, then the following GSM's TP-DCS to SMPP's data_coding, coding and character set conversion will take place in case message received as MO submit message and sent as AT Deliver_sm message:

MO			**AT**		
[TP-DCS] value(bit)	**meaning**	**TP-UD/TP-UDL**	**[data_coding] value(bit)**	**Short_message / sm_length**	**Is Optional TLV dest_addr_subunit included in Deliver_sm**
(00000000)	GSM 7bit No Message Class	• GSM 7-bit data • not concatenated message max 160 septets • concatenated message max 153 septets /segment	(00000000)	• GSM 7-bit data • not concatenated message max 160 septets • concatenated message max 153 septets /segment • custom character set will be applied if configured	Not included
(00010000)	GSM 7bit Class 0		(00000000)		Included, If TP-DCS 4th bit is set
(00010001)	GSM 7bit Class 1		(00000000)		
(00010010)	GSM 7bit Class 2		(00000000)		
(00010011)	GSM 7bit Class 3		(00000000)		
(00xx00xx)	GSM 7bit others		(00000000)		
(00001000)	UCS2 No Message Class	• UCS2(16-bit data) • not concatenated message max 140 octets • concatenated message max 134 octets /segment	(00001000)	• (UCS2(16-bit code) • not concatenated message max 140 octets • concatenated message max 134 octets /segment	Not included
(00011000)	UCS2 Class 0		(00001000)		Included, If TP-DCS 4th bit is set
(00011001)	UCS2 Class 1		(00001000)		
(00011010)	UCS2 Class 2		(00001000)		
(00011011)	UCS2 Class 3		(00001000)		
(00xx10xx)	UCS2 others		(00001000)		
(00000100)	8bit data No Message Class	• 8-bit data • not concatenated	(00000100)	• 8-bit data • not concatenated	Not included

MO			AT		
[TP-DCS] value(bit)	**meaning**	**TP-UD/TP-UDL**	**[data_coding] value(bit)**	**Short_message / sm_length**	**Is Optional TLV dest_addr_subunit included in Deliver_sm**
(00010100)	8bit data Class 0	• message max 140 octets • concatenated message max 134 octets /segment	(00010100)	• message max 140 octets • concatenated message max 134 octets /segment	
(00010101)	8bit data Class 1		(00010101)		
(00010110)	8bit data Class 2		(00010110)		
(00010111)	8bit data Class 3		(00010111)		
(00xx01xx)	8bit data others		(00xx01xx)		
other than above patterns (xxxxxxxx)			other than above patterns (xxxxxxxx)		

Note: x is an arbitrary digit; it can be 0 or 1.

3. The above conversion is also applicable for the below scenarios:
   - MO-AT
   - MO-AT-Store
   - MO-Store-AT
4. If the new parameter `hubdcscharcodingconversion` and `rtrdcscharcodingconversion` are set to `japan` and `MO Modifier` is also configured to change the value of TP-DCS then value configured in Data Coding Scheme (DCS) of `MO Modifier` will reflect in MO-AT Scenario.

   Note: It is not recommended to use the MO Modifier if the DCS conversion feature set to `japan` (i.e. '`hubdcscharcodingconversion`' and '`rtrdcscharcodingconversion`' set to '`japan`'). However, if one uses the MO Modifier and the DCS conversion feature with `japan` together, the behavior (i.e. user data conversion and character set conversion) is not as per the expectation.
5. If the 4th bit of TP-DCS is set, then the message class information will be stored in the optional TLV `dest_addr_subunit` of deliver_sm, as per the mapping below:

Message Class		Dest_addr_subunit
bit 0 and 1 of TP-DCS	**Message class meaning**	**Value**
0	Class 0	1
1	Class 1: Default meaning: ME-specific.	2
2	Class 2:(U) SIM specific message	3

Message Class		Dest_addr_subunit
bit 0 and 1 of TP-DCS	**Message class meaning**	**Value**
3	Class 3: Default meaning: TE specific (see 3GPP TS 27.005 [8])	4

Note: It is applicable for SMPP 3.4 and 5.0 only; for SMPP 3.3 it will not send the optional TLV and message class information.

3.13 Bind Error Handling

The HUB classifies bind errors as:

- Temporary with fallback
- Temporary without fallback
- Permanent

The classification of a bind error and the session model of the application determine how the HUB handles the error:

Session Model	Temporary Error with Fallback	Temporary Error without Fallback	Permanent Error
Distribute—Service centers are only used when they are needed. The number of inside sessions, service center nodes, and service centers depends on the relationship between the outside and inside bandwidth (the window size and the number of sessions).	After a bind attempt fails with a temporary error with fallback, the HUB will attempt to bind to an unused service center. If all bind attempts fail, the HUB stops attempting to bind.	If a bind attempt fails with a temporary error without fallback, the HUB: **1.** Stops attempting to set up an inside session with the service center **2.** Sends a NACK to the application **3.** Removes the related outside session	When a bind attempt fails with a permanent error, the HUB blacklists the termination point for the application. The HUB will not allow that application to attempt to bind to that termination point again until the blacklist is cleared. This behavior is the same for all session models. By default, there are no permanent bind errors in the HUB. You must customize the HUB's error mapping to make an error permanent.
Replicate—The HUB sets up a session for every service center that is related to an application.	Fallback is not possible with the replicate session model because in this model, there is no unused service center that the HUB can attempt to bind to. Therefore, the behavior is the same as if the bind attempt fails with a temporary error without fallback.	If a bind attempt fails with a temporary error without fallback, the HUB: **1.** Stops attempting to set up an inside session with the service center **2.** Sends a NACK to the application	

Session Model	Temporary Error with Fallback	Temporary Error without Fallback	Permanent Error
		3. Removes the related outside session	
Inside only—Not related to an outside bind request; however, when the application is activated, the HUB sets up sessions and sends bind requests to the service center (the number of sessions and the service center that is used depend on the bandwidth that is required).	If only one service center is required, then after a bind attempt fails with a temporary error with fallback, the HUB will attempt to bind to an unused service center. If all bind attempts fail, the HUB stops attempting to bind for `hubreconnectdelay`, time, then starts attempting to bind again.	If only one service center is required, then after a bind attempt fails with a temporary error without fallback, the HUB stops attempting to set up an inside session with the service center.	

For detailed information about inside sessions, outside sessions, and session models, refer to *Terminology*, *Application Routing Concepts*, and *Session Management*.

Customizing Bind Error Classification

You can customize the classification of a bind error using the HUB's error mapping functionality, as described in *Error Mapping*.

For example, assume that if a service center rejects a bind attempt with SMPP error 14 because the application password is wrong, the application should not continue to retry the bind. Therefore, this error should be classified as permanent, to ensure that the HUB will blacklist the service center termination point for the application.

SMPP error 14 is mapped to HUB normalized error `authenticationFailure`, so to customize the error classification, you:

1. Modify the default SMPP reverse error map or create a new SMPP reverse error map
2. Change the classification of `authenticationFailure` to permanent
3. If you created a new error map, assign it to the desired application

Bind Error Handling Example

This figure illustrates a scenario with two applications, both of which use the distribute session model. In this scenario, the inside/outside bandwidth is the same.

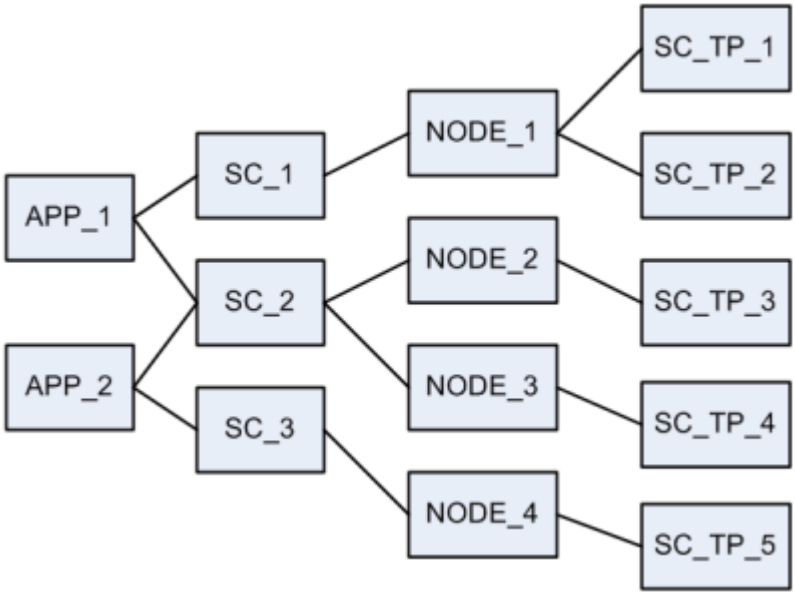


Figure 19: Bind error handling example

When application APP_1 binds, the HUB sets up an inside session with service center SC_1:

- If the bind fails with a temporary error with fallback, the HUB will attempt to set up a session with SC_2.
- If the bind fails with a temporary error without fallback, the HUB will stop attempting to set up a session, send a NACK to the application, and remove the outside session.
- If the bind fails with a permanent error, the HUB will blacklist SC_TP_1 for APP_1 and attempt to set up a session with SC_TP_2.

Clearing the Bind Error Blacklist

To remove a termination point from the blacklist, deactivate and then reactivate the termination point in the MGR. To clear the blacklist for an application, deactivate and then reactivate the application in the MGR.

3.13.1 Retrieving the Current Blacklist

You can retrieve the current blacklist via SNMP, using the following command:

```
$ tp_walk --tp_hub bindBlackListTable
```

Sample output:

```
bindBlackListIndex.1 = INTEGER: 1
bindBlackListApplicationIndex.1 = INTEGER: 1
bindBlackListApplicationName.1 = STRING: 77777UCPreplicate
bindBlackListScTpIndex.1 = INTEGER: 1
bindBlackListScTpName.1 = STRING: UCP7772replicate
```

3.14 Error Mapping

To normalize error messages among applications, SMSCs, and Mobile Messaging components, the HUB maps the error code and the error text to its own internal, normalized errors. The HUB maps errors in two ways:

- Reverse error mapping—Maps error codes that are generated by applications or SMSCs to the HUB's internal, normalized errors
- Forward error mapping—Maps the HUB's internal, normalized errors to error codes that are understood by applications and SMSCs

Figure 20: Forward and reverse error mapping

There are default forward and reverse error mapping tables for the SMPP, UCP, and CIMD application protocols. You can use the MGR:

- To customize the mapping in the default tables or to add new tables, as described in *Configuring Error Mapping Tables*
- To assign custom tables to specific applications and/or SMSCs, as described in *Assign Custom Error Mapping Tables to Applications and Service Centres*

New tables are automatically populated with the default error mapping for the selected protocol.

Refer to *Error Mapping and Normalization* for the default error mapping for SMPP, UCP, and CIMD.

Custom Reverse Error Mapping

If the HUB cannot find an entry for an error code in a custom reverse error mapping table, it will map the error code to the normalized error that is configured for unknown incoming errors.

You can further customise the reverse error mapping for applications or SMSCs that send errors with the same error code but different error text by using reverse error text codes, as described in *Add a Reverse Error Text Code*.

Custom Forward Error Mapping

Before the HUB can determine which forward error map is assigned to a particular application, the application must be authenticated. Therefore, for errors that are related to application authentication fields (such as system ID, system type, password, and address range), the HUB will always use the values that are in the default error map for that application's protocol. You can modify application authentication errors in a custom forward error map, but the modification will not have any effect.

3.14.1 Custom Error Mapping Example

In this error mapping example, a UCP application sends a submit message via the HUB/RTR to an SMSC (see diagram).

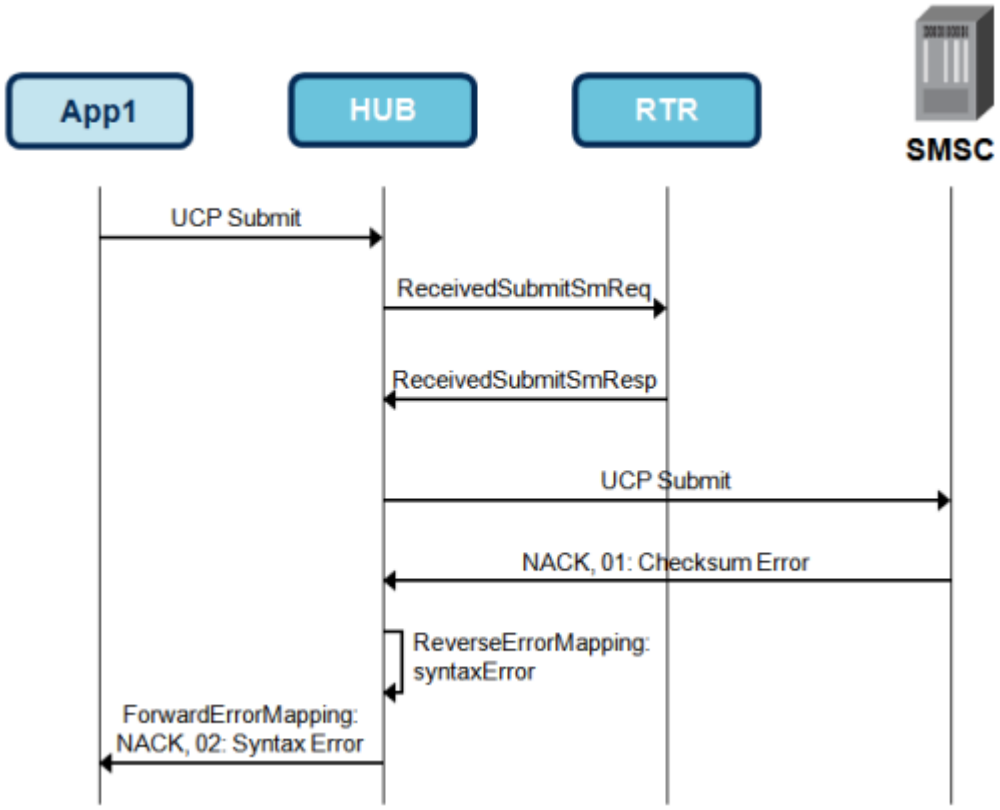


Figure 21: Custom error mapping example

The UCP application App1 sends a submit request (UCP51) to the HUB, which successfully parses it. The HUB forwards this submit request to the SMSC, based on the information of the RTR.

The SMSC supports UCP but it responds with a NACK (error code 01 and text "Checksum error"). The (custom) reverse error table for this SMSC on the HUB contains an entry that translates this error message into the normalised error `syntaxError`.

The default forward error table associated with the originating application then translates the normalised error `syntaxError` into the configured corresponding error value (error code 02 and text "Syntax Error").

3.14.2 Configuring Error Mapping Tables

This section describes the configuration of forward and reverse error mapping tables for the HUB. Forward error mapping tables map the HUB's internal error codes to error responses that are understood by applications and SMSCs. Reverse error mapping tables map error messages that are generated by applications or SMSCs to the HUB's internal error codes.

3.14.2.1 Modify a Forward Error Mapping Table

To modify a forward error mapping table:

1. In the left navigation bar, select **SMS Applications ➤ Error Mapping ➤ Forward Error Mapping**. The Forward Error Mapping tab appears.

Forward Error Mapping

ID	ST	Name	Last Updated
1		SMPP	2011-03-10 16:35:16
2		UCP	2011-03-10 16:35:16
3		CIMD	2011-03-10 16:35:16
4		Reserved4	2011-03-10 16:35:16
5		Reserved5	2011-03-10 16:35:16
6		Reserved6	2011-03-10 16:35:16
7		Reserved7	2011-03-10 16:35:16
8		Reserved8	2011-03-10 16:35:16
9		Reserved9	2011-03-10 16:35:16

Figure 22: Forward error mapping tables

2. Click the name of the forward error mapping table that you want to modify.
   The error mapping table's tab appears.

Error Codes

Error Code	New Code	Text
intSystemError	8	Internal system error
intShuttingDown	8	Session shutting down
intMxpFailure	8	Error routing message
intMxpTimeout	69	Error routing message
intTxFailure	69	Error sending message
intTemporaryError	100	Temporary error sending message
intPermanentDestError	101	Permanent error sending message
intPermanentMsgError	102	Permanent error sending message
intDestinationNotAvailable	69	Destination not available
intSourceNotAvailable	69	Source not available
intThroughputExceeded	88	Throughput exceeded
intWindowSizeViolation	255	Window size exceeded

Figure 23: Sample forward error mapping table

Note: The default error mapping tables (those with IDs 1-3) and the reserved error mapping tables (those with IDs 4-9) cannot be deleted. Also, the reserved tables cannot be modified.

3. In the Error Codes section, click the name of the error mapping table entry that you want to modify.
   The table entry's tab appears.
4. Optionally enter a description of the error mapping table entry in the **Description** box.
5. To modify the error code that should be sent to the application or SMSC, modify the code in the **Replace Error Code** box.
6. To modify the optional error text that should be sent to the application or SMSC, modify the text in the **Replace Error Text** box.
7. Click **Save.**
   The MGR saves the entry and closes the tab.
8. On the error mapping table's tab:
   - Click another table entry to modify it, or
   - Click **Save** to save the error mapping table

3.14.2.2 Add a Forward Error Mapping Table

To add a new forward error mapping table:

1. In the left navigation bar, select **SMS Applications ➤ Error Mapping ➤ Forward Error Mapping**.
   The Forward Error Mapping tab appears.
2. Click **Add New**.
   A new Forward Error Mapping tab appears.
3. In the **Name** box, enter a name for the table.
4. In the **Description** box, optionally enter a description of the table.
5. From the **Application Protocol** list, select the application protocol for the table:
   - UCP
   - SMPP
   - CIMD
6. Click **Save.**
   The MGR saves the table and closes the tab.

   The MGR automatically creates the table entries, based on the default error mapping for the application protocol that you selected. Entries cannot be deleted from the table.
7. On the Forward Error Mapping tab, click the name of the table that you just created.
   The table opens in a new tab.
8. Modify the table entries as desired.
9. When you are finished, click **Save**.
   The MGR saves the table and closes the tab.
10. Activate the table.

3.14.2.3 Modify a Reverse Error Mapping Table

To modify a reverse error mapping table:

1. In the left navigation bar, select **SMS Applications ➤ Error Mapping ➤ Reverse Error Mapping**.
   The Reverse Error Mapping tab appears.


Reverse Error Mapping

ID	ST	Name	Last Updated
1		SMPP	2011-03-10 16:35:15
2		UCP	2011-03-10 16:35:15
3		CIMD	2011-03-10 16:35:15
4		Reserved4	2011-03-10 16:35:15
5		Reserved5	2011-03-10 16:35:15
6		Reserved6	2011-03-10 16:35:15
7		Reserved7	2011-03-10 16:35:15
8		Reserved8	2011-03-10 16:35:15
9		Reserved9	2011-03-10 16:35:15

Figure 24: Reverse error mapping tables

2. Click the name of the reverse error mapping table that you want to modify.
   The error mapping table's tab appears.

Error Codes

Error	Normalised	Class	Action
1	invalidMsgLength	Message Permanent Error	None
2	invalidCommandLength	Message Permanent Error	None
3	invalidCommandId	Message Permanent Error	None
4	invalidBindStatusForCmd	Message Permanent Error	None
5	alreadyLoggedIn	Destination Temporary Error	None
6	invalidPriorityFlag	Message Permanent Error	None
7	invalidRegDeliveryFlag	Message Permanent Error	None

Figure 25: Sample reverse error mapping table

Note: The default error mapping tables (those with IDs 1-3) and the reserved error mapping tables (those with IDs 4-9) cannot be deleted. Also, the reserved tables cannot be modified.

3. In the Error Codes section, click the name of the error mapping table entry that you want to modify. The table entry's tab appears.
4. Optionally enter a description of the error mapping table entry in the **Description** box.
5. To modify the HUB internal error code to use, select a code from the **Normalised Error Code** list.
6. To modify the error class, select a type of error from the **Error Class** list:

Option	Description
Source Error	Errors that the HUB detects in the incoming message
Routing Error	Errors that occur while routing the message, such as a timeout on the MXP path or no routing rule for the message
Destination Temporary Error	Errors reported by the destination, for which a fallback path should be attempted (if available)
Destination Permanent Error	Errors reported by the destination for an invalid recipient; no fallback will be attempted
Message Permanent Error	Errors reported by the destination for an invalid message; no fallback will be attempted

7. From the Error Action list, select the action to take:

Option	Description
None	Do not take any action
Fallback to another destination	Fallback to another service centre that is assigned to this application's service class (does not have any effect if the application session model is replicate)

8. Click **Save.**
   The MGR saves the entry and closes the tab.
9. On the error mapping table's tab:
   - Click another table entry to modify it, or
   - Click **Save** to save the error mapping table

3.14.2.4 Add a Reverse Error Mapping Table

To add a new reverse error mapping table:

1. In the left navigation bar, select **SMS Applications ➤ Error Mapping ➤ Reverse Error Mapping**.
   The Reverse Error Mapping tab appears.
2. Click **Add New**.
   A new Reverse Error Mapping tab appears.
3. In the **Name** box, enter a name for the table.
4. In the **Description** box, optionally enter a description of the table.
5. From the **Application Protocol** list, select the application protocol for the table:
   - UCP
   - SMPP
   - CIMD
6. From the **Default Error Code** list, select the internal error code to use if the application or SMSC sends an error that the HUB does not recognize.
7. From the **Default Error Class** list, select the default error class to use if the application or SMSC sends an error that the HUB does not recognize:

Option	Description
Source Error	Errors that the HUB detects in the incoming message
Routing Error	Errors that occur while routing the message, such as timeout on the MXP path, no routing rule for the message, and so on
Destination Temporary Error	Errors reported by the destination, for which a fallback path should be attempted (if available)
Destination Permanent Error	Errors reported by the destination for an invalid recipient; no fallback will be attempted
Message Permanent Error	Errors reported by the destination for an invalid message; no fallback will be attempted

8. From the **Default Error Action** list, select the default to execute if the application or SMSC sends an error that the HUB does not recognize:
   - None
   - Fallback to another destination
9. Click **Save.**
   The MGR saves the table and closes the tab.
10. On the Reverse Error Mapping tab, click the name of the table that you just created.
    The table opens in a new tab. It contains the default error mapping for the application protocol that you selected.
11. Modify the table entries as desired.
12. When you are finished, click **Save**.
    The MGR saves the table and closes the tab.

3.14.2.5 Add a Reverse Error Text Code

To accommodate applications or SMSCs that send errors containing the same error code but different text, you can add reverse error text codes to the entries of a reverse error mapping table. To add a reverse error text code:

1. In the left navigation bar, select **SMS Applications ➤ Error Mapping ➤ Reverse Error Mapping**.
   The Reverse Error Mapping tab appears.
2. Click the name of the reverse error mapping table that you want to modify.
   The error mapping table's tab appears.
3. Click the name of the table entry that you want to modify.
   The table entry's tab appears.
4. In the Error Text Codes section, click **Add New**.
   A new Reverse Error Text table tab appears.
5. From the **Mapping Table** list, select the reverse error mapping table (defaults to the table that you clicked previously).
6. In the **Error** box, enter the error code to map (defaults to the table entry that you clicked previously).
7. In the **Description** box, optionally enter a description of the error text code.
8. From the **Normalised Error Code** list, select the HUB's internal error to which to map the application/SMSC error.
9. From the **Error Class** list, select the type of error:

Option	Description
Source Error	Errors that the HUB detects in the incoming message
Routing Error	Errors that occur while routing the message, such as timeout on the MXP path, no routing rule for the message, and so on
Destination Temporary Error	Errors reported by the destination, for which a fallback path should be attempted (if available)
Destination Permanent Error	Errors reported by the destination for an invalid recipient; no fallback will be attempted
Message Permanent Error	Errors reported by the destination for an invalid message; no fallback will be attempted

10. From the **Error Action** list, select the action:
    - None
    - Fallback to another destination
11. In the **Message** box, enter the error text that the application or SMSC will return.
12. Click **Save.**
    The MGR saves the error text code and closes the tab.

3.14.3 Assign Custom Error Mapping Tables to Applications and Service Centres

To assign custom forward and reverse error mapping tables to a service centre:

1. In the MGR, go to **Environment ➤ SMSC ➤ Service Centre**.
2. Select the desired service centre.
3. Select the desired error mapping tables for:

   - **SMPP Forward Error Map**
   - **SMPP Reverse Error Map**
   - **UCP Forward Error Map**
   - **UCP Reverse Error Map**
   - **CIMD Forward Error Map**
   - **CIMD Reverse Error Map**

To assign custom forward and reverse error mapping tables to an application:

1. In the MGR, go to **SMS Applications ➤ Applications**.
2. Select the desired application.
3. Select the desired forward and reverse error mapping tables. The tables that are available depends on the selected application protocol.

3.14.4 Relaying SMPP 5.0 error codes to SMPP 3.3/3.4 interface

The new error codes introduced in the SMPP 5.0 specification, -i.e. ESME_RSERTYPUNAUTH, ESME_RPROHIBITED, ESME_RSERTYPUNAVAIL and ESME_RSERTYPDENIED, are not directly reported by the HUB or the RTR, but these can be reported indirectly via the error mapping feature, depending on the relevant SMPP reverse and forward error mapping table configurations.

Hence while relaying an error response received on a SMPP 5.0 interface and containing any of the above error codes towards a SMPP 3.3/3.4 interface, HUB converts these error codes to a generic error code compatible with SMPP 3.3/3.4. The generic error code value is taken from the semi-static parameter "*hubcommonerrorcodeforsmpp50errormapping*".

The routing paths for which this functionality can be relevant are AO-AT, AT-AT and AT-AO.

3.15 Application-Specific Charging Information

When the HUB is implemented in a system with the Prepaid Billing Controller (PBC), you can use the application-specific charging feature to rate messages that are destined to certain applications with a certain number of charging units.

The application-specific charging information allows you to indicate:

- How many charging units should be used to rate messages that are destined to an application.
- How many charging units should be used to rate messages that are sent by an application (these messages are typically reverse-charged to the recipient).

The application-specific charging feature can be used in any system that uses SCAP Diameter as a real-time charging interface and/or that uses the FCDR format.

To configure the application-specific charging feature:

1. In **SMS Applications ➤ Applications**, create a new application or open an existing application that will use originator and/or recipient charging units.

- To use charging units for messages sent by the application, select **Use Originator Charging Units** and enter the number of originating charging units to use in **Originator Charging Units** (0-65,535). By default, the charging units are not selected or specified.
- To use charging units for messages sent to the application, select **Use Recipient Charging Units** and enter the number of originating charging units to use in **Recipient Charging Units** (0-65,535). By default, the charging units are not selected or specified.

2. In **Routing ➤ EC Applications ➤ Applications**, create a new EC application for the PBC or open the existing EC application for the PBC.
   - To include charging units for messages sent by the application in the ECI message to the PBC, select **Inc. Originating App. Charging Units**. By default, they are not included in the ECI message.
   - To include charging units for messages sent to the application in the ECI message to the PBC, select **Inc. Terminating App. Charging Units**. By default, they are not included in the ECI message.
3. In **Billing ➤ Profile**, create a new billing profile or open an existing billing profile; ensure that the file format is FCDR.
   - To include the charging units for messages sent by the application, select **Inc. Originating App. Charging Units**. By default, they are not included in the billing profile.
   - To include charging units for messages sent to the application, select **Inc. Terminating App. Charging Units**. By default, they are not included in the billing profile.

Chapter 4

Entities and Attributes

Topics:

4.1 Introduction

The HUB is responsible for processing session management requests and load balancing incoming messages according to the defined configuration. The configuration operates on the external entities in the SMS environment: the routing entities (such as the RTR), applications, and/or termination points (such as an SMSC).

Several entities can be used as operands of routing and counting rules. Each entity contains a unique specification of parameters required by the HUB to use the entity in the total system configuration.

This chapter describes how to create and configure AO- and AT-specific routing entities. The HUB depends upon the RTR and always requires a RTR to provide its functionality. Because the HUB uses the RTR's routing rules, many entities are described in the RTR Operator Manual.

In the RTR, the following entities are defined:

- Application-related entities:
  - Application (SMS applications that can send and/or receive SMS messages)
  - Groups (to enable advanced throughput control per group of applications)
  - Categories (to enable application labelling across groups)
  - Service classes (to enable quality of service differentiation)
- SMSC-related entities:
  - SMSC group specification of SMSCs (for load balancing purposes)
  - Service centre specification of IP parameters (only applies in relationship with a HUB)
  - Service centre node specification (only applies in relationship with a HUB)
  - Termination point specification per service centre node (only applies in relationship with a HUB)
- Country
- Network

Entities are defined using the MGR GUI.

4.2 Entity Relationships

The diagram shows the the relationships and cardinality between entities.

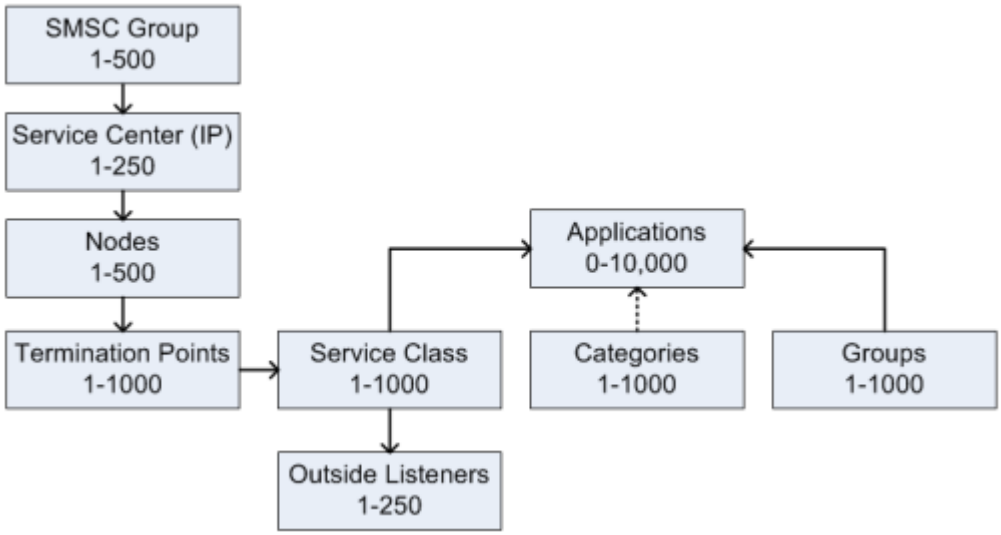


Figure 26: Entity relationships

You use the MGR interface to create and delete entities. Refer to the MGR Operator Manual for details.

4.2.1 Inside sessions IPv6 support

HUB will support making connections with Service Centre using IPv4 and IPv6 Addresses.

- If Service Centre Node IP is of type IPv4 then HUB Internal Address of type IPv4 must be provisioned otherwise no inside session will be attempted to Termination Points on Service Centre Node.
- Similarly if Service Centre Node IP is of type IPv6 then HUB Internal Address of type IPv6 must be provisioned otherwise no inside session will be attempted to Termination Points on Service Centre Node.

Same will hold true for Distributed and Replicated session model. HUB will support connections with Service Centre using both IPv4 and IPv6 address.

4.3 Outside Listener Entity

The outside listener entity is the HUB TCP listen port for SMS application protocols. Each outside listener is associated with an application protocol (UCP, SMPP, or CIMD2). There can be more outside listeners configured for the same application protocol.

Each outside listener can be associated with one service class. Before using applications, at least one outside listener must be defined.

Note: If your system is handling AO-AO traffic and you intend to deactivate a termination point for a service class, first deactivate all outside listeners for that service class. This prevents errors from occurring when connected applications continue to send AO-AO traffic after the termination point was deactivated.

For information about configuring outside listener entities, refer to the MGR Operator Manual.

Up to 250 outside listener entities can be defined.

Note: The outside listener session limit (max sessions) is per HUB. The session limit is considered to be a HUB dimensioning parameter and must be seen separately from the availability of other HUB devices.

4.3.1 Outside Listener IPv6 support

HUB will support listening for Application connection over both IPv4 Address and IPv6 Address.

- If Outside Listener is of type IPv6, then it will listen on IPv6 External address and port configured in Outside Listener.
- If Outside Listener is of type IPv4, then it will listen on IPv4 External address and port configured in Outside Listener.

Two outside listener can have same port if they listener on different address type.

Note: If HUB External IPv6 address is not configured then HUB will not listen for outside session on IPv6.

4.3.2 Provisioning HUB External IP

HUB devices use `hubPropIpAddressOwnExternal` and `hubPropIpv6AddressOwnExternal` parameters to start the outside listener. Value of these parameters are configurable on the MGR GUI. While adding a new HUB device, the fields "**External IPv4 Address" and** "**External IPv6 Address"** appears on the GUI. Enter here the respective IP addresses/hostnames that will be used by the HUB device as its external IP(s).

The following should be ensured while provisioning the "**External IPv4 Address" or** "**External IPv6 Address"** on the MGR GUI:

1. These parameters are mandatory only when using HUB Multi-Instance.
2. If a shared IP address is used between two HUB's (i.e. HUB IP failover), then this field should keep the default value `0.0.0.0`.

   Note: This point is valid only for provisioning the"**External IPv4 Address"**.

3. When configured, the HUB will only accept application connections on these addresses.

For more information on configuring the devices on MGR GUI, refer to the "Configuring Devices" section of the MGR Operator Manual. For more information on the HUB External IPv4 and HUB External IPv6, refer to the chapter *Provisioning HUB External IP*.

4.3.3 DNS Query Mechanism

This section describes the DNS lookup mechanism when a hostname is configured in a particular field. If the hostname is configured, then the DNS query will be performed. Following steps explain how DNS query will be performed:

1. DNS lookup is performed for all the configured hostnames.
2. If the specific type of address that is "AAAA(for IPv6)" and "A(for IPv4)" is not found on lookup, then no IP address is set for that parameter. For example:

   a. The field "**External IPv6 Address"** requires an IPv6 address and it won't be set until an IPv6 address is returned through the DNS query.

   b. The IP address of Service center node can accept an IPv4/IPv6 address or a hostname. While doing DNS query, IPv6 address shall be preferred over IPv4 address.
   c. If the DNS resolves the hostname to the correct type, that is IPv4/IPv6 address, but the IP address is not configured on the system, then it will not open the corresponding port.

3. The a-synchronization DNS query interval is controlled by the semi-static parameter `dnsfailtimeout`. If the value of this parameter is “0” no further DNS queries shall be performed. However, when a connection breaks the DNS query is performed once again but for only once.
4. As long as the DNS query does not get resolved, a log message is printed to indicate that the DNS query is not successful.
5. DNS module maintains a TTL value for the resolved host addresses. Once the TTL value expires for the hostname, the DNS lookup is performed again to get the new IP address.
6. If user configures a hostname which does not exist in DNS sever, then in that case there will be infinite number of DNS queries performed . To avoid this situation, a maximum of 50 DNS queries are performed.

4.4 Application Entity

The application entity is the RTR and HUB’s definition of an application. Before routing rules can route messages to an application, it must be defined and activated. You can create application templates for flexible default assignments.

UCP, SMPP, and CIMD2 applications are available.

Note: License settings determine which application types are available on a system.

The originator address of Incoming application-originated (AO) messages is typically a short number; however, it may be alphanumeric. To associate incoming AO messages with defined applications, you can set up to 10 alphanumeric aliases for each application (using the MGR). The RTR/HUB can then match a message to an application using the short number or the alias. This functionality ensures that these AO messages are handled correctly and that the RTR can accurately update counters for the application.

For more information about configuring applications, refer to the MGR Operator Manual.

Up to 1000 application entities can be defined. With an extended application license, up to 10,000 applications can be defined.

The HUB has two separate outside window size configuration options "**Outside SMPP Transmit window size**" and "**Outside SMPP Receive window size**". The range for both window size values is between 1 to 1024.

The default value for both Outside SMPP window size values is 255.

The range for Inside SMPP window size is from 1 to 1024.

In case the Throughput AT Maximum parameter is configured as zero (0) for an application and the semi-static parameter `discardoutboundatmsgwhenmaxthroughputiszero` is set to "true", then all outbound AT messages and notifications destined for that application will be immediately discarded by the RTR with a permanent error.

This logic would also apply when an AT message or notification is supposed to be first stored in the AMS, which would subsequently initiate a delivery attempt towards an application. In such a case, before attempting to store the message or notification in the AMS the RTR will first check whether the

maximum AT throughput values of the destination application is set to 0 and whether the parameter `discardoutboundatmsgwhenmaxthroughputiszero` is set to "true". If both these conditions are met, the RTR will immediately discard the AT message or notification with a permanent error instead of storing it in the AMS.

4.5 Application Group Entity

The application group entity is the RTR's grouping mechanism for SMS applications. It enables throughput control across all SMS applications within the group.

Before applications can be used, at least one application group must be defined. An application can only be a member of one application group. If differentiated application group throughput control is not required, all SMS applications can be placed in a single default application group.

For information about configuring application group entities, refer to the MGR Operator Manual.

Up to 1000 application group entities can be defined.

4.6 Service Class Entity

The service class entity is the Quality of Service mechanism for SMS applications. The service class:

- Links outside listeners to specific termination points (TCP connect ports)
- Specifies the default session model used by applications when connecting to an outside listener
- Introduces an extra level of throughput control across all SMS applications within the service class

Before using applications, an outside listener and a service class must be defined.

On the application level, the default session model can be overruled by specifying the session model. If differentiated Quality of Service is not required, all SMS applications can be placed in a single default service class.

For information about configuring service class entities, refer to the MGR Operator Manual.

Up to 1000 service class entities can be defined.

4.7 Device Management

Before using the HUB as an application router in the network, the HUB device must defined and activated in the MGR.

For information about configuring devices, refer to the MGR Operator Manual.

Note: Adding, removing, activating, and deactivating devices is subject to user authorization levels. If you have insufficient authorization to perform your assigned configuration tasks, contact your local system administrator.

Chapter 5

Load Distribution

Topics:

5.1 Introduction

The HUB provides session-based load distribution toward SMSC termination points. Multiple application sessions established to one HUB listener port can be distributed over multiple SMSC listeners running on multiple SMSC nodes.

The advantage is that application traffic becomes less dependant on a single SMSC listener. By distributing sessions over multiple SMSC listeners running on multiple SMSC nodes, SMSC capacity can be used in an optimal manner.

Session load balancing works in a round-robin fashion:

- If the application disconnects the session, the HUB also disconnects the corresponding sessions toward the SMSC.
- If the SMSC disconnects the session, the HUB also disconnects the corresponding sessions toward the application.

Up to 12,000 TCP/IP connections (default 1000) can be established per HUB.

Note: The total number of TCP/IP connections includes all outside sessions and all inside sessions.

5.2 Example

In this example, two types of SMS applications connect to different HUB ports, each with 10 sessions:

Application	HUB port	Service class	Number of sessions	SMSC node / port
123	8000	1	4	3 / 8002
123	8000	1	3	4 / 8002
123	8000	1	3	5 / 8002
456	8001	2	5	7 / 8003
456	8001	2	5	8 / 8003

Chapter 6

Throughput Control

Topics:

6.1 Introduction

The RTR and the HUB can provide throughput control for all application-originating (AO) and application-terminating (AT) traffic. Throughput is controlled by the RTR and is configured in the HUB. Fine tuning of throughput control is recommended to avoid congestion of SMSC nodes or applications.

Up to 4000 messages per second can be handled per HUB node.

6.2 Adjusting Throughput

Throughput can be controlled at several points in the dynamic configuration.

Item	Property	Description
AO routing rule (AOR)	Maximum	Maximum messages per second allowed for the primary destination of the rule (default 65,535)
SMS application group	Throughput AO	Maximum messages per second allowed from applications in the group to the HUB (default 65,535)
	Throughput AT	Maximum messages per second allowed from the HUB to applications in the group (default 65,535)
SMS application	Throughput AO Maximum	Maximum messages per second allowed from the application to the HUB (default 65,535) Our system will try to use the "Maximum" throughput if there are still spare resources to do so.
	Throughput AO Committed	Committed messages per second allowed from the application to the HUB (default 65,535) Our system commits to support the "Committed" throughput.

Item	Property	Description
	Throughput AT Maximum	Maximum messages per second allowed from the HUB to the application (default 65,535) Our system will try to use the "Maximum" throughput if there are still spare resources to do so.
	Throughput AO Committed	Committed messages per second allowed from HUB to the application (default 65,535) Our system commits to support the "Committed" throughput.
	AO Delay Time	Number of seconds to delay AO messages
Service class	AT Throughput	Maximum messages per second allowed from the HUB toward the associated applications (default 65,535)
	AO Throughput	Maximum messages per second allowed from the associated applications to the HUB (default 65,535)
SS.7 SMSC	Throughput	Maximum messages per second allowed from the HUB to the SMSC (default 65,535)
Service centre	AO Throughput	Maximum messages per second allowed from the HUB to the service centre (default 65,535)
Service centre node	AO Throughput	Maximum messages per second allowed from the HUB to the service centre node (default 65,535)
Service centre node termination point	AO Throughput	Maximum messages per second allowed from the HUB to the termination point (default 65,535)

6.3 Performance When AO Throughput is Exceeded

If an application exceeds the allowed maximum AO throughput, the RTR will temporarily buffer messages from that application for 3 seconds. After this time, the RTR will negatively acknowledge (NACK) the messages, due the to time-out that will occur.

In general, each application should be configured such that:

```
3-second delay * throughput > window size * number of sessions
```

To ensure an optimal smoothing of the capacity usage, the RTR performs throughput regulation in slices of 100 milliseconds, using the proportional part of the throughput.

6.4 Configuring AO Delay

The HUB can delay AO messages for a period of time (up to 60 seconds) before continuing to process them. This delay can be configured on a per-application basis, using the application's **AO Delay Time** parameter in the MGR interface. The delay time should be set to 0 for high-throughput applications.

Note: Implementing the AO delay increases the HUB's memory usage. For 1000 messages, the HUB's memory usage increases by approximately 8 MB. This value increases proportionally with the number of messages.

Only the following types of messages will be delayed:

Protocol	Message Type
UCP	• UCP 01 (CALL INPUT) • UCP 30 (SMS MESSAGE TRANSFER) • UCP 51 (SUBMIT SHORT MESSAGE)
SMPP	• SUBMIT_SM • DATA_SM
CIMD2	CIMD2 03 (SUBMIT MESSAGE)

If the delay time is set to anything other than 0, the HUB will reject the following types of messages from the application:

- UCP 02 (MULTIPLE ADDRESS CALL INPUT)
- SMPP SUBMIT_MULTI
- CIMD2 03 (SUBMIT MESSAGE) with more than one recipient

The maximum number of messages that can be delayed in the HUB at one time for all configured applications is controlled by the `hubmaxqueuedmessages` parameter in the semi-static configuration file.

Setting the delay time does not change the way the HUB handles window size. The HUB continues to discard all requests that are received outside the window size for a session.

The application's outside window size should be set to 1 for the delay time to be effective.

The `applicationAoCurrentDelayed` gauge shows the current number of messages in an application's delayed queue. The `hubAoCurrentDelayed` gauge shows the current number of messages that are delayed in the HUB.

Chapter 7

Failover Mechanism

Topics:

7.1 Introduction

A redundant HUB setup can be configured in the following ways:

- Using a third-party IP load balancer. In this configuration, the IP load balancer distributes the traffic from all applications in the IP domain to two or more HUBs. All HUBs are in active mode and the IP load balancer manages failover.
- Using HUB IP failover functionality (two-node HUB failover configuration). This configuration involves one HUB connecting the applications in the IP domain to the SMS network and another HUB on hot standby in case the active HUB becomes unavailable due to software, hardware, or network problems.

Note: The HUB IP failover mechanism is a licensed feature.

This chapter describes a two-node HUB configuration.

7.2 Network Setup

Only one HUB is active at a time. Data flow from and to the IP domain therefore goes *either* to one HUB *or* the other. As long as the primary HUB (HUB A in the diagram below) is active, there will be no IP communication between the other network nodes and the secondary HUB (B).

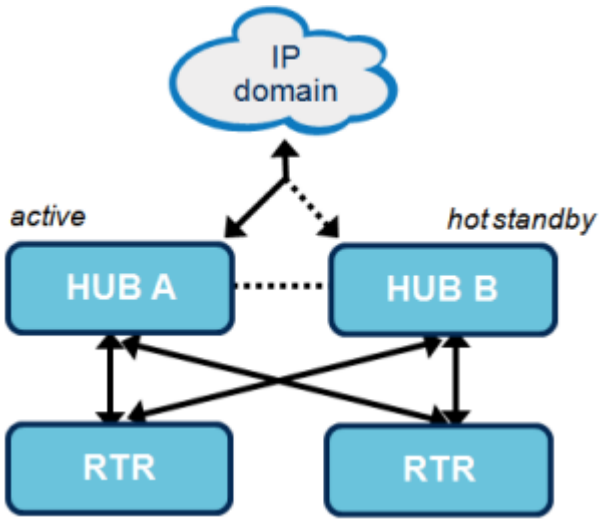


Figure 27: Single active HUB

Both HUBs are connected to the same Ethernet (same broadcast domain), and have an IP address on that Ethernet that allows communication to the application servers in the IP domain. Additionally, a common virtual IP address that is communicated to the SMS applications.

At initialisation this primary HUB (the node with the lowest host identifier and process identifier of the two) sets up a virtual IP address with which all applications in the IP domain will communicate. Additionally it will start sending out heartbeats. The secondary HUB is in "hot standby" mode; it

monitors the primary HUB's heartbeat and, if several heartbeats have been lost, assumes that the primary HUB has become unavailable.

7.3 Failover Handling

The HUB on hot standby is configured identically to the active HUB and will take over the sessions in case of failover.

The failover control module on the HUB on standby monitors the active HUB's heartbeat and claims the virtual IP address in case of heartbeat loss, notifying the other network nodes by sending out an ARP (a broadcast message that ties the IP address to the MAC address of its Ethernet interface).

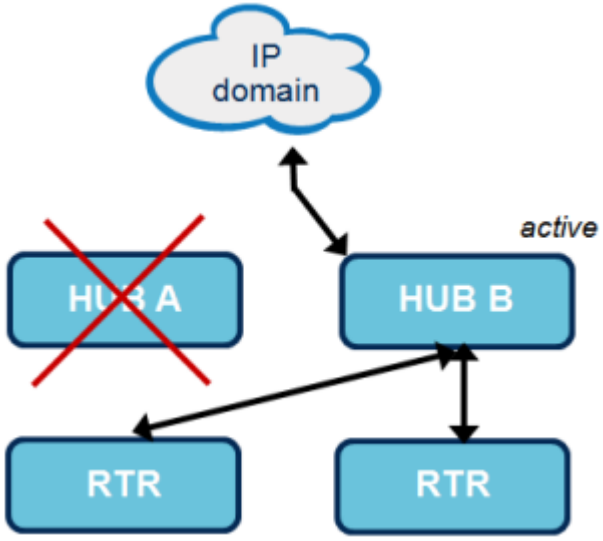


Figure 28: HUB failover

The primary HUB (A) is now inactive and the secondary HUB (B) has become active.

Because the primary HUB (A) is no longer available, applications in the IP Domain will need to set up new sessions with the secondary HUB (B).

If HUB A becomes available again, it will act as a hot standby for HUB B.

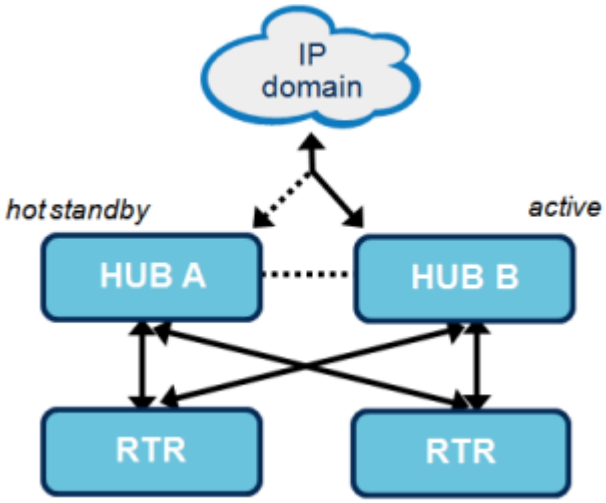


Figure 29: HUB hot standby

7.4 Configuration

Failover control is enabled via SNMP (see *OAM Interface (SNMP)*) and can be configured by setting the HUB parameters `hubipfailovercontrol` and `hubipfailovertimeout`in the `common_config.txt` configuration file located in the `/usr/TextPass/etc` directory. More information can be found in the *Configuration* chapter for these parameters.

The common IP address is configured via the `tp_ip_config.txt` file located in the `/usr/TextPass/etc` directory.

This is a sample `tp_ip_config.txt` file where the common IP address is 10.0.0.123, on a 10.0.0.0/16 network:

```
#interface     ip address      broadcast address       netmask
bge0:1         10.0.0.123      10.0.0.255              255.255.0.0
```

After the configuration of these parameters in the `common_config.txt` and of the common IP address in the `tp_ip_config.txt` file, the functionality can be enabled by (re)starting the hub using:

```
$tp_stop --tp_hub
$tp_start --tp_hub
```

Chapter 8

OAM Interface (SNMP)

Topics:

8.1 Introduction

The Simple Network Management Protocol (SNMP) is an industry standard for management and configuration of network components. uses SNMPv1 to configure and monitor interfaces, system status, and settings.

Note: Because NewNet Mobile Messaging components store its configuration in volatile memory, the default configuration as stored in the configuration file is always restored after booting the Mobile Messaging component.

8.2 MIB Files

All information that can be configured or viewed with SNMP is described in the Management Information Base (MIB) files.

Each MIB file is stored in a separate `*.my` file. Due to the size of the MIB files, they are not included in this manual. However, they can easily be viewed on the system. The MIB files are located in `/usr/local/share/snmp/mibs/`.

8.3 SNMP Manager

For configuration and monitoring purposes, an SMNP Manager or Management Station issues SNMPv1 requests to the HUB. SNMP Managers should send such a request to UDP port 11261 of the HUB. The HUB does not force an SNMP Manager to originate requests from any specific UDP port (any UDP port may be used for this purpose).

Note: If '`snmpPropListenAddressType`' parameter in semi-static configuration file is set to 'dual', then HUB will accept requests on both IPv4 and IPv6.

8.4 Trap Service

Up to eight SNMP Managers can subscribe to the HUB's trap service. When a trap condition occurs, the HUB sends an SNMP trap to any SNMP Management Station that is subscribed to the trap service.

To subscribe an SNMP Manager to the trap service, add an entry to the Alarm Station Table that contains the IP address (IPv4 or IPv6) or Hostname of the SNMP Manager and a UDP port number to which SNMP traps should be sent for that particular SNMP Manager. The Alarm Station Table is also SNMP manageable; refer to the `TEXTPASS-GEN` MIB for more information about this table.

The HUB always originates SNMP traps from UDP port 11262 and terminates them in the UDP ports that are specified in the Alarm Station Table. The community string that the HUB specifies in SNMP traps is always equal to public.

SNMP traps that the HUB generates are logged locally in `/var/adm/messages` (on Solaris systems) or `/var/log/messages` (on Linux systems) by a SNMP Trap daemon using UDP port 11173.

Note:

1. If '`snmpPropAlarmOwnIpv6Address`' parameter in semi-static configuration file is set, then specified address will be used as source address for sending SNMP traps to SNMP Manager with address of type IPv6.
2. If '`snmpPropAlarmOwnIpAddress`' parameter in semi-static configuration file is set, then specified address will be used as source address for sending SNMP traps to SNMP Manager with address of type IPv4.

Refer to the NewNet Mobile Messaging SNMP Trap Reference Guide, available on the documentation CD, for a description of the SNMP traps and their related MIB. For each trap described in the NewNet Mobile Messaging SNMP Trap Reference Guide, the trap severity level is indicated.

8.4.1 RTR Traps

The following RTR-related traps are defined:

- `tprUnAvailable`—[Critical] No RTRs are available; the HUB will now graceful shutdown its listeners and connections.
- `tprAvailable`—[Error] At least one RTR is available; the HUB will now initialise its listeners.

8.5 Corresponding Clear SNMP Traps

The HUB includes SNMP traps that have a corresponding clear trap. The HUB generates the traps when an alarm condition occurs and generates the appropriate clear trap when the alarm condition ceases. This functionality enables the SNMP Management Station to automatically clear an alarm condition.

The SNMP traps and their corresponding clear traps are:

Alarm Trap	Clear Trap
applicationIncorrectOutsideSmppPassword	applicationCorrectOutsideSmppPassword
applicationIncorrectOutsideUcpPassword	applicationCorrectOutsideUcpPassword
applicationIncorrectOutsideCimdPassword	applicationCorrectOutsideCimdPassword
applicationInsideSmppLoginRefused	applicationInsideSmppLoginSucceeded
applicationInsideUcpLoginRefused	applicationInsideUcpLoginSucceeded
applicationInsideCimdLoginRefused	applicationInsideCimdLoginSucceeded
applicationInsideSufficientCapacityNotAvailable	applicationInsideSufficientCapacityAvailable

The outside "password incorrect" traps are cleared when the application establishes a correct log-in, while previously, an incorrect password trap was sent. The inside "login refused" traps are cleared when a session for the application to the particular service center can be established, while previously, the log-in failed.

Note: Please refer to the "NMM SNMP Trap Reference Guide" for more details on applicationInsideSufficientCapacityNotAvailable and applicationInsideSufficientCapacityAvailable traps.

8.6 Trap Filtering

SNMP trap filters to be applied on the NewNet Mobile Messaging component(s) and can be customised per configured alarm station. This filtering can be configured using a combination of black-listing and white-listing of traps in two tables that are associated with the alarm stations table:

- **Whitelist table**—Contains a list of all traps that should be sent toward an alarm station (wild-cards are allowed)
- **Blacklist table**—Contains a list of traps that should be blocked for a particular alarm station (wild-cards are allowed)

Note: The whitelist is applied before the blacklist. An empty whitelist is identical to a whitelist of "*". An empty blacklist does not block any trap.

The following rules and restrictions apply:

- Creating a whitelist for a trap belonging to a specific MIB, implicitly blacklists all other traps from that MIB, for the alarm station the whitelist is configured on.
- It is *not* possible to have a black- and whitelist for traps that belong to the same MIB and for the same alarm station. It is possible to combine both a black- and whitelist for a specific trapreceiver, as long as the black- and whitelist do not contain entries from the same MIB.

8.7 Device Type Variable Binding

The generic SNMP library that the HUB uses can, when a SNMP trap is invoked, automatically add an extra variable binding to the trap. This variable contains the product name as specified in the global variable module. The product name is the last variable in the trap message. This feature allows trap receivers to distinguish among traps from different products.

Example:

```
11:28:58 TEXTPASS-GEN-MIB::networkDiscoveryNodeAdded TEXTPASS-GEN-
MIB::lastSnmpErrorString.0 = STRING: "node_type=RTR, port=25092,
ip1=10.0.0.46"       TEXTPASS-GEN-MIB::deviceType.0 = STRING: "HUB"
from system1.asd.mbalance.com
```

Chapter 9

Application Interface

Topics:

9.1 Introduction

The application and the HUB can communicate using any of the following protocols:

- UCP over TCP/IP
- SMPP over TCP/IP
- CIMD2 over TCP/IP

The access protocol for the application interface is always TCP/IP. HUB nodes operate independently; therefore, each SMS application is responsible for establishing and managing connections to all HUBs. IP fail-over can be implemented by using load balancers.

Note: The HUB does not establish a connection to the SMS application even when an AT message should be delivered to the application. If the HUB receives an AT message for an application that is not connected, a fall-through rule action is executed (if defined).

An application can establish multiple simultaneous sessions toward a HUB. The HUB load balances AO sessions toward RTRs or SMSCs over all configured RTRs and SMSCs when connections from an application are established.

Note: When a previously activated application is deactivated, any messages (notifications or otherwise) for that application will be rejected.

9.2 UCP Interface

When using the UCP interface, the application should connect to the HUB as a UCP client to a UCP server, as if it were an SMS application connecting to an SMSC.

The operations of the command and response strings for the available HUB function modules are described in corresponding chapters on the modules.

Refer to the NewNet Mobile Messaging UCP Protocol Implementation Compliance Statement document, available on the documentation CD, for the conformance statements of the HUB UCP protocol implementation with the UCP version 5.1 standard.

Note: This manual assumes that the reader is familiar with UCP application programming. For more information about UCP and application development using UCP, refer to the EMI-UCP protocol specification.

9.2.1 UCP Sessions

The standard procedure for interacting with the HUB using UCP is as follows:

1. Set up a TCP/IP connection between the remote SMS application and the HUB. The remote SMS application is responsible for initiating the connection and should manage this connection.
2. Open a UCP session using the UCP 60 open session operation.
3. Now a session is established and the HUB can exchange UCP messages with the remote SMS application.
4. When the session is over and no more interaction is required with the HUB, all UCP sessions (and all TCP/IP connections) should be closed by the remote SMS application.

Steps **1** and **2** should be repeated to set up multiple UCP sessions between the SMS application and a HUB.

Note: A maximum of 255 simultaneous UCP sessions can be established between an SMS application and a HUB.

The HUB can disconnect UCP sessions if one of the following occurs:

1. The UCP session does not carry UCP traffic for a period longer than the specified max *inactivity time*.
2. The HUB counts more time-outs than configured for this interface.
3. The HUB receives an invalid UCP message on the session (such as UCP messages longer than 1024 bytes).
4. A session disconnect (inside or outside) occurs.

9.2.2 UCP Operations

The HUB supports the following UCP operations:

Operation	Support

UCP 01 CALL INPUT	• The HUB decodes the content of the UCP 01 operation and handle this as a submit request in case the UCP 01 request is initiated by an SMS application (AO traffic).
	• The HUB handles the UCP 01 request as unknown message in case the UCP 01 request is initiated from an SMSC (AT traffic).
	• Modifiers for UCP 01 are supported for AO-AO traffic.
	• The HUB supports UCP 01 requests initiated from the RTR in case of AO-Store-AO.
	• For legacy UCP application support, the UCP 01 request will be initiated by the HUB if:
	  • Legacy UCP application support is enabled
	  • The previous UCP 52/53 request is negatively acknowledged by the application with error code 03
	  • Conversion of UCP 52/53 to UCP 01 is possible.
	
	The HUB can handle UCP 01 requests with AMsg fields up to 640 characters. Refer to *UCP 01 Support* for more details.

UCP 02 MULTIPLE ADDRESS CALL INPUT	The multiple address call input request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The syntax screening of the message is successful • The configured rules do not block this specific AO message The HUB can handle UCP 02 requests with AMsg fields up to 640 characters. Refer to *UCP 02 Support* for more details.

Operation	Support
UCP 30 SMS MESSAGE TRANSFER	The SMS message transfer request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The syntax screening of the message is successful • The SRI-SM (if configured to be performed in the RTR based on recipient number on or before AO rule evaluation) is successful • The configured rules do not block this specific AO message
UCP 31 SMT ALERT	The SMT alert request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The syntax screening of the message is successful • The configured rules do not block this specific AO message
UCP 51 SUBMIT SHORT MESSAGE	The submit short message request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The SendRoutingInfoForSm (if configured to be performed in the RTR based on recipient number on or before AO rule evaluation) is successful • The configured rules do not block this specific AO message The HUB can handle UCP 51 requests with AMsg fields up to 640 characters. The HUB ignores the UCP 51 parameters associated with the following functionality: • Last resort • Priority • Reply path
UCP 52 DELIVER SHORT MESSAGE	The deliver short message request will be initiated by the HUB if: • A routing rule is specified to route messages toward this application • The application is connected and the transmission window is not yet full • The AT throughput is not exceeded for this application The HUB passes UCP 52 requests for messages that exceed 160 characters to the RTR as unknown requests. The following optional UCP 52 parameters are not filled in: • More messages to send (MMS) • Reply path indicator (RPI) • Message class (MCLs) • HPLMN and XSER, except for data coding scheme (DCS) and user data header (UDH)

Operation	Support
UCP 53 DELIVER NOTIFICATION	The deliver notification short message will be initiated by the HUB if: • A corresponding UCP 51 was received with notification requested and the notifications are allowed for this application • The session in which the corresponding UCP51 was received in still open • The AT throughput is not exceeded for this application The HUB rejects UCP 53 requests for messages that exceed 160 characters. If the delivery status of the UCP 53 is 1 (buffered) or 2 (not delivered), the reason code indicates the type of error that occurred.
UCP 54 MODIFY MESSAGE (requires AMS)	The modify message request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The address code of the recipient (ADC) and the SCTS, optionally with the originator address and authentication code, deliver a match in the store messages • The configured rules do not block this specific AO message • Other incoming UCP messages are forwarded to the SMSC as "unknown message” • All AT UCP messages are relayed to the application
UCP 55 INQUIRY MESSAGE	Refer to *UCP Inquiry and Delete* for details.
UCP 56 DELETE MESSAGE	Refer to *UCP Inquiry and Delete* for details. The HUB supports UCP 56 requests with AMsg fields up to 640 characters.
UCP 57 RESPONSE INQUIRY MESSAGE	Refer to *UCP Inquiry and Delete* for details.
UCP 58 RESPONSE DELETE MESSAGE	Refer to *UCP Inquiry and Delete* for details.
UCP 60 SESSION MANAGEMENT	Only the subtype of operation 1 (open session) is supported. The open session request will be positively acknowledged if: • The short number and password combination matches • The short number is configured as an active UCP application • The maximum number of sessions for this application is not exceeded

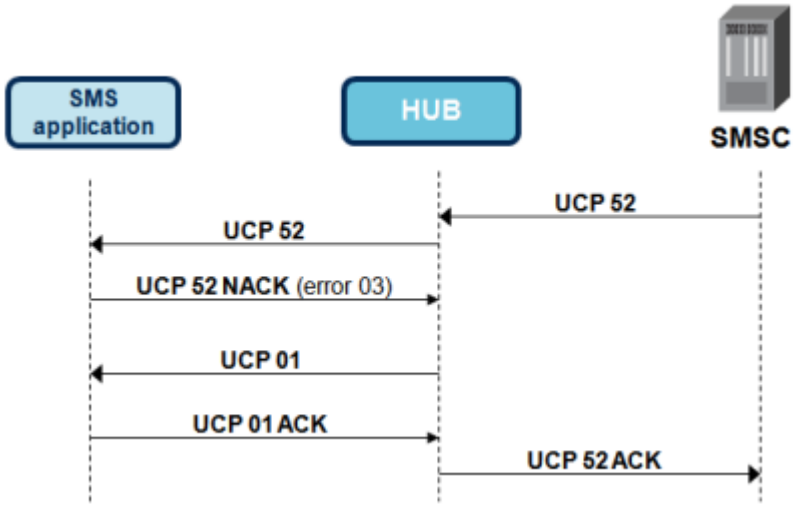
Note: UCP operations that are related to specific routings paths may not be allowed on your HUB due to license restrictions.

9.2.3 Legacy UCP Support

9.2.3.1 Legacy UCP Application Support

The HUB supports legacy UCP applications for outside and outside dial out sessions. This allows support for legacy UCP application which are unable to handle UCP 52 or UCP 53 requests The UCP 01 operation is used for legacy applications that do not support the UCP 52/53 requests.

An example of a successful legacy UCP application support scenario:



The HUB converts UCP 52/53 request messages into a legacy UCP 01 request when the UCP 52/53 request is negative acknowledged (NACK) with error code '03' (operation not supported).

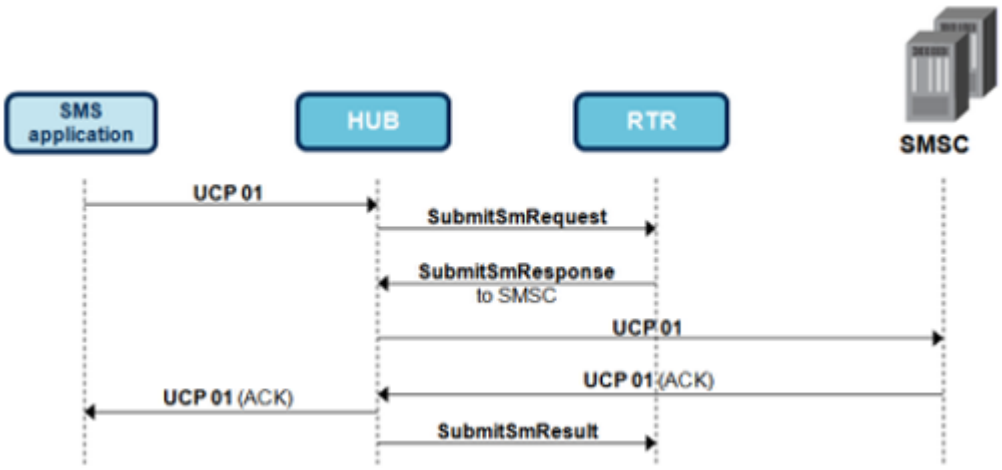
The HUB will only send a UCP 01 request toward a legacy application when it is able to convert the requests. If the HUB is not able to convert the request, the request will be negative acknowledged (NACK) towards the SMSC. UCP 52/53 requests containing binary user data or alphanumeric *OAdC* fields are *not*converted to UCP 01 requests. Unsupported message fields are silently ignored The legacy UCP application support can be enabled via the semi-static parameter `hublegacyucpapplicationsupport`.

Refer to the *Configuration* chapter for more information about this parameter and configuration details.

9.2.3.2 UCP 01 Support

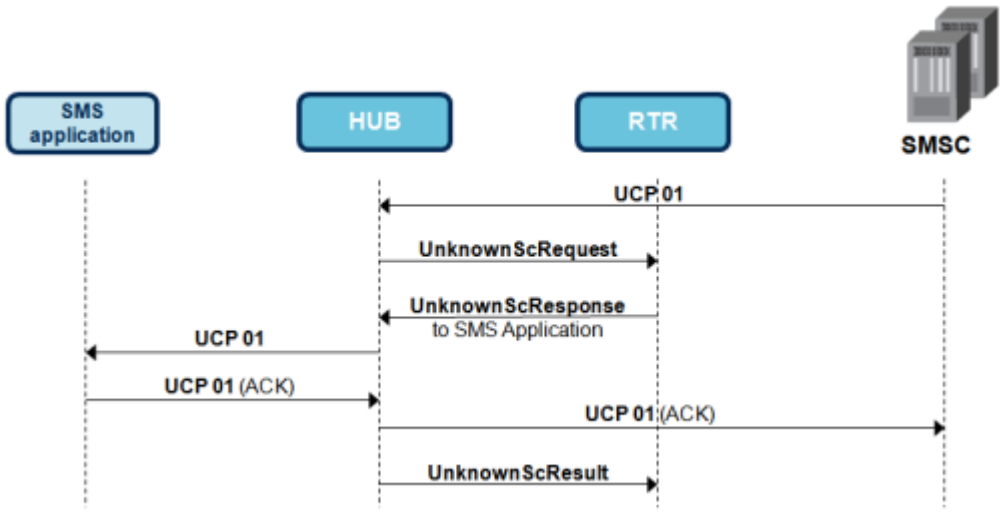
The HUB is able to decode the content of the UCP 01 message and handle this as a submit request in case the UCP 01 request is initiated by an application.

This flow shows a UCP 01 initiated from an application (AO traffic):

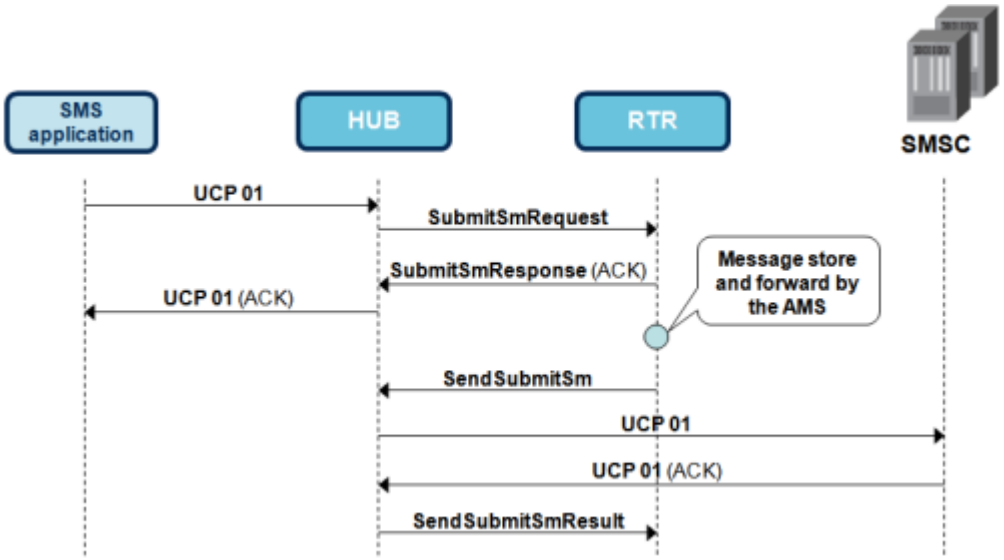


When the UCP 01 request is initiated from an SMSC the HUB handles the UCP 01 request as unknown message.

This flow shows a UCP 01 initiated from a SMSC (AT traffic):



This flow shows a UCP 01 handled by the RTR in case of an AO-Store-AO routing path (with AMS):



The following special cases and/or exceptions exist:

- The OAdC field within the UCP 01 is optional, therefore the UCP 01 does not always contain an originator. The HUB, in this case, uses the application short number as originator toward the RTR only.
- The HUB will not send a UCP 01 operation to an application when these are triggered by the RTR by a *sendDeliverSmRequest*.
- The RTR is able to modify the content of messages with modifier fields. The HUB will block the forwarding of a modified submit request in two cases:
  - AMsg field with a binary date
  - Alphanumeric OAdC field.

9.2.3.3 UCP 02 Support

The HUB supports UCP 02 messages. To reduce to impact on the RTR the handling of UCP 02 requests depends on a global configuration setting in the HUB (`hubucphandlingucp02`). This setting specifies how UCP 02 request should be handled. `hubucphandlingucp02` can have three different values:

- *Not Supported*, the HUB will NACK the UCP 02 request with an internal error `operationNotSupported` which is default set to UCP error code 03 and configurable via the error mapping functionality.
- *Forward Unknown*, the HUB will forward the UCP 02 request as an unknown message to the SMSC. The message is sent as a UCP 02.
- *Split Up*, the HUB will split up the UCP 02 request in multiple submit requests for every recipient address. The UCP 02 messages are split up and sent as UCP 51 messages.

In case of split up of UCP 02, the submits are sent to the RTR. The RTR decides if the submit must be handled by the RTR or that the request must be forwarded to SMSC. The HUB acknowledges the pending UCP 02 requests when all submits are handled.

This flow shows a split up where the messages are divided into multiple submits which are forwarded to the SMSC:

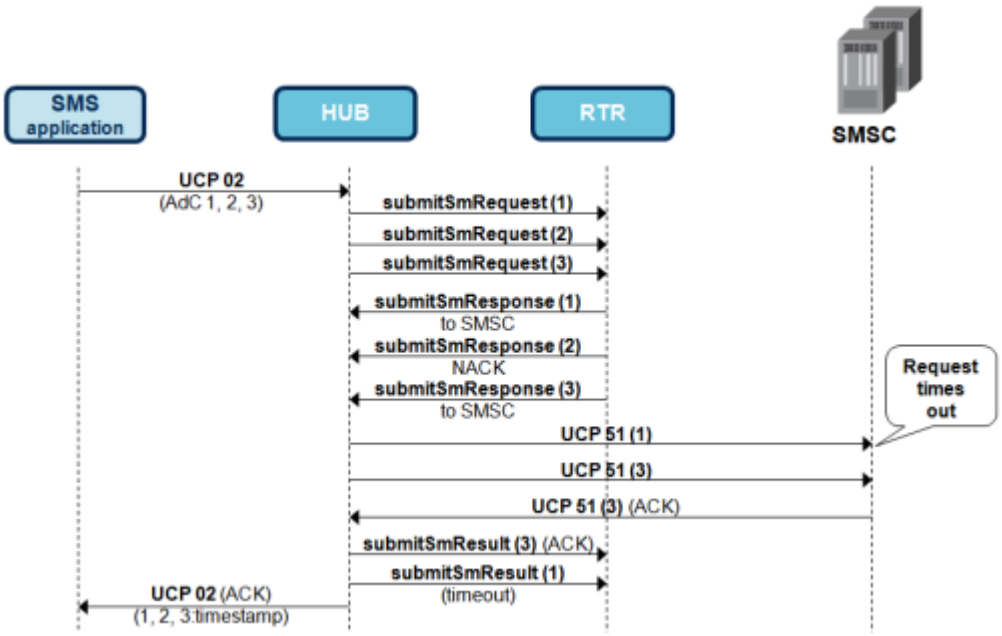


Figure 30: Messages divided into multiple submits

A timeout of a split up submit is handled as a NACK. The UCP 02 will be Nacked if all split up submits are Nacked. Successful recipients will have an SMSC timestamp.

This flow shows a split up where the messages are divided into multiple submits which are stored in the AMS:

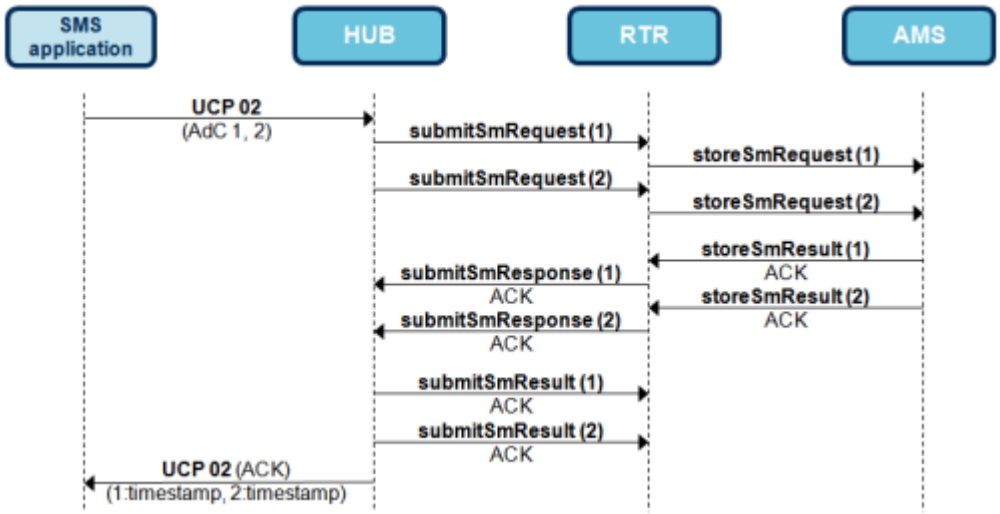


Figure 31: Messages divided into multiple submits, stored in AMS

The following special cases and exceptions exist:

- When one of the divided submit request times out, the HUB will handle this as a NACK.
- The OAdC field within the UCP 02 is optional, therefore the UCP 02 does not always contain an originator. The HUB, in this case, uses the application short number as originator.

- The HUB does not support a *sendSubmitSmRequest* for UCP 02. Meaning that when the RTR initiates a *sendSubmitSmRequest* with operation code 02 the HUB will ignore operation code 02 and send a UCP 51 request to the SMSC.

9.2.3.3.1 Configuration Dependencies

A UCP 02 can handle a maximum of 20 recipient addresses. This means that a UCP 02 can lead to a maximum of 20 internal submit messages to the RTR/SMSC when using split up. Therefore the following configuration settings need special attention:

- Throughput control—For throughput control the RTR must be able to handle at least 20 submit messages per second. Therefore the following throughput control settings need to be set correctly via the MGR:
  - **Throughput AO Maximum** from the application. Accessible via **SMS Applications ➤ Applications**
  - **ThroughputAo** from the associated applications in the group. Accessible via **SMS Applications ➤ Groups**
  - **AO Throughput** from the associated applications in the service class. Accessible via **Environment ➤ Service Class**
- Window sizes—The following application window sizes need to be set correctly via the MGR (accessible via **SMS Applications ➤ Applications**):
  - **Inside UCP Window Size**
  - **Outside UCP Window Size**
  - **Max Outside UCP Sessions**

  For the relation between window sizes when using UCP 02 the following rule can be used:

  Inside UCP Window Size = (Outside UCP Window Size + (Max Outside UCP Sessions * 20)

  Note: The maximum configurable **Inside UCP Window Size** is 100. If the rule calculates a higher value, the maximum of 100 should be used.

9.2.4 UCP Inquiry and Delete

The HUB provides support for the UCP 55/57 (inquiry) and UCP 56/58 (delete) operations.

- If the HUB is part of an SMS Store configuration that includes an AMS, the HUB will forward the message via the RTR to the AMS to implement these operations.
- If the HUB is part of an Application Gateway (AGW) configuration that does not include an AMS, the HUB will forward the UCP operations to one or more SMSCs. The answers of the back-end SMSCs are combined in a single answer towards the application.

Note: The HUB only sends one UCP 55 or UCP 56 response. UCP 57 and 58 operations from each SMSC are sent independently to the application.

9.2.4.1 Successful Message Inquiry with AMS

A successful message inquiry scenario with an AMS:

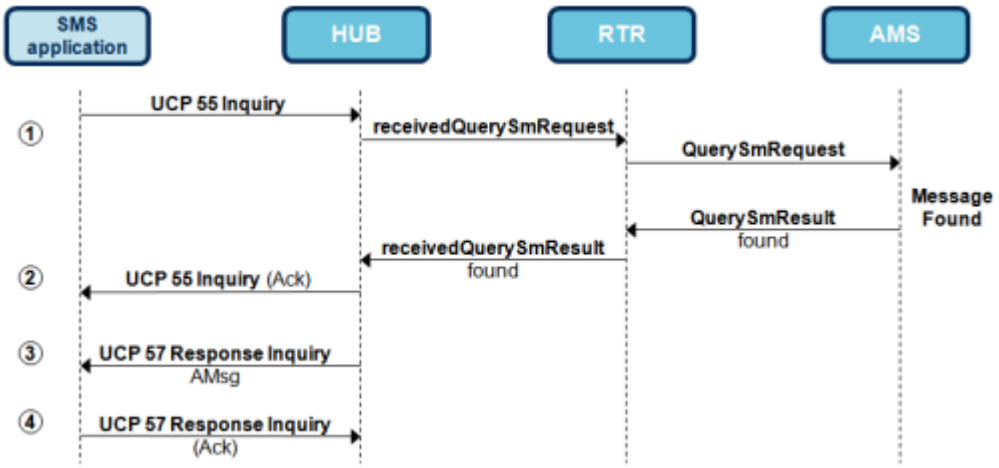


Figure 32: Successful message inquiry with AMS

1. On reception of a correct UCP 55, the HUB sends a *receivedQuerySmRequest* toward the RTR. The RTR then forward the request to the AMS.
2. The HUB acknowledges the pending UCP 55 when receiving an *receivedQuerySmResult* from the RTR.

   Note: In case the receivedQuerySmResult returns with a messageNotFound or deliveryInProgress, the HUB will acknowledge the UCP 55 and send a UCP 57 Response Inquiry with no timestamps in the AMsg field.

3. The HUB sends a UCP 57 Response Inquiry toward the application. The *AdC* field in the UCP 57 message contains the short number of the application.

   Note: The AdC in the UCP 57 does not have to be the application short number. If in the original UCP 55 the OAdC is an MSISDN, then the AdC in the UCP 57 would also be an MSISDN (OAdC of the UCP 55 is switched to AdC in the UCP 57). If the OAdC in the UCP 55 is a short number, then the AdC in the UCP 57 is also a short number.

   The *AMsg* field is formatted as `[TEXT1] <AdC> [TEXT2] {TIMESTAMP}`, where `{TIMESTAMP}` is a list of matching Service Centre timestamps, separated by spaces and `<AdC>` is the recipient address of the buffered short message(s). The maximum *AMsg* size can be 640 characters. If it does exceed 640 characters, only complete timestamps will be included, remaining timestamps will not be included, and an event will be logged to `syslog`. No multiple UCP 57 Response Inquiry messages will be sent in this case.

   The HUB has a global configuration setting (`hubucplonginquirydeleteresponse`) to specify if long messages are allowed for UCP 57 and UCP 58, because not all applications can handle *AMsg* of 640 characters. The maximum *AMsg* size is 640 characters when `hubucplonginquirydeleteresponse` is set to "true".

   Refer to [3] for more details on UCP message fields and refer to the *Configuration* chapter for configuration details of the `hubucpinquirytext1` (`[TEXT1]`), `hubucpinquirytext2` (`[TEXT2]`), and `hubucplonginquirydeleteresponse` fields.

4. The application responds with an UCP 57 acknowledgement.

9.2.4.2 Successful Message Deletion with AMS

A successful message deletion scenario with an AMS:

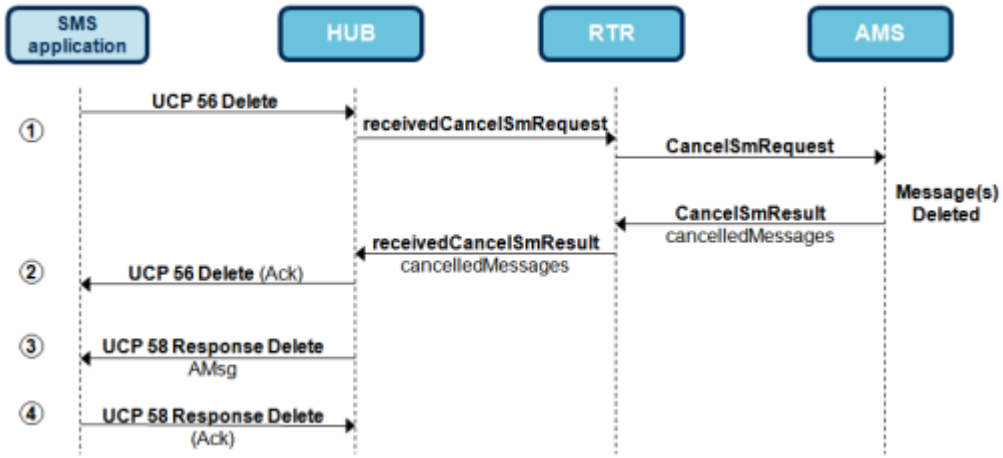


Figure 33: Successful message deletion with AMS

1. On reception of a correct UCP 56, the HUB sends a *receivedCancelSmRequest* toward the RTR. The RTR then forwards the request to the AMS. The HUB can handle up to 50 service centre timestamps in the AMsg field of a received UCP56.
2. The HUB acknowledges the pending UCP 56 when receiving an *receivedCancelSmResult* from the RTR.

   Note: In case the receivedCancelSmResult returns with a messageNotFound or deliveryInProgress, the HUB will acknowledge the UCP 56 and send a UCP 58 Response Delete with no timestamps in the AMsg field.

3. The HUB sends a UCP 58 Response Delete toward the application. The *AdC* field in the UCP 58 message contains the short number of the application.

   The *AMsg* field is formatted as `[TEXT3] <AdC> [TEXT4] {TIMESTAMP} [TEXT5]`, where `{TIMESTAMP}` is a list of matching Service Centre timestamps and `<AdC>` is the recipient address of the deleted short message(s). The maximum *AMsg* size can be 640 characters. If it does exceed 640 characters, only complete timestamps will be included, remaining timestamps will not be included, and an event will be logged to `syslog`.

   The HUB has a global configuration setting (`hubucplonginquirydeleteresponse`) to specify if long messages are allowed for UCP 57 and UCP 58 because not all applications can handle *AMsg* of 640 characters. The maximum *AMsg* size is 640 characters when `hubucplonginquirydeleteresponse` is set to "true".

   Refer to [3] for more details on UCP message fields and refer to the *Configuration* chapter for configuration details of the `hubucpdeletetext3` (`[TEXT3]`), `hubucpdeletetext4` (`[TEXT4]`), `hubucpdeletetext5` (`[TEXT5]`), and `hubucplonginquirydeleteresponse` fields.
4. The application responds with an UCP 58 acknowledgement.

9.2.4.3 Successful Message Inquiry without AMS

A successful message inquiry scenario without an AMS:

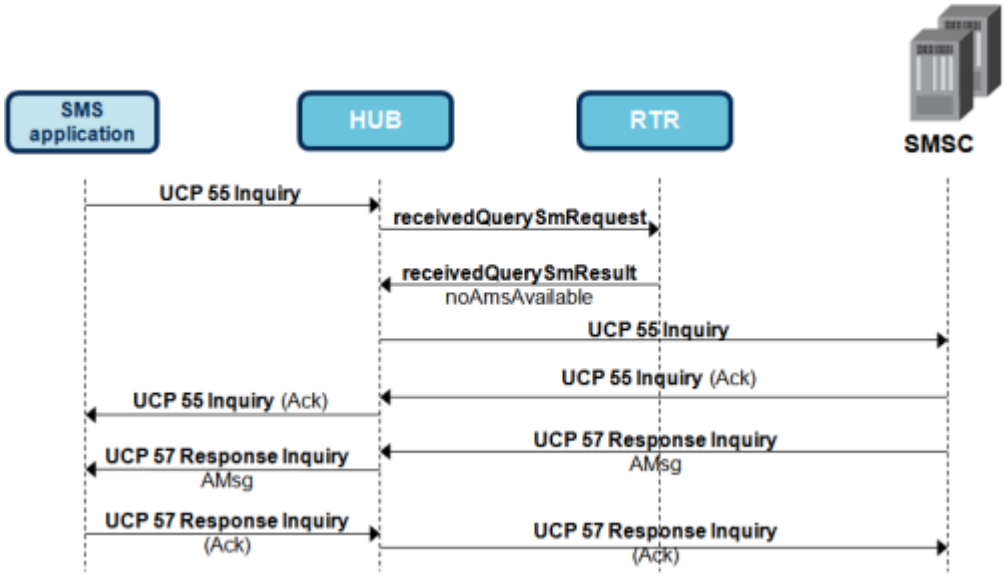


Figure 34: Successful message inquiry without AMS

In case *receivedQuerySmResult* returns with a *noAmsAvailable* or times out, the UCP 55 will be forwarded towards the SMSC(s).

9.2.4.4 Successful Message Deletion without AMS

A successful message deletion scenario without an AMS:

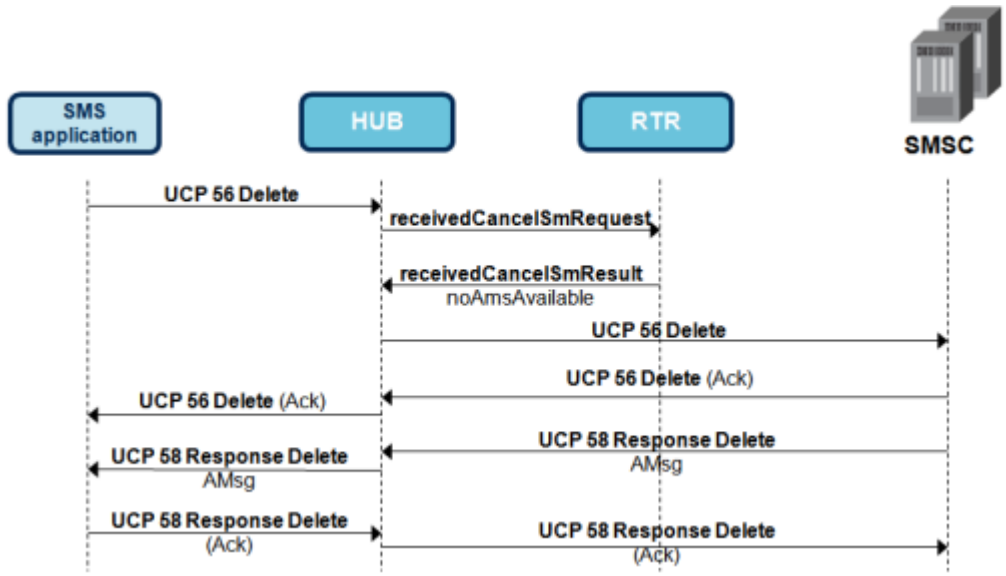


Figure 35: Successful message deletion without AMS

In case *receivedCancelSmResult* returns with a *noAmsAvailable* or times out, the UCP 56 will be forwarded towards the SMSC(s).

9.2.5 Customizing UCP Notification Operations

The HUB enables you to customize UCP 53 notification operations by modifying the text in the AMsg field. You can use this functionality to translate delivery notifications to languages other than English.

Note: Some external applications (ESMEs) use the notification text for processing UCP 53 operations (instead of the reason code). Therefore, it is recommended that you do not change the notification text unless it is required for compatibility reasons. If you do intend to use this functionality, it is important to communicate that to external applications so they can be modified, if necessary.

The HUB supports:

1. Customization of the notification text (AMsg field) for each delivery status, optionally using substitution parameters to insert items such as error texts and error codes
2. Configuration of a default error string, error code, error text, and UCP reason code (Rsn field) for each delivery status
3. Changing the default mapping of the HUB's internal, normalized errors to error codes, error texts, and UCP reason codes (Rsn fields)

This functionality is configured in the semi-static configuration file. For information about the configuration entities, refer to *hubnotificationtext Entity* and *hubnotificationerrormapping Entity*. For a list of reason codes that can be used in the configuration, refer to *UCP 53 Notification Reason Codes*. For information about the normalized errors that can be used in the configuration, refer to *Error Mapping and Normalization*.

Sample Customized UCP 53 Operation

The following is a sample customized UCP 53 operation:

```
stx12/00376/O/53/0211123456/0172223322/////////////
051198101841/1/107/051198102200/3//4E6163687269636874206675 65
722030313732363831333733352C204964656E746966697A696572756E672
039383131303531303138343 12C2069737420676573706569636865727420
776F7264656E2C20646120456D706661656E67657220766F7275656265726
76568656E64206E6963687420657272656963686261722028436F64652031
3830292E/////////////9Detx
```

In this example:

- The Rsn field contains *107*
- The SCTS field contains *051198101841*
- The AMsg field contains *Nachricht fuer 0172223322, Identifizierung 981105101841, ist gespeichert worden, da Empfaenger voruebergehend nicht erreichbar (Code 180).*

Sample Configuration for Successful Delivery

In this sample hubnotificationtext entity:

- notificationstatus indicates that this entry configures the properties of the deliverySuccessful delivery status
- formatstring defines the notification text to use, including substitution parameters (as described in *formatstring*)

- `usageofdefaults` indicates that the default configuration defined by this entity should always be used, and `hubnotificationerrormapping` should not be used
- `defaulterrorcode` defines the default error code to use for deliverySuccessful
- `defaulterrortext` defines the error text to use for deliverySuccessful
- `defaultucpreasoncode` defines the UCP reason code to use in the `Rsn` field for deliverySuccessful

```
<hubnotificationtext notificationstatus="deliverySuccessful"
formatstring="Message for %MRAD, with identification
%SDD%SDM%SDY%STH%STM%STS has been delivered on %DDM/%DDD/%DDX at
%DTI:%DTM%DTP." usageofdefaults="always" defaulterrorcode="19"
defaulterrortext="no error" defaultucpreasoncode="12" />
```

Sample Configuration for Failed Delivery

In this sample `hubnotificationtext` entity:

- `notificationstatus` indicates that this entry configures the properties of the deliveryFailed delivery status
- `formatstring` defines the notification text to use, including substitution parameters (as described in *formatstring*)
- `usageofdefaults` indicates that the default configuration defined by this entity should only be used if there is not an error (if there is an error, the appropriate `hubnotificationerrormapping` entity will be used)
- `defaulterrorcode` defines the error code to use for deliveryFailed
- `defaulterrortext` defines the error text to use for deliveryFailed
- `defaultucpreasoncode` defines the UCP reason code to use in the `Rsn` field for deliveryFailed

```
<hubnotificationtext notificationstatus="deliveryFailed"
formatstring="Message for %MRAD, with identification
%SDD%SDM%SDY%STH%STM%STS, failed (%MERS)."
usageofdefaults="usageIncaseOfNoError" defaulterrorcode="890"
defaulterrortext="Delivery failed" defaultucpreasoncode="108" />
```

Sample Configuration for Expired Validity Period

In this sample `hubnotificationtext` entity:

- `notificationstatus` indicates that this entry configures the properties of the validityExpired delivery status
- `formatstring` defines the notification text to use, including substitution parameters (as described in *formatstring*)
- `usageofdefaults` indicates that the default configuration defined by this entity should always be used, and `hubnotificationerrormapping` should not be used
- `defaulterrorcode` defines the default error code to use for validityExpired
- `defaulterrortext` defines the error text to use for validityExpired

- `defaultucpreasoncode` defines the UCP reason code to use in the `Rsn` field for validityExpired

```
<hubnotificationtext notificationstatus="validityExpired"
formatstring="Message for %MRAD, identification
%SDD%SDM%SDY%STH%STM%STS is expired (Code %MERR)."
usageofdefaults="always" defaulterrorcode="50"
defaulterrortext="Expired" defaultucpreasoncode="108" />
```

Sample Internal Error Mapping

In this sample `hubnotificationerrormapping` entity:

- `id` defines the SNMP ID of the entity
- `internalerror` indicates that this entity configures the properties that should be used if the HUB's internal, normalized error is sriSmTeleserviceNotProvisioned
- `errorcode` defines the error code to use; you can display this string in the `hubnotificationtext formatstring` by using the `%MERR` substitution parameter
- `errortext` defines the error text to use; you can display this string in the `hubnotificationtext formatstring` by using the `%MERS` substitution parameter
- `ucpreasoncode` defines the UCP reason code to use

```
<hubnotificationerrormapping id="30"
internalerror="sriSmTeleserviceNotProvisioned" errorcode="890"
errortext="Telecom service not provisioned" ucpreasoncode="108" />
```

Note: This functionality is not available for Application Gateway (AGW) routing paths such as AO-AO and AT-AT.

9.3 SMPP Interface

This section describes the HUB's SMPP interface.

Refer to the NewNet Mobile Messaging SMPP Protocol Implementation Compliance Statement document, available on the documentation CD, for the conformance statements of the HUB SMPP protocol implementation with the SMPP version 3.4 and version 5.0 protocols.

Note: SMPP Applications that do not send bind requests, but are either authenticated by their IP address or do not use authentication at all (i.e. anonymous access), are assumed to use SMPP version 3.3.

Note: This manual assumes that the reader is familiar with SMPP application programming. For more information about SMPP specifications and application development using SMPP, refer to *http://www.smsforum.net*.

9.3.1 SMPP Interface Configuration

The standard procedure for interacting with the HUB via the application interface using SMPP is as follows:

1. Set up a TCP/IP connection between the remote SMS application and the HUB.

   The remote SMS application is responsible for initiating the connection and should manage this connection.

2. Bind the connection using one of the following link types:
   a) Receiver link using the BIND_RECEIVER command
   b) Transmitter link using the BIND_TRANSMITTER command
   c) Transceiver link using the BIND_TRANSCEIVER command

3. Now the link is bound.

   On the SMPP link, the HUB can route SM messages to the remote SMS application using the DELIVER_SM operation for sending AT messages and notifications to an application and receive the submit_sm operation for sending AO messages from an application.

4. When the session is over and no more interaction is required with the HUB, all SMPP links should be closed with the UNBIND command and all TCP/IP connections should be closed.

9.3.2 SMPP Operations

The HUB supports the following SMPP operations:

Operation	Support
BIND	All three types of BIND are supported: receiver, transmitter, and transceiver. The BIND request will be positively acknowledged if: • The `system_id`, `system_type` and `password` combination provided in a BIND matches an active application. • The maximum number of sessions of a particular type (receiver, transmitter, transceiver) for this application is not exceeded.
OUTBIND	The HUB uses the OUTBIND request to setup dial-out connections. The HUB sends these requests to dial-out applications.
UNBIND	The UNBIND operation is to de-register an instance of an SMPP application from the HUB and inform the HUB that the SMPP application no longer wishes to use this session for the submission or delivery of messages.
GENERIC_NACK PDU	The generic negative acknowledgement to an SMPP PDU submitted with an invalid message header. A GENERIC_NACK response is returned in the following cases: • Invalid `command_length` • Unknown `command_id`

Operation	Support
SUBMIT_SM	The SUBMIT_SM request will be positively acknowledged if: • The application is connected as transmitter or transceiver • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The screening of the message is successful (user data does not exceed 140 bytes or 160 7-bit characters) • The configured rules do not block this specific AO message
SUBMIT_MULTI	Only supported in Application Gateway (AGW) configuration.
DELIVER_SM	The HUB will initiate the DELIVER_SM request if: • A routing rule is specified to route a message toward this application • The application is connected as receiver or transceiver and the transmission window is not yet full • The AT throughput is not exceeded for this application Depending on the `Stat` field of the DELIVER_SM delivery receipt, the error code indicates the type of error that occurred.

DATA_SM	The HUB treats DATA_SM requests from an application as SUBMIT_SM requests.
	• The maximum payload is 640 characters.
	• Segmentation of long payloads takes place in the RTR.
	• The HUB supports the following submission TLVs (tag, length, value):
	  • `dest_port`
	  • `source_port`
	  • `payload_type`
	  • `message_payload` (640 characters or less)
	  • `sar_msg_ref_num/sgement_seqnum/total_segments`
	  • `more_messages_to_send`
	  • `qos_time_to_live`
	  • `originator_address_for_billing_overrride`
	• Delivery notifications toward an application will be sent as DATA_SM if the AO message was submitted as DATA_SM.
	Note: If the payload_type is 1, the RTR will send the payload in UDH IEI 9. WCMP messages can only be sent if the payload fits in a single MT message.

QUERY_SM	This command is issued by the SMPP application to query the status of a previously submitted short message. The matching mechanism is based on the assigned `message_id` and source address.
CANCEL_SM	This command is issued by the SMPP application to cancel a previously submitted short message that is still pending delivery. If the `message_id` is set to the ID of a previously submitted message, then provided the source

Operation	Support
	address supplied by the SMPP application matches that of the stored message, that message will be cancelled. **Note:** A message_id of NULL, used for canceling a group of messages, is not supported.
REPLACE_SM	The REPLACE_SM request will be positively acknowledged if: • The application is connected as transmitter or transceiver • The AO functionality is enabled for this application • The AO modification allowed configuration on the service class is set to true • The configured rules do not block this specific AO message
ENQUIRE_LINK	This operation can be sent by either the SMPP application or HUB and is used to provide a confidence check of the communication path between an SMPP application and the HUB. **Note:** SMPP operations that are related to specific routings paths may not be allowed on your HUB due to license restrictions.

9.3.3 SMPP Version 5.0 Support

The HUB supports SMPP 5.0 specific behavior for only those ESME applications which bind with the interface version set as "0x50" (indicating SMPP version 5.0).

For outside sessions, the HUB accepts incoming BIND requests (of transmitter, receiver and transceiver type) with the SMPP 5.0 interface version. It can also initiate BIND requests (of all three types mentioned above) with the SMPP 5.0 interface version for inside sessions, depending on the **Inside SMPP Version** and **Inside SMPP Bind Type** configuration settings on the MGR (refer to MGR Operator Manual for more details).

Note: If HUB receives an incoming BIND request from an ESME with the interface version set to any value between 0 (zero) and '0x33' (both inclusive), then it is considered as a SMPP 3.3 interface. Also, if HUB receives an incoming BIND request with the interface version set to '0x34' or any higher value other than '0x50'; then it is considered as a SMPP 3.4 interface.

The new SMPP 5.0 error codes ESME_RSERTYPDENIED, ESME_RSERTYPUNAUTH, ESME_RSERTYPUNAVAIL and ESME_RPROHIBITED are supported by the HUB for error translation, in case the outgoing SMPP interface is 3.4/3.3 (see section *Relaying SMPP 5.0 error codes to SMPP 3.3/3.4 interface*).

On the other hand, if the outgoing SMPP interface is 5.0 then no such error translation is required, as the HUB/AGW can send any of these newly added error codes to the ESME in that case. In particular, if the default error mapping configurations are used for SMPP, then the HUB can transparently relay the above error codes from an incoming SMPP 5.0 interface to an outgoing SMPP 5.0 interface (for AO-AO and AT-AT traffic).

The HUB also transparently relays the following optional SMPP 5.0 TLVs from an outside application (ESME) interface to an inside application (IP SMSC) interface and vice-versa, provided both the interfaces support SMPP 5.0. However, HUB does not parse/process any of these TLVs.

- *congestion_state*
- *dest_addr_np_country*
- *dest_addr_np_information*
- *dest_addr_np_resolution*
- *Endpoint identification TLVs (source_network_id, source_node_id, dest_network_id, dest_node_id)*
- *ussd_service_op*

Note that the HUB explicitly checks the outgoing SMPP interface version while transparently forwarding any of the above TLVs. If the outgoing interface version is other than 5.0, then the above TLVs are omitted from the outbound SMPP PDU. There is one exception in the case of *ussd_service_op*, which is also allowed for SMPP 3.4 interfaces but only in the SUBMIT_SM PDU.

9.3.4 SMPP Delivery Receipts Configuration

This section describes the configuration of fields in SMPP Delivery Receipts:

1. Message_state: This field is used by SMSC in the `deliver_sm` and `data_sm` PDUs to indicate to the ESME the final message state for SMSC Delivery receipts.

   The value of this field can be configured through `hubnotifysmppmsgstatedeliveryfailed` and `hubnotifysmppmsgstatecancelled` semi-static parameters in the following scenarios:

   - `hubnotifysmppmsgstatedeliveryfailed`: When an incoming AO Message encounters a permanent delivery error in MT/AT/SRI-SM, with or without store scenarios, then the configured value of this semi-static parameter determines the value of message state field in the optional TLV of the delivery receipt.
   - `hubnotifysmppmsgstatecancelled`: When an original AO Message is cancelled from the network/Mobile Centre, then the configured value of this semi-static parameter determines the value of the message state field in the optional TLV of the delivery receipt.

   Note: Refer to *Semi-Static Configuration* for more details.

2. Message_state text used in the short_message: This field is used to provide the informational content of final message state for an SMSC Delivery Receipt and inserted into the short message parameter of deliver_sm or data_sm operation.

   The value of this field can be configured through `hubnotifysmppmsgstatetextdeliveryfailed` and `hubnotifysmppmsgstatetextcancelled` semi-static parameters in the following scenarios:

   - `hubnotifysmppmsgstatetextdeliveryfailed`: When an incoming AO Message encounters a permanent delivery error in MT/AT/SRI-SM, with or without store scenarios, then the configured value of this semi-static parameter determines the value of the message state text to be used in short message of the delivery receipt.
   - `hubnotifysmppmsgstatetextcancelled`: When an original AO Message is cancelled from the network/Mobile Centre, then the configured value of semi-static parameter determines the value of the message state text to be used in short message of the delivery receipt.

3. Text (i.e. first 20 characters of the original submit message) used in the short message: This field is used to provide the informational content of the first 20 characters of the original submit message for an SMSC Delivery Receipt and inserted into the short message parameter of deliver_sm or data_sm operation.

   The configuration of this field is handle through semi-static parameter `includeuserdataindeliveryreceiptmessage` in the following ways:

When this parameter is set to 'TRUE':

- When the original submit message is GSM 7-bit encoding default alphabet, the first 20 characters of user-data of the submit original message is added after the label '`text:`' in the delivery receipt. For example:

```
id:IIIIIIIIII sub:SSS dlvrd:DDD submit date:YYMMDDhhmm done date:YYMMDDhhmm
stat:DDDDDDD err:000 text:<up to 20 character>
```

- When the original submit message is other than GSM 7-bit encoding alphabet, the user data is not being captured after the label '`text:`' in the delivery receipt. For example:

```
id:IIIIIIIIII sub:SSS dlvrd:DDD submit date:YYMMDDhhmm done date:YYMMDDhhmm
stat:DDDDDDD err:000 text:
```

- When the original submit message is GSM 7-bit encoding default alphabet and no user data is contained in the submit original message, then only the label '`text:`' is captured in the delivery receipt. For example:

```
id:IIIIIIIIII sub:SSS dlvrd:DDD submit date:YYMMDDhhmm done date:YYMMDDhhmm
stat:DDDDDDD err:000 text:
```

- The incoming AT notification, AGW (HUB) simply relay the received notification message. So there shall be no modification in the text field for the delivery receipt.

When this parameter is set to FALSE:

- The user data is not included in the delivery receipt. The label used in the delivery receipt is '`Text`:-'. For example:

```
id:IIIIIIIIII sub:SSS dlvrd:DDD submit date:YYMMDDhhmm done  date:YYMMDDhhmm
stat:DDDDDDD err:000 Text:-
```

Note: The functionality of the parameter `includeuserdataindeliveryreceiptmessage` depends on `includeuserdatainnotificationrequest`. For more details about `includeuserdataindeliveryreceiptmessage` refer to *Semi-Static Configuration*. For more details about `includeuserdatainnotificationrequest` refer to the RTR Operator Manual.

9.4 CIMD2 Interface

When using the CIMD2 interface, the application should connect to the HUB as a CIMD2 client to a CIMD2 server, as if it were an SMS application connecting to an SMSC.

The operations of the command and response strings for the available HUB function modules are described in corresponding chapters on the modules.

Note: This manual assumes that the reader is familiar with CIMD2 application programming. For more information about CIMD2 and application development using CIMD2, refer to *http://www.forum.nokia.com*.

9.4.1 CIMD2 Sessions

The standard procedure for interacting with the HUB using CIMD2 is:

1. Set up a TCP/IP connection between the remote SMS application and the HUB. The remote SMS application is responsible for initiating the connection and should manage this connection.

Note: If configured, the HUB will prompt each CIMD application connection with a greeting message. This greeting message can be configured in the parameter `hubPropCimdGreetingMessage`.

2. Open a CIMD2 session using the CIMD2 LOGIN operation.
3. After a positive LOGIN response, a session is established and the HUB can exchange CIMD2 messages with the remote SMS application.
4. When the session is over and no more interaction is required with the HUB, all CIMD2 sessions (and all TCP/IP connections) should be closed by the remote SMS applications, optionally using the CIMD2 LOGOUT operation.

Steps 1 and 2 should be repeated to set up multiple CIMD2 sessions between the SMS application and a HUB.

Note: A maximum of 255 simultaneous CIMD2 sessions can be established between an SMS application and a HUB (for every short number). One HUB can accept up a maximum of 1000 TCP/IP connections coming from SMS applications.

CIMD2 sessions can be disconnected by the HUB if one of the following occurs:

- The CIMD2 session does not carry CIMD2 traffic for a period longer than the specified max inactivity time.
- The HUB receives an invalid CIMD2 message on the session (e.g. CIMD2 messages longer than 1024 bytes).

9.4.2 CIMD2 Operations

The HUB supports the following CIMD2 operations:

Operation	Support
CIMD2 01 LOGIN	The open session request will be positively acknowledged if: • The User Identity and password combination matches • The short number is configured as an active CIMD2 application • The maximum number of sessions for this application is not exceeded
CIMD2 03 SUBMIT MESSAGE	The submit message request will be positively acknowledged if: • The AO functionality is enabled for this application • The AO throughput is not exceeded for this application • The screening of the message is successful (user data does not exceed 140 bytes or 160 7-bit characters) • The SRI-SM is successful • The first delivery attempt (FDA) relative value is 1 • The **Tariff Class** value is configured for this application • The **Service Description** value is configured for this application • The **Priority** value is configured for this application • The configured rules do not block this specific AO message
CIMD2 20 DELIVER MESSAGE	The deliver short message request will be initiated by the HUB if: • A routing rule is specified to route message towards this application

Operation	Support

	• The application is connected and the transmission window is not yet full
	• The AT throughput is not exceeded for this application
	• The following CIMD2 parameters are filled in conditionally:
	  • The originating IMSI is only passed to the application if the RTR/HUB has retrieved this IMSI or when it is passed by the MSC with the MO message (applicable for MAP phase 2+ only). Configuration determines the cases for which the RTR/HUB retrieves the originating IMSI.
	  • The originated visited MSC address is only passed when the SCCP calling party address of the UDT conveying the MO message has a routing indicator equal to GT.

CIMD2 23 DELIVER STATUS REPORT	The deliver status report operation will be initiated by the HUB if: • A corresponding CIMD2 message was received with status report requested and the notifications are allowed for this application • The session in which the corresponding CIMD2 was received in still open • The AT throughput is not exceeded for this application The HUB only uses value 3 (delivery failed) and value 4 (delivery successful).
CIMD2 DELIVERY REQUEST	The delivery request operation will return a DELIVERY REQUEST RESPONSE message indicating that the HUB does not have any messages pending for the application.
CIMD2 GET PARAMETERS	The only GET PARAMETER that is supported by the HUB is 501 (MC Time). Other values result in rejecting the GET PARAMETERS operation.

The CIMD2 SUBMIT MESSAGE parameters associated with the First Delivery Attempt Absolute functionality are always rejected by the HUB.

In addition to the CIMD2 operations listed above, the HUB also supports the CIMD2 operations LOGOUT and ALIVE.

Note: CIMD2 operations that are related to specific routings paths may not be allowed on your HUB due to license restrictions.

Chapter 10

Event Logging

Topics:

10.1 Introduction

To facilitate troubleshooting, the HUB logs all events to the `syslog`. The logged events include the application, service centre, error code, and error message, when applicable. For events in which the application is not known, the HUB logs a general event.

10.2 Event Filtering

10.2.1 Filtering on Severity Level

The HUB assigns a severity level to all incidents and events:

- Error
- Warning
- Info

It is possible to configure the minimum security level that an event must have to be logged. By default, only error and warning events are logged.

10.2.2 Filtering on Event Occurrences

The HUB only logs an event when:

- It occurs for the first time for an application;
- A configurable number of seconds have passed since the last logged event of this type for an application;
- The event has occurred a configurable number of times since the last logged event of this type for an application.

If an event occurs multiple times for an application, its counter is incremented in the `syslog`, until a configurable amount of time has passed or until the number of occurrences warrants a new log entry.

This functionality also applies to general events.

10.3 Application Events

The following table lists the application-related events that can be logged.

Event	Severity	Description
Outside Login Failures	Warning	General event. Wrong password (general), no inside sessions, too many sessions.
Outside Login Success	Info	

Event	Severity	Description
Outside Session Disconnect	Warning	The application closed the session, not the HUB (ask/trace).
Outside Session Closing	Warning	Session closed by the HUB.
Outside Rx Window Violation	Error	The application made an error sending a transaction twice (trace, check timeout settings).
Outside AT Response Time-out	Warning	The application did not respond in time.
Outside AT received NACK	Warning	The application sent a NACK on deliver.
Outside AO sent NACK	Warning	The system sent a NACK on submit.
Outside Inactivity Time-out	Warning	Session closed due to no activity.
Inside Login Failure	Error	The HUB could not log in at SMSC (check inside settings for application).
Inside Login Success	Info	
Inside Connect Failure	Error	TCP/IP session setup failed (retired later or wrong SCTP configuration).
Inside Session Disconnect	Error	Closed by external; SMSC closed the session.
Inside Session Closing	Warning	Closed by HUB; probably because there were not enough/no outside sessions.
Inside Rx Window Violations	Error	SCTP sent a transaction twice (trace, check timeout settings).
Inside AO Response Time-out	Warning	No response in time on forwarded submit.
Inside AO Received NACK	Warning	NACK received on forwarded submit.
Inside AT Sent NACK	Warning	NACK sent on received deliver.
AO Throughput Limit Crossing	Warning	RTR functionality: application(s) is/are sending too many messages.
AT Throughput Limit Crossing	Warning	RTR functionality: SMSC(s) is/are sending too many messages.

10.4 Examples

The following are sample event logs:

```
14:05:44.701 [LOG Warn.] App: 'Appl 1' - Inside AO response timeout
to 'sc1:node1:21006'

14:05:52.701 [LOG Warn.] App: 'Appl 1' - Inside AO response timeout
to 'sc1:node1:21006' [2 times]

14:05:53.201 [LOG Warn.] App: 'app2' - Inside session closed to
'sc1:node1:21006' - reason:Connection shutdown
```

```
14:05:55.155 [LOG Warn.] App: 'Appl 1' - Outside AO sent nack - UCP
error:4-Operation not allowed [48 times]
```

Chapter 11

Statistics

Topics:

11.1 Introduction

The HUB provides many statistics counters for all types of results and activities related to the reception and delivery of messages.

The statistics counters are defined in the HUB MIB and the APC MIB. For more details on the counters refer to these MIB files.

11.2 HUB Counters

Statistics counters for HUB inside and outside sessions are defined, for example:

- Total number of AO/AT messages send/received
- Total number of AO/AT messages rejected/successful/error
- Number of Temporary/Permanent Recipient/Permanent Message errors per destination.

11.3 Application Counters

Statistics counters for applications are defined, for example:

- Application inside and outside sessions counters
- Application MO/MT counters
- Application Send-Routing-Info-For-SM counters
- Application Report-SM-Delivery-Status counters
- Number of successful/failed deliveries for various routing path.

11.4 Operation Counters

The HUB includes statistics counters that track, per UCP, SMPP, and CIMD operation:

- Number of Requests
- Number of Acknowledgements (ACKs)
- Number of Negative Acknowledgements (NACKs)
- Total time spent in processing a message transfer requests
- Shortest response time for processing a message transfer request
- Time stamp when shortest response time of a message transfer request occurred
- Longest response time for processing a message transfer request
- Time stamp when longest response time of a message transfer request occurred.

The operation counters are defined per:

- HUB

- Application
- Service Centre
- Service Centre Node
- Service Centre Termination Point.

Chapter 12

Configuration

Topics:

12.1 Introduction

The HUB is a highly configurable product with a distributed architecture and a central configuration management, offering central configuration facility. The HUB configuration contains the following type of configuration data:

- Dynamic configuration—Defines SMS routing parameters required for the RTR to perform SMS routing, such as SMSCs, applications, and routing rules.
- Semi-static configuration—Defines fundamental HUB parameters for the HUB to function in its network environment, such as TCP/IP addressing.

The semi-static configuration is an XML-based file, described in this chapter. The dynamic configuration is stored by the MGR in a MySQL database.

12.2 Dynamic Configuration

The dynamic configuration defines SMS routing parameters required for the HUB and RTR to perform SMS routing, such as SMSCs, applications, and routing rules. The dynamic HUB configuration data contains:

- HUB devices
- Countries and networks
- Applications and SMSC
- Routing rules and modifiers
- Counting rules

This configuration is called dynamic because, in general, these parameters change frequently. These parameters are related to the routing behaviour of the HUB.

The dynamic configuration of the HUB is managed via the MGR.

12.3 Semi-Static Configuration

The semi-static configuration files are called semi-static because, in general, the parameters do not change frequently. Changing the parameters often affects other network elements or the network connectivity of the HUB. The semi-static configuration files are configured directly in XML.

The semi-static configuration defines fundamental HUB parameters such as:

- Parameters regarding SS7 addressing
- Parameters regarding TCP/IP addressing
- Parameters regarding billing
- Scripts that must be executed after completion of HUB configuration
- Specific parameters for AO and AT routing

The semi-static configuration consists of two files:

- Host-specific configuration file: Contains parameters for a specific HUB and is located at `/usr/TextPass/etc/<hostname>_config.txt`, where `<hostname>` is the host name of the HUB.
- Common configuration file: Contains parameters that are common to all HUBs and is located at `/usr/TextPass/etc/common_config.txt`.

Configuration parameters can be placed in either file. In case of a conflict in the settings of a parameter, the host-specific configuration file always takes precedence over the common configuration file.

12.3.1 UTF-8 Encoding

Only UTF-8 characters are supported in the semi-static configuration file. Ensure that the following line is present at the top of the file:

```
<?xml version="1.0" encoding="utf-8" ?>
```

12.3.2 tpconfig Entity

This section describes the `tpconfig` attributes.

12.3.2.1 hubcimdgreetingmessage

Mandatory/Optional

Optional

Location

Common configuration file

Description

The greeting message that sends when a CIMD application connects.

12.3.2.2 hubcloseinsideafterxtimeouts

Mandatory/Optional

Optional

Location

Common configuration file

Description

The number of transmit timeouts to occur before closing the inside session. Timeouts occurring in the same second will be counted as one. The default value of zero means that none of the inside sessions will be closed.

Valid Values

0-50 (default 0)

12.3.2.3 hubcloseoutsideafterxtimeouts

Mandatory/Optional

Optional

Location

Common configuration file

Description

The number of transmit timeouts to occur before closing the outside session. Timeouts occurring in the same second will be counted as one. The default value of zero means that none of the outside sessions will be closed.

Valid Values

0-50 (default 0)

12.3.2.4 hubclosesessionswithrst

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines how the connections from the HUB will be closed. The default "true" will close the connections using a TCP/IP RST. If it is set to "false", then all connections are closed using FIN.

Valid Values

- true (default)
- false

12.3.2.5 hubcommonerrorcodeforsmpp50errormapping

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the generic error code compatible with SMPP 3.3/3.4 to which HUB will convert the additional SMPP 5.0 error codes, while relaying an error response received on a SMPP 5.0 interface towards a SMPP 3.3/3.4 interface. The value of this parameter must be specified as a decimal integer and must correspond to a valid error code as per the SMPP 3.4 specification. Default value is 255, corresponding to ESME_RUNKNOWNERR (0x000000FF).

Vaild Values

All valid error codes as per the SMPP 3.4 specification (in decimal).

12.3.2.6 hubdcscharcodingconversion

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

This parameter indicates how the coding conversion, the custom character set conversion and the conversion from Data Coding to TP-DCS or vice-versa must be performed for inbound AO scenarios and MO AT scenarios.

Valid Values

- transparent: The transparent behavior of data coding scheme, coding and character set conversion.
- japan: Japanese network specific behavior of data coding scheme, coding and character set conversion.
- gsm23038: Behavior as per 3gpp specification will be followed (This option is part of a future scope, and is currently not supported).

Default

transparent (0)

12.3.2.7 hubdiscardonsmppservicetypemismatch

Mandatory/Optional

Optional

Location

Host configuration file

Description

Indicates whether the service type received in SMPP `submit_sm` or `data_sm` should be matched against the **SMPP Service Type** configured in the corresponding application. When this parameter is set to "true" and the service type does not match, then the message is rejected with the error `ESME_RINVSERTYP` (0x00000015).

Valid Values

- false (default)
- true

12.3.2.8 hubenableappinsidesessiontraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

Generate session traps for each outgoing and released connection of an external application (the session from the toward the termination point).

Valid Values

- true (default)
- false

12.3.2.9 hubenableapplicationtraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to 'true', application specific traps will be generated. Affected traps are those for creating new applications for learning.

Valid Values

- true
- false

Default

false

12.3.2.10 hubenableappoutsidesessiontraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

Generate session traps for each incoming and released session of an external application (the session from the application toward the HUB).

Valid Values

- true
- false (default)

12.3.2.11 hubenableinsidesmppunbind

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true", the HUB will send an unbind request to gracefully shut down an inside SMPP session. The HUB will close the session at the TCP level after it receives an unbind response or after the maximum response time (`insidesmppunbindmaxresponsetime`) has passed. If set to "false", the HUB will not send an unbind request before closing the session at the TCP level..

Valid Values

- true
- false (default)

12.3.2.12 hubenablemessageidprefixing

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true", SMPP message IDs originating from SMSCs or AMSs will be prefixed when forwarded to the application. For query/cancel/replace operations, the prefix will be removed before sending the request to the appropriate SMSC or AMS.

Note: Message prefixing is not available with SMPP v3.3 or with UCP.

Valid Values

- true
- false (default)

12.3.2.13 hubenableoutsidesmppunbind

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true", the HUB will send an unbind request to gracefully shut down an outside SMPP session. The HUB will close the session at the TCP level after it receives an unbind response or after the maximum response time (`outsidesmppunbindmaxresponsetime`) has passed. If set to "false", the HUB will not send an unbind request before closing the session at the TCP level.

Valid Values

- true
- false (default)

12.3.2.14 hubenablescnodesessiontraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

Generate session traps for each outgoing and released connection of an external service center node (the session from the toward the service center node of a service center termination point).

Valid Values

- true
- false (default)

12.3.2.15 hubenablesclterminationpointsessiontraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

Generate session traps for each outgoing and released connection of an external service center termination point (the session from the toward the service center termination point).

Valid Values

- true
- false (default)

12.3.2.16 hubenableservicecentresessiontraps

Mandatory/Optional

Optional

Location

Common configuration file

Description

Generate session traps for each outgoing and released connection of an external service center (the session from the toward the service center of a service center node).

Valid Values

- true
- false (default)

12.3.2.17 hubenabletcpkeepalive

Mandatory/Optional

Optional

Location

Common configuration file

Description

Controls the general enable/disable of the TCP keep-alive functionality. When set to true, TCP keep-alive is automatically enabled on every application session socket started thereafter. When set to false, TCP keep-alive is disabled unless you enable it for individual application(s) and/or termination point(s) in the MGR interface.

Valid Values

0-3600 (default 0)

12.3.2.18 hubenablevendorspecificbillingid

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true", this enables the use of a vendor specific optional field (TLV) for billing identification in SMPP `data_sm/submit_sm` messages and process the billing identification within the system. Refer to the `hubvendorspecificbillingidtag` and `hubvendorspecificbillingformattag` parameters.

Valid Values

- true
- false (default)

12.3.2.19 hubenablevendorspecificcontentrating

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true", this enables the vendor-specific optional field (TLV) for content rating in SMPP `data_sm/submit_sm` messages. Refer to the `hubvendorspecificcontentratingtag` parameter.

Valid Values

- true
- false (default)

12.3.2.20 hubinsidesmppunbindmaxresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the maximum number of seconds that the HUB will wait for an unbind response before closing the session at the TCP level. This parameter only applies if `enableinsidesmppunbind` is set to "true".

Valid Values

1-3600 (default 5)

12.3.2.21 hubipaddressowninternal

Mandatory/Optional

Optional

Location

Host-specific configuration file

Description

The internal IPv4 address/hostname of this HUB node. If set, this address will be used as source address for setting up inside sessions. For more information on configuring hostname, refer to section *DNS Query Mechanism* .

12.3.2.22 hubipfailovercontrol

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates whether the HUB should do IP failover control.

Valid Values

- true
- false

Default

false

12.3.2.23 hubipfailovertimeout

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds that the HUB waits before doing the actual failover.

Valid Values

0-3600

Default

5

12.3.2.24 hublegacyucpapplicationsupport

Mandatory/Optional

Optional

Location

Common configuration file

Description

Enables or disables support for legacy UCP applications. Legacy UCP application support is enabled when this parameter is set to "onerror". When enabled, the HUB converts UCP52/53 request messages into a legacy UCP01 request when the UCP52/53 request was negative acknowledged with error code 03. The HUB will only send a UCP01 request toward a legacy application when it is able to convert a UCP52/53 request into a legacy UCP01 request.

Valid Values

- disabled (default)
- onerror

12.3.2.25 hublifecheckinterval

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds between sending a lifecheck message on an active connection. This lifecheck functionality is only used for inside sessions (to keep the connection with the SMSC alive). The value 0 disables the lifecheck.

Valid Values

0-3600 (default 60)

12.3.2.26 hublogintimeout

Mandatory/Optional

Optional

Location

Common configuration file

Description

Time in seconds the HUB waits for a log-in request when a connection for an outside session is established. If no log-in request is received within this time, the connection will be closed.

Valid Values

0-3600 (default 5)

12.3.2.27 hublogparsingerrors

Mandatory/Optional

Optional

Location

Common configuration file

Description

Log parsing errors to syslog.

Valid Values

- true (default)
- false

12.3.2.28 hubmaxdelayedmessages

Mandatory/Optional

Optional

Location

Common configuration file

Description

The maximum number of delayed messages allowed in the system.

Valid Values

1000-100,000 (default 1000)

12.3.2.29 hubmaxmipresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds the HUB waits for a response from a RTR after sending a MIP message.

Valid Values

1-3600 (default 5)

12.3.2.30 hubmaxmxpresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds the HUB waits for a response from a RTR after sending a MXP message. Note that if the RTR attempts a MT delivery, it will only respond to the HUB after that attempt is completed.

Valid Values

1-3600 (default 30)

12.3.2.31 hubmaxmxptprretries

Mandatory/Optional

Optional

Location

Common configuration file

Description

The number of times another RTR is retried in case a node fails or an MXP request times out. RTRs that are already tried are not retried again for this request. In case the value is 0 then no retries are done. Keep in mind that it could be possible that the original request times out because the total retry time is too long. The setting of `hubmaxmxpresponsetime` is important to calculate the maximum response time.

Valid Values

0-3 (default 1)

12.3.2.32 hubmaxnotifyctxts

Mandatory/Optional

Optional

Location

Common configuration file

Description

The maximum number of notification contexts this HUB can store. The HUB will remember each forwarded submission with a notification request. When the notification is received for this session, the HUB will forward it to the original session.

Setting this property to 0 will disable the notification context storage and will result in the notifications being sent to any session. Already allocated context will be freed when the reallocated session is closed. A value between 1 and 999 will not be accepted.

Valid Values

0-1,000,000 (default 500,000)

12.3.2.33 hubmaxqueuedmessages

Mandatory/Optional

Optional

Location

Common configuration file

Description

Maximum number of AO messages that can be delayed in the HUB at one time (for all configured applications). The number of seconds for which each message is delayed is configured in each application's **AO Delay Time** parameter in the MGR.Setting an AO delay time does not change the way the HUB handles window size. The HUB continues to discard all requests that are received outside the window size for a session.

Valid Values

1000-100,000 (default 1000)

12.3.2.34 hubmaxresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Maximum time (in seconds) that the RTR waits for a response from the HUB before it considers the relating message as timed out.

Default

15

12.3.2.35 hubmaxsyntaxerrors

Mandatory/Optional

Optional

Location

Common configuration file

Description

The number of consecutive syntax errors that may be received on a session. After receiving this amount of messages with syntax errors, the connection will be closed.

Valid Values

0-3 (default 3)

12.3.2.36 hubmaxtotalapplicationsessions

Mandatory/Optional

Optional

Location

Common configuration file

Description

Maximum number of total application sessions that may be active.

Valid Values

1-12,000 (default 1000)

12.3.2.37 hubnotifysmppmsgstatecancelled

Mandatory/Optional

Optional

Location

Host/Common configuration file

Description
This parameter is used to set the message_state in the optional TLV of the Delivery Receipts when a message has been cancelled from the network/Mobile Centre, i.e. the message is cancelled by application using the cancel_sm request.

Note: It is possible to set this semi-static parameter according to the following valid values, but some of the values may not be valid in SMPP 3.3 and SMPP 3.4.

For example: The values `scheduled` and `skipped` have been introduced in the SMPP 5.0 and setting these values for SMPP 3.4 is possible but it is illegal according to the spec.

Valid Values

- `scheduled`
- `enroute`
- `delivered`

- `expired`
- `deleted`
- `undeliverable`
- `accepted`
- `unknown` (default)
- `rejected`
- `skipped`

12.3.2.38 hubnotifysmppmsgstatedeliveryfailed

Mandatory/Optional

Optional

Location

Host/Common configuration file

Description
This parameter is used to set the message_state in the optional TLV of the Delivery Receipts when an incoming AO Message encountered a permanent delivery error in MT/AT/SRI-SM, with or without Store Scenarios, and deemed permanently undeliverable.

Note: It is possible to set this semi-static parameter according to the following valid values, but some of the values may not be valid in SMPP 3.3 and SMPP 3.4.

For example: The values `scheduled` and `skipped` have been introduced in the SMPP 5.0 and setting these values for SMPP 3.4 is possible but it is illegal according to the spec.

Valid Values

- `scheduled`
- `enroute`
- `delivered`
- `expired`
- `deleted`
- `undeliverable` (default)
- `accepted`
- `unknown`
- `rejected`
- `skipped`

12.3.2.39 hubnotifysmppmsgstatetextcancelled

Mandatory/Optional

Optional

Location

Host/Common configuration file

Description
This parameter is used to set the message_state, text used in short_message of Delivery Receipts, when a message has been cancelled from the network/Mobile Centre, i.e. message is cancelled by application using the cancel_sm request.

Note: It is possible to set this semi-static parameter according to the following valid values, but some of the values may not be valid in SMPP 3.3.

For example: The value `rejectd` has been introduced in the SMPP 3.4 and setting this value for SMPP 3.3 is possible but it is illegal according to the spec.

Valid Values

- `delivrd`
- `expired`
- `deleted` (default)
- `undeliv`
- `acceptd`
- `unknown`
- `rejectd`

12.3.2.40 hubnotifysmppmsgstatetextdeliveryfailed

Mandatory/Optional

Optional

Location

Host/Common configuration file

Description
This parameter is used to set the message_state text, used in short_message of Delivery Receipts, when an incoming AO Message encountered a permanent delivery error in MT/AT/SRI-SM, with or without Store Scenarios, and deemed permanently undeliverable.

Note: It is possible to set this semi-static parameter according to the SmppMessageStateText, but some of the values may not be valid in SMPP 3.3.

For example: The value `rejectd` has been introduced in the SMPP 3.4 and setting this value for SMPP 3.3 is possible but it is illegal according to the spec.

Valid Values

- `delivrd`
- `expired`
- `deleted`

- undeliv (default)
- acceptd
- unknown
- rejectd

12.3.2.41 huboutsidesmppunbindmaxresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the maximum number of seconds that the HUB will wait for an unbind response before closing the session at the TCP level. This parameter only applies if enableoutsidesmppunbind is set to "true".

Valid Values

1-3600 (default 5)

12.3.2.42 hubpropusenotificationaddressinucp53

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to *true* then the original notification address is used in the UCP53 message. This only applies when a notification address is given of type TCP/IP. Otherwise the original OAdC will be used. The default behaviour ("false") is that the short number of the destination application will be filled in.

Valid Values

- true
- false (default)

12.3.2.43 hubreconnectdelay

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds to wait before reconnecting a HUB initiated session that was disconnected by the remote end. The value 0 disables reconnecting.

Valid Values

0-3600 (default 5)

12.3.2.44 hubrejectwindowviolations

Mandatory/Optional

Optional

Location

Common configuration file

Description

Flag indicating whether messages exceeding the configured window size should be responded to with a negative acknowledgement. Normal behaviour is to silently discard the messages violating the window size.

Valid Values

- true
- false (default)

12.3.2.45 hubrouteconcatenated

Mandatory/Optional

Optional

Location

Common configuration or Host-Specific configuration file

Description

This field fine-tunes the way in which a HUB distributes AO messages to the available Routers.

The HUB normally applies a Round-Robin scheme aiming at an equal load. For concatenated segments the following options exist to balance that aim with a possible need to strongly enforce in-order-delivery of segments:

- Distribute by Round-Robin method. This will attempt to distribute all segments equally to routers; this may cause out-of-order delivery and result in problems on outdated user equipment.

- Use same RTR for concatenated SMS to same recipient. This avoids out-of-order delivery but may cause high throughput os select routers if many messages are aimed at the same recipients or the ratio of concatenated messages is high due to content, language and alphabet.
- Base decision on recipient plus reference number. This is a compromise that uses the same router for the segments of a message but varies routers between messages.

Vaild Values

- 0 Distribute by Round-Robin method
- 1 Use same RTR for concatenated SMS to same recipient (default)
- 3 Base decision on recipient plus reference number

12.3.2.46 hubscterminationpointretrydelay

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds to wait before a termination point may be retried after it has become unreachable.

Valid Values

1-3600 (default 60)

12.3.2.47 hubsessionusagethreshold1

Mandatory/Optional

Optional

Location

Common configuration file

Description

Threshold 1, expressed as a percentage, used for generation of traps that inform about the total usage of application sessions. When the usage starts exceeding this threshold, generates a hubSessionUsageGrowing trap. When the rate start getting below this threshold, generates a hubSessionUsageShrinking trap. The value for this threshold should be less than the value for threshold 2.

Valid Values

0-100 (default 50)

12.3.2.48 hubsessionusagethreshold2

Mandatory/Optional

Optional

Location

Common configuration file

Description

Threshold 2, expressed as a percentage, used for generation of traps that inform about the total usage of application sessions. When the usage starts exceeding this threshold, generates a hubSessionUsageGrowing trap. When the rate start getting below this threshold, generates a hubSessionUsageShrinking trap. The value for this threshold should be greater than the value for threshold 1 and less than the value for threshold 3.

Valid Values

0-100 (default 70)

12.3.2.49 hubsessionusagethreshold3

Mandatory/Optional

Optional

Location

Common configuration file

Description

Threshold 3, expressed as a percentage, used for generation of traps that inform about the total usage of application sessions. When the usage starts exceeding this threshold, generates a hubSessionUsageGrowing trap. When the rate start getting below this threshold, generates a hubSessionUsageShrinking trap. The value for this threshold should be greater than the value for threshold 2 and less than the value for threshold 4.

Valid Values

0-100 (default 85)

12.3.2.50 hubsessionusagethreshold4

Mandatory/Optional

Optional

Location

Common configuration file

Description

Threshold 4, expressed as a percentage, used for generation of traps that inform about the total usage of application sessions. When the usage starts exceeding this threshold, generates a hubSessionUsageGrowing trap. When the rate falls below this threshold, generates a hubSessionUsageShrinking trap. The value for this threshold should be greater than the value for threshold 3 and less than the value for threshold 5.

Valid Values

0-100 (default 95)

12.3.2.51 hubsessionusagethreshold5

Mandatory/Optional

Optional

Location

Common configuration file

Description

Threshold 5, expressed as a percentage, used for generation of traps that inform about the total usage of application sessions. When the usage starts exceeding this threshold, generates a hubSessionUsageGrowing trap. When the rate falls below this threshold, generates a hubSessionUsageShrinking trap. The value for this threshold should be greater than the value for threshold 4.

Valid Values

0-100 (default 100)

12.3.2.52 hubsmpppluspprefixhandling

Mandatory/Optional

Optional

Location

Common configuration or Host-Specific configuration file

Description

If this parameter is set to true, HUB will perform two actions:

1. When HUB receives a SUBMIT_SM or DATA_SM message with a source address containing a '+' (plus) sign in the first position of the address, source address TON="INTERNATIONAL" ('1') and source address NPI="ISDN E163/E164" ('1'), HUB will remove the + sign.
2. When HUB receives a SUBMIT_SM or DATA_SM message with a destination address containing a '+' (plus) sign in the first position of the address, destination address TON="INTERNATIONAL" ('1'), destination address NPI="ISDN E163/E164" ('1'), HUB will remove the '+' sign.

If this parameter is set to FALSE, HUB will not remove '+' sign from source/destination address fields in incoming SUBMIT_SM or DATA_SM messages. A '+' sign at the first position of the source/destination address field will cause the message to be discarded even if the TON="INTERNATIONAL" and the NPI="ISDN E163/E164".

Valid Values

- true
- false (default)

12.3.2.53 hubsmppv34notificationtextformat

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies how the HUB will handle message IDs in the text of SMPP v3.4 (and later) delivery notifications:

- notext: No notification text is used in delivery notifications.
- emptyidonexcessdigits: When the message ID exceeds 10 digits, the `id` field is left empty.
- truncate: When the message ID exceeds 10 digits (in hex format), it is truncated to 8 hex digits and presented as a decimal value. If the message ID is not numeric, the `id` field is left empty. This is the default option.
- decimal: The `id` field contains the decimal value of the message ID, which is potentially more than 10 digits. If the message ID is not numeric, the `id` field is left empty.
- hex: The `id` field contains the hexadecimal value of the message ID, which is potentially more than 10 digits. If the message ID is not numeric, the `id` field is left empty.
- transparent: The message ID (hex or ASCII) is placed in the `id` field with no modification.

Valid Values

- notext
- emptyidonexcessdigits
- truncate (default)
- decimal
- hex
- transparent

12.3.2.54 hubucpaddressconversion

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines how the outgoing UCP AdC/OAdC addresses are handled. When set to "stripOwnCountry", addresses containing the international prefix and the country code of the HUB itself will be converted to national format. When set to "transparent", no conversion takes place.

Valid Values

- transparent
- stripowncountry (default)

12.3.2.55 hubucpdeletetext3

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the first configurable text item (`[TEXT3]`) of a UCP response delete message (UCP58). This text item is used in the AMsg parameter of the UCP58 request. The content of the AMsg parameter is formatted as:

```
[TEXT3] &lt;AdC&gt; [TEXT4] {TIMESTAMP} [TEXT5]
```

Where `{TIMESTAMP}` is a list of matching service center timestamps, separated by spaces. UTF-8 string is allowed when it can be mapped to the GSM default alphabet.

Valid Values

Default is "Destination".

12.3.2.56 hubucpdeletetext4

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the second configurable text item ([TEXT4]) of a UCP response delete message (UCP58). Refer to hubucpdeletetext3 for more information.

Valid Values

Default is "identification:".

12.3.2.57 hubucpdeletetext5

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the third configurable text item ([TEXT5]) of a UCP response delete message (UCP58). Refer to hubucpdeletetext3 for more information.

Valid Values

Default is "has been deleted.".

12.3.2.58 hubucphandlingucp02

Mandatory/Optional

Optional

Location

Common configuration file

Description

This parameter specifies how UCP 02 (multiple address call input operation) requests should be handled by the HUB. A UCP 02 request can be handled as follows:

- forwardunknown: The UCP 02 request is handled as an unknown request and the HUB will forward the UCP 02 request as an unknown message to the SMSC. The message is sent as a UCP 02.
- notsupported: A NACK is returned with error 'operationNotSupported'. The message is not handled by the HUB.

- `splitup`: Divides the UCP 02 request into single submit requests. The UCP 02 messages are split up and sent as UCP 51 messages. The HUB will acknowledge the pending UCP 02 request when all divided submits are processed by the HUB.

Valid Values

- forwardunknown (default)
- notsupported
- splitup

12.3.2.59 hubucpinquirytext1

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the first configurable text item (`[TEXT1]`) of a UCP response inquiry message (UCP57). This text item is used in the AMsg parameter of the UCP57 request. The content of the AMsg parameter is formatted as:

```
[TEXT1] &lt;AdC&gt; [TEXT2] {TIMESTAMP}
```

Where `{TIMESTAMP}` is a list of matching service center timestamps, separated by spaces. UTF-8 string is allowed when it can be mapped to the GSM default alphabet.

Valid Values

Default is "Destination".

12.3.2.60 hubucpinquirytext2

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the second configurable text item (`[TEXT2]`) of a UCP response inquiry message (UCP57). Refer to `hubucpinquirytext1` for more information.

Valid Values

Default is "identification:".

12.3.2.61 hubucplonginquirydeleteresponse

Mandatory/Optional

Optional

Location

Common configuration file

Description

This parameter is used to enable or disable the support for a long AMsg filed for UCP 57 and UCP 58 response messages.

When set to "false" the HUB will allow a maximum AMsg of 160 characters. When set to "true", the HUB will allow a maximum AMsg of 640 characters for UCP 57 and UCP 58 messages.

Valid Values

- true
- false (default)

12.3.2.62 hubucpmt4defaultgsmencoding

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines how the encoding of message data in a UCP, MT=4 (transparent message) with DCS set to GSM 7-bit is to be interpreted. Normally the message data is encoded in 7-bit packed format for transparent forwarding to a handset, but sometimes 8-bit encoding is used.

Valid Values

- packed7bit (default)
- unpacked8bit

12.3.2.63 hubuserecipientaddressinucp52

Mandatory/Optional

Optional

Location

Common configuration file

Description

If set to "true" then the given recipient address is used in the UCP52 message. This only applies when the recipient address is available. Otherwise the short number will be used. The default behaviour ("false") is that the short number of the destination application will be filled in.

Valid Values

- true
- false (default)

12.3.2.64 hubvendorspecificbillingformattag

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the billing format tag that is used as prefix for the billing identification for processing within the system. VSBI should be enabled (see `hubenablevendorspecificbillingid`) for this setting to have any consequence.

Valid Values

255

12.3.2.65 hubvendorspecificbillingidtag

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the TLV tag value of the vendor specific billing identification. VSBI should be enabled (see `hubenablevendorspecificbillingid`) for this setting to have any consequence.

Valid Values

5121 (i.e. 0x1401)

12.3.2.66 hubvendorspecificcontentratingtag

Mandatory/Optional

Optional

Location

Common configuration file

Description

Specifies the TLV tag value of the vendor-specific content rating. VSCR should be enabled (see `hubenablevendorspecificcontentrating`) for this setting to have any consequence.

Valid Values

5124 (i.e. 0x1404)

12.3.2.67 hubverifychecksum

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines whether the checksum in incoming UCP and CIMD, if present, messages shall be verified. If verification of the checksum is enabled and a message is received for which the checksum verification fails, the message will be rejected.

Valid Values

- true
- false (default)

12.3.2.68 includeuserdataindeliveryreceiptmessage

Mandatory/Optional

Optional

Location

Common configuration file

Description

This parameter indicates whether user data is included in the delivery receipt or not. When this parameter is set to true and message encoding is GSM 7-bit default alphabet, then the first 20 characters of the user-data of the original message will be added after the label '`text:`' in the delivery receipt. For other message encoding, the user-data will not be captured after the label '`text:`' in the delivery receipt.

When this parameter is set to false, the user data is not included in the delivery receipt. The label used in the delivery receipt will be '`Text:-`'.

Valid Values

- true
- false (default)

12.3.2.69 runtexthubprocess

Mandatory/Optional

Mandatory (for running the HUB)

Location

Host-specific configuration file

Description

Indicates whether the HUB process should be started on this node. Should be "true" for running the HUB.

Valid Values

- true
- false

Default

false

12.3.2.70 ipaddress

Mandatory/Optional

Optional

Location

Host-specific configuration file

Description

IP address of the server.

12.3.2.71 dnsfailtimeout

Mandatory/Optional

Optional

Location

Common configuration file

Description

The number of seconds after which DNS lookup shall be re-attempted if a DNS lookup failure occurs. If the value of this parameter is "0", then no further DNS queries shall be performed.

Valid Values
0-3600(seconds). Default is 60 seconds.

12.3.2.72 hubipv6addressowninternal

Mandatory/Optional

Optional

Location

Host-Specific configuration file

Description

The internal IPv6 address/hostname of the HUB node. If set, this address shall be used as source address for setting up inside sessions. For more information on configuring hostname, refer to section *DNS Query Mechanism* .

12.3.2.73 snmppropalarmownipaddress

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description

The IPv4 address of the TextPass node. If set, this address will be used as source address for sending SNMP traps. This parameter is used to populate Trap Agent address for IPv4/IPv6 address. If this parameter is not set, then the IPv4 address of first network interface is used to populate Trap Agent Address in SNMP Traps.

Valid Value
IPv4 address

Default

Empty string

12.3.2.74 snmppropalarmownipv6address

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description

The IPv6 address or hostname of the TextPass node. If set, this address will be used as source address for sending SNMP traps.

Valid Values

IPv6 address or hostname(maximum length can be of 255 characters)

Default

Empty string

12.3.2.75 snmpproplistenaddresstype

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description

This parameter indicates whether SNMP Listener type is IPv4 only or Dual-stack.

Valid Values

ipv4 or dual

Default

ipv4

12.3.3 hubnotificationtext Entity

This section describes the `hubnotificationtext` attributes. `hubnotificationtext` should be configured in the common configuration file.

12.3.3.1 notificationstatus

Mandatory/Optional

Optional

Location

Common configuration file

Description

The delivery status within the HUB.

Valid Values

- inProgress
- validityExpired
- deliveryFailed
- deliverySuccessful
- cancelled
- deleted
- default

12.3.3.2 formatstring

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

The notification text string (in UTF-8).

Valid Values

Only characters that are valid in the GSM default alphabet are allowed. The maximum number of GSM 7-bit characters is 140, inclusive of escape characters.

The following substitution parameters are available:

Parameter	Will be replaced with...
%MRAD	Recipient address in national format
%MERS	Error text, as configured the `hubnotificationerrormapping errortext`
%MERR	Error code, as configured in the `hubnotificationerrormapping defaulterrorcode`
%SDY	Adapted submit date two-digit year (00-99)
%SDX	Adapted submit date four-digit year (1970-2038)
%SDM	Adapted submit date month (01-12)
%SDD	Adapted submit date day (01-31)
%STH	Adapted submit date time in 24-hour notation (00-23)
%STI	Adapted submit date time in 12-hour notation (00-12)
%STM	Adapted submit date minutes (00-59)
%STS	Adapted submit date seconds (00-59)
%STP	Adapted submit date time indicator (AM or PM)
%DDY	Delivery date two-digit year (00-99)
%DDX	Delivery date four-digit year (1970-2038)
%DDM	Delivery date month (01-12)
%DDD	Delivery date day (01-31)
%DTH	Delivery date time in 24-hour notation (00-23)
%DTI	Delivery date time in 12-hour notation (00-12)
%DTM	Delivery date minutes (00-59)
%DTS	Delivery date seconds (00-59)
%DTP	Delivery date time indicator (AM or PM)
%%	Escape character to use when the percentage sign (%) needs to be included in the notification text

If multiple messages for the same recipient arrive in the same second, the Mobile Messaging system increases the submit time. This is the adapted submit time. This ensures that the submit time is unique.

Default

An empty string.

12.3.3.3 usageofdefaults

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Determines when the `hubnotificationtext` `defaulterrorcode`, `defaulterrortext`, and `defaultucpreasoncode` should be used.

Valid Values

- usageIncaseOfNoError: Only use the `hubnotificationtext` defaults when none of the HUB's internal, normalized errors are reported
- never: Never use the `hubnotificationtext` defaults
- always: Always use the `hubnotificationtext` defaults (never use the `hubnotificationerrormapping`)

Default

usageIncaseOfNoError

12.3.3.4 defaulterrorcode

Mandatory/Optional

Optional

Location

Common configuration file

Description

The default notification error code, which applies when no internal error is reported. `formatstring` specifies how this error can displayed in the notification text.

Valid Values

0-99999

Default

0

12.3.3.5 defaulterrortext

Mandatory/Optional

Optional

Location

Common configuration file

Description

The default notification error text (in UTF-8), which applies when no internal error is reported.

Valid Values

Only characters that are valid in the GSM default alphabet are allowed. The maximum number of GSM 7-bit characters is 100, inclusive of escape characters.

Default

An empty string.

12.3.3.6 defaultucpreasoncode

Mandatory/Optional

Optional

Location

Common configuration file

Description

The default notification error UCP reason code, which applies when no internal error is reported. The reason code will be stored in `Rsn` field of the UCP 53 request.

Valid Values

0-255

Default

0

12.3.4 hubnotificationerrormapping Entity

This section describes the `hubnotificationerrormapping` attributes. `hubnotificationerrormapping` should be configured in the common configuration file.

12.3.4.1 id

Mandatory/Optional

Optional

Location

Common configuration file

Description

SNMP index of the HUB notification error mapping item.

12.3.4.2 internalerror

Mandatory/Optional

Optional

Location

Common configuration file

Description

The HUB's internal, normalized error that should be mapped to a notification error code, error text, and UCP reason code (`Rsn` field).

Default

The HUB's default notification error mapping is:

id	internalerror	errorcode	errortext	ucpreasoncode
1	defaultNotificationMapping	108	Temporary Delivery Failure	108
2	mtTimeout	10	Network time-out	10
3	sriSmTimeout	10	Network time-out	10
4	sriSmUnknownSubscriber	101	Unknown subscriber	101
5	mtAbsentSubscriber	107	Absent subscriber	107
6	sriSmAbsentSubscriber	107	Absent subscriber	107
7	mtFacilityNotSupported	106	Facility not supported	106
8	sriSmFacilityNotSupported	106	Facility not supported	106
9	sriSmSystemFailure	120	HLR system failure	120
10	mtSystemFailure	118	System fail	118
11	sriSmCallBarred	103	Call barred	103
12	mtOtherMapError	126	System failure	126
13	sriSmOtherMapError	126	System failure	126
14	mtTcapAborted	126	System failure	126
15	mtSccpAborted	126	System failure	126
16	sriSmTcapAborted	126	System failure	126
17	sriSmSccpAborted	126	System failure	126
18	mtUnexpectedDataValue	127	Unexpected data value	127
19	mtEquipmentProtocolError	110	Protocol Error	110
20	mtEquipmentNotSmEquipped	111	MS not equipped	111

id	internalerror	errorcode	errortext	ucpreasoncode
21	mtIllegalSubscriber	115	MS not a subscriber	115
22	mtUnidentifiedSubscriber	115	MS not a subscriber	115
23	mtNoPagingResponse	107	Absent Subscriber	107
24	mtImsiDetachError	107	Absent Subscriber	107
25	mtRoamRestriction	107	Absent Subscriber	107
26	sriSmMsDeregistered	107	Absent Subscriber	107
27	sriSmMsPurged	107	Absent Subscriber	107
28	mtShortMsgTypeNotSupported	110	Protocol Error	110
29	mtCanNotReplaceShortMsg	110	Protocol Error	110
30	mtUnspecifiedProtocolId	110	Protocol Error	110
31	mtMsgClassNotSupported	110	Protocol Error	110
32	mtUnspecifiedDataCodingScheme	110	Protocol Error	110
33	mtTpduNotSupported	110	Protocol Error	110
34	mtNoSmStorageCapabilityInSim	110	Protocol Error	110
35	mtErrorinMs	110	Protocol Error	110
36	mtSimAppToolKitBusy	110	Protocol Error	110
37	mtSimDataDownloadError	110	Protocol Error	110
38	mtApplSpecificError	110	Protocol Error	110
39	mtEquipUnspecifiedErrorCause	110	Protocol Error	110
40	mtUeDeregistered	107	Absent Subscriber	107
41	mtNoResponseViaIpsmGw	107	Absent Subscriber	107

If desired, you can modify the `errortext` and/or extend the mapping table with more internal errors. It is not recommended that you change the internal error mapping.

12.3.4.3 errorcode

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

The notification error code. You can show this error code in the `hubnotificationtext formatstring` by using the `%MERR` substitution parameter.

Valid Values

0-99999

Default

0

12.3.4.4 errortext

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

The notification error text. You can show this error text in the `hubnotificationtext formatstring` by using the `%MERS` substitution parameter.

Valid Values

Only UTF-8 characters are allowed.

Default

An empty string.

12.3.4.5 ucpreasoncode

Mandatory/Optional

Optional

Location

Common configuration file

Description

The notification error UCP reason code. The reason code will be stored in `Rsn` field of the UCP notification message.

Valid Values

0-255

Default

0

12.3.5 Network Discovery Configuration

To enable communication between RTRs, HUBs, AMSs, and IIWs, network discovery must be configured with the following `tpconfig` attributes in `common_config.txt`:

- `networkdiscoverymulticastaddress`
- `networkdiscoverynetworkaddress`
- `networkdiscoverynetworkmask`

These values may only be changed when the `deviceAdminState` of the device in question is "inactive".

NewNet Mobile Messaging nodes use network discovery to notify each other of their presence in the network. All nodes send out heartbeats via UDP multicast to inform other nodes of their presence. These heartbeats contain the necessary parameters to set up the communication channels between the nodes. The default heartbeat interval is 5 seconds.

All components are aware of each other's presence. The communication channels are set up only between specific components: HUB-HUB, HUB-RTR, AMS-AMS, RTR-AMS, IIW-IIW, and RTR-IIW.

Because the IP TTL is 1, all NewNet Mobile Messaging nodes should be on the same subnet for network discovery to work.

For troubleshooting purposes, use `tp_walk networkDiscoveryTable` to see all discovered nodes/products.

12.3.6 Activating Configuration Files

To activate the configuration files, execute the following command per HUB:

```
tp_config [<host-specific file> [<common config file>]]
```

If no configuration files are specified as command argument, `tp_config` will assume that it needs to use both configuration files.

Note: When `tp_config` is executed, the HUB will restart. Because restarting the HUB affects service, it is recommended to activate configuration files during low-traffic hours. Note that the `tp_start` command, which starts the HUB, executes `tp_config` as well.

12.4 Configuration File Distribution

The `tp_fclient` tool manages the replication of XML configuration data files from a server. `tp_fclient` enables a client system to subscribe to all changed XML configuration data files that the MGR on the assigned server produces.

The `tp_fserver` tool manages the replication of XML configuration data files from a server. `tp_fserver` enables a server system to interact with clients.

For more information about `tp_fclient` and `tp_fserver`, refer to the Tools Operator Manual.

tp_fclient Configuration

To configure the `tp_fclient` tool:

1. To enable the `tp_start` tool to start the `tp_fclient` tool, add the following line to the host-specific configuration file (located in `usr/TextPass/etc`):
   ```
   runtpfclientprocess="true"
   ```
2. Remove any instances of `tp_fclient` as `postbootscript` command in the host-specific configuration file. For example:
   ```
   <postbootscript command="/usr/TextPass/bin/tp_fclient a.b.c.d"/>
   <postbootscript command="/usr/TextPass/bin/tp_fclient --continuous a.b.c.d &amp;amp;"/>
   ```
3. In the `/etc/hosts` file, add the host name (or alias) of the MGR node.

When `tp_start` starts multiple applications at the same time, it will always start the `tp_fclient` process first. `tp_start` will first invoke `tp_fclient` without the `--continuous` option, which will cause `tp_fclient` to retrieve files once unconditionally. `tp_start` will start `tp_fclient` subsequently without the `--continuous` option, which will cause `tp_fclient` to retrieve files only when a change occurs.

Note: The files in `/usr/local/apache/mBalance/TPManager/data` that are replicated by the file transfer utilities must be owned by the user textpass that starts the scripts. If not, an error will occur and the script will abort.

Starting and Stopping tp_fclient

The `tp_fclient` tool can also be started and stopped independently:

```
$ tp_start --tp_fclient
$ tp_stop --tp_fclient
```

Chapter 13

Security

Topics:

13.1 Introduction

This chapter provides an overview of the HUB's security.

13.2 Controlling System Access

System access to the HUB system should be tightly controlled, because system access allows the users to execute maintenance commands and to gather confidential large account information. Access to the *root* account on the HUB should especially be limited to a small number of people.

Refer to the Sun Solaris documentation for more information.

For Linux implementations refer to the Red Hat Enterprise Linux documentation.

13.3 User Group Privileges

The default user groups are:

- Group administrator—Users in this group have all privileges
- Group super user—Users in this group also have all privileges
- Group customer support—Users in this group have view rights and the right to change their own settings

For more information about user groups, refer to the MGR Operator Manual.

13.4 Authenticating Applications

Authentication of application access to SMS RTRs is controlled by using a login UCP 60 operation (i.e. short number and password) and SMPP bind operations. Configurable password complexity applies for application passwords. systems are assumed to operate in a network environment that is secured by means of firewalls and corresponding adequate security measures.

13.5 Remote System Access

Because the HUB is typically connected to the internet via a firewall, it is important to prohibit remote system access via insecure means (Telnet/rlogin) through the Internet. A good policy is to prohibit insecure remote system access and only allow remote system access via SSH.

13.6 Keeping Software Up-to-Date

To limit exposure to known security weaknesses, it is important to keep the (system) software up-to-date by applying OS (security) patches. Please check with NewNet Support to verify if new patches are supported by NewNet.

13.7 Hardening Solaris

The HUB Solaris configuration can be hardened by the Sun Solaris Security Toolkit (JASS). This toolkit stops many unnecessary services and configures other services in a more secure way. Other security toolkits are not supported.

Please note that hardening Solaris should not be done on current production systems. After applying the toolkit, the system should be thoroughly tested before it is used again in a production situation. Inform your system integrator or NewNet before applying the toolkit.

13.8 Detecting and Reporting Security Violations

Access to log files or audit files and other system resources on systems is controlled using available Sun Solaris security mechanisms.

Refer to the Sun documentation for more information.

For Linux implementations refer to the Red Hat Enterprise Linux documentation.

Chapter 14

Software License

Topics:

14.1 Introduction

Some software components are licensed features, which means that the appropriate software licenses must be purchased before the corresponding functionality can be used.

This chapter discusses commercially licensed features, how to check your licenses, and how to install new licenses.

14.2 Licensed Items

For this release, the following HUB software components are licensed:

Licensed functionality	Possible Values	M/O
Routing path AO-AO (application load balancing)	Enabled/Disabled [1]	O
Routing path AT-AT (application concentration)	Enabled/Disabled [1]	O
Routing path AT-AO (bridging)	Enabled/Disabled [1]	O
Routing path AO-AO-AO (AO fallback routing)	Enabled/Disabled [1]	O
Routing path AO-AT (inter-ESME routing)	Enabled/Disabled [1]	O
UCP interface	Enabled/Disabled [2]	O
SMPP interface	Enabled/Disabled [2]	O
CIMD2 interface	Enabled/Disabled [2]	O
Standard HUB statistics (required with STV)	Enabled/Disabled	O
Full HUB statistics and counting rules (required with STV)	Enabled/Disabled	O
Extended application support (up to 10,000 applications)	Enabled/Disabled	O
Session model: Inside only	Enabled/Disabled [3]	O
Session model: Outside only	Enabled/Disabled [3]	O
Session model: Permanent dialout	Enabled/Disabled [3]	O
Session model: Replicate	Enabled/Disabled [3]	O
Session model: Distribute	Enabled/Disabled [3]	O
IP failover control	Enabled/Disabled	O

Notes:

1. At least one routing path must be active (licensed); otherwise, no messages can be routed.
2. At least one application protocol must be active (licensed).
3. At least one session model must be active (licensed).

14.2.1 Multi-Instance License

Multiple instances feature allows you to run multiple HUBs (up to 10 instances) on the same node. Multi-instance license should be enabled for NMM user to run one additional instance of HUB. To run one additional instance of HUB from newly created NMM user (using script `tp_manage_user`), instance license for newly create user id (operating system user identifier) should be enabled in license file.

14.3 License Behaviour

14.3.1 Functional License

All licensed items are subject to a specification in the HUB license key. If a licensed item is not specified in the license, this functionality is not available on the HUB and in the MGR.

14.3.2 Capacity License

The HUB depends on the capacity license of the SMS RTR and does not have a separate capacity license. Refer to the RTR Operator Manual for more information.

14.4 Checking Your License

To view the current license values, execute the following command at the command prompt:

```
tp_system --tp_hub [system]
```

Where `[system]` is the IP address or host name of the HUB.

The following is a sample of the output of the `tp_system` tool.

```
Identification:
    TextPass/HUB R04.02.00
    SunOS 5.10  Generic_127111-01
    Solaris build

Uptime:
    3 days 20h:27m:25s

License key:
    XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX

License information:
    License version 6
    License number TST-94035
    Hardware ID 84a210f2
    Ext Serial No not used
    Zone Serial No not used
    License exceeds in 586 hours
    License exceeds at Tue Mar 23 00:00:00 2010
    License issue number 27
    SMPP access enabled
    UCP access enabled
```

```
    CIMD2 access enabled
    OIS access disabled
    Routing path AO-AO    enabled
    Routing path AT-AT    enabled
    Routing path AT-AO    enabled
    Routing path AO-AO-AO enabled
    Routing path AO-AT    enabled
    Standard statistics          enabled
    Additional statistics        enabled
    Extended Application support enabled
    Inside Only       SessionModel enabled
    Outside Only      SessionModel enabled
    Permanent Dialout SessionModel enabled
    Replicate         SessionModel enabled
    Distribute        SessionModel enabled
    Ip Failover Control enabled

Alarm stations:
    127.0.0.1:11173
```

If multi-instance license is enabled for user id `200` (`'textpass'`) in the license file, then `tp_system` will display the instance user id as shown below:

```
        Instance Serial No 200
```

14.5 Activating a New License

A new license key is required to activate new services or adapt connectivity or performance settings. Please contact your NewNet account MGR to obtain a new license key.

To activate a new license key:

1. Place a valid HUB license file in the `TextPass/etc` directory.
2. At the command prompt, execute the following command (this is usually not service affecting):

   ```
   tp_system --tp_hub --read_licensekey [system]
   ```

   Where `[system]` is the IP address or host name of the HUB.

Note: Also ensure that a valid license is placed and activated on the MGR node as well.

It is not required to restart the HUB when activating a new license; however, you should always restart the MGR if the license affects functionality, to ensure that the correct functionality will appear in the interface.

To verify the new license, execute the following command:

```
tp_system --tp_hub --show_licensekey [system]
```

Where `[system]` is the IP address or host name of the HUB.

If the HUB is started without a valid license, HUB initialisation will fail after a series of errors. After this, a valid license can be activated.

14.6 License Warnings

The following SNMP traps warn when the license limits are approaching:

- `licenseWillExpire` - The license will expire in the given number of hours.
- `licenseExpired` - The validity period of the license has expired.

The traps are generated at these intervals:

- `licenseWillExpire` - Before it expires, 14, 7, 3, and 1 days in advance.
- `licenseExpired` - At the moment it expires and then every hour until fixed.

Note: To adapt the license in a timely manner and avoid expiry, these traps should be properly handled by the Network Management System.

Chapter 15

System Management

Topics:

15.1 Introduction

This chapter describes the command-line tools available to assist in determining the status of the system, as well as for basic troubleshooting.

15.2 Stopping the System

The HUB is designed to run unattended and almost maintenance-free for normal operation. If the HUB process needs to be stopped, execute the following command at the command prompt of the HUB node:

```
tp_stop --tp_hub
```

This command will gracefully shut down the HUB and stop the HUB process and the watchdog process.

Note: `tp_stop` can stop the trace receiver tool automatically, when the `--trace` command is used.

15.3 Starting the System

To start the HUB process, execute the following command at the command prompt of the HUB node:

```
tp_start --tp_hub
```

This command will start the HUB and the watchdog process.

When the HUB node restarts after an unplanned outage (such as a power failure), the HUB process is restarted automatically and resumes service.

`tp_start` can start the trace receiver tool automatically, when the `--trace` command is used.

Note: The HUB will not start if no RTRs are present in the network (or if they are present but not detected by the HUB).

As an alternative to using the `tp_start` tool, the HUB can be rebooted, after which it will automatically restart.

Note: If the HUB operates as part of a cluster, the following cluster-specific names for the command-line tools apply: `hub_start`, `hub_stop`, and `hub_alive`.

15.4 Watchdog Process

The watchdog process and the Mobile Messaging component process communicate via Unix signals.

The watchdog process expects contact from the Mobile Messaging component process every second. If the component process does not contact the watchdog for six seconds, the watchdog stops and restarts the component process.

If a signal is missed, the watchdog writes the following message in the syslog:

```
Missing health signal, missed <number of signals> signals, allowed <max number of
missed signals>
```

If the watchdog stops the component process, it writes the following messages:

```
Missed <number of signals> health signals: trying to cleanly abort process <process
 ID>
Application killed (<process ID>), waiting <number of seconds> seconds before
restarting
```

When the watchdog attempts to restart the component process, it writes the following message:

```
Application restarted, number of unsuccessfully restarts <number of restarts>,
application was running for <number of seconds> seconds
```

If the component process dies or is stopped by the watchdog three times within 30 minutes, the watchdog stops attempting to restart the process and writes the following message:

```
Application terminated, too many restarts within predefined interval
```

To monitor the syslog in RedHat Enterprise Linux, execute:

```
# tail -f /var/log/messages
```

To monitor the syslog in Sun Solaris, execute:

```
# tail -f /var/adm/messages
```

15.5 System Verification

15.5.1 Basic System Verification

For basic verification of the HUB status, execute the following command at the command prompt:

```
tp_system --tp_hub [system]
```

Where `[system]` is a resolvable host name or the IP address of the HUB. The response contains the time that the HUB process has been running. If there is no response, the specified system cannot be reached or the HUB is not running correctly.

For more information about command-line tools, refer to the Tools Operator Manual.

15.5.2 Advanced System Verification

For more detailed information about the HUB system, use the `tp_walk --tp_hub` command-line tool (as user *textpass*) to retrieve information about specific HUB counters. Useful attribute groups for HUB verification are:

- `hubProperties`
- `hubCounters`
- `networkDiscoveryTable`
- `applicationTable`
- `applicationSessionTable`
- `applicationIpAddressTable`
- `serviceCentreTable`
- `scNodeTable`

- `scTerminationPointTable`
- `outsideListenerTable`
- `serviceClassTable`
- `applicationGroupTable`
- `applicationCategoryTable`

15.6 Tracing Application Traffic

The HUB includes a flexible and configurable trace framework that supports the tracing of communication between:

- The HUB and applications
- The HUB and SMSC
- The RTRs

Tracing should only be performed by trained and trusted personnel, as it is an expert-level function that includes traffic communication that is visible.

Tracing will impact system performance.

The trace framework consists of:

- Trace filters (created in the MGR) that allow for the conditional collection of trace data
- The trace receiver (`tp_trace_receiver`), a command-line tool that receives trace data and writes it to a PCAP file
- `tp_filter`, a command-line tool that shows the current trace filter configuration

You can configure up to 10 trace filters, with 100 conditions each.

The trace filter configuration determines the trace points:

- HUB outside application interface (to ESME)
- HUB inside application interface (to SMSC)
- HUB/RTR MXP communication

Note: When the character set conversion functionality is used, trace data may be pre-character conversion or post-character conversion, depending on the trace point specified.

For more information about the trace receiver and its configuration file, refer to the Tools Operator Manual.

15.6.1 Recommended Tracing Method

Tracing should be initiated from a single point (such as the MGR), then started on all HUB and RTR nodes. Trace data can then be collected centrally and viewed in a tool such as Wireshark.

The recommended method to collect and analyse trace information from application traffic is:

1. Run `snoop` on the HUB and collect the relevant data in a trace file.
2. Download the trace file on a PC workstation.
3. Analyse the `snoop` trace using Wireshark.

This method minimizes the impact on HUB performance.

If there are multiple concurrently active trace filters, it is recommended that every individual engineer has a personal instance of the trace receiver, to which the engineer's trace filter can send data.

15.6.2 Presentation of Trace Data in Wireshark

The HUB trace framework rebuilds an Ethernet frame by adding TCP, IP, and Ethernet headers to the message. These headers contain CRC and sequence number, which are not used by the HUB. Therefore, Wireshark may display some warnings alongside the trace data. You can disable these warnings by going to **Edit ➤ Preferences** and change the following protocol settings:

- IP—Disable the **Validate the checksum if possible** setting.
- TCP (used to trace application/SMSC data)—Disable the **Validate the TCP checksum if possible** and **Analyse the TCP sequence numbers if possible** settings.
- SCTP (used to trace MXP data)—Apply the **Checksum Type: None** setting.

15.6.3 Trace Filter Configuration

This section describes how to configure trace filters in the MGR.

15.6.3.1 Creating Trace Filters

To create a trace filter:

1. In the left navigation bar, select **Tracing ➤ Trace Filter**.
   The Trace Filters tab appears.
2. Click **Add New**.
   A new Trace Filters tab appears.
3. Enter a unique name for the trace filter in the **Name** box (maximum 31 characters).
4. Optionally enter a description of the trace filter in the **Description** box.
5. In the **Server IP** box, enter the IP address of the server to which to send the traces from this filter.
6. In the **Server Port** box, enter the UDP port of the server to which to send the traces from this filter.
7. Click **Save**.
   The MGR creates the filter and closes the tab.
8. Configure conditions for the trace filter.
9. After you have created all desired conditions, activate the trace filter.

15.6.3.2 Configuring Trace Filter Conditions

Prerequisites:

- Trace filter

You can configure many conditions for each trace filter. To configure a trace condition:

1. In the left navigation bar, select **Tracing ➤ Trace Filter**.
   The Trace Filters tab appears.

2. Click the trace filter to which you want to add a condition.
   The trace filter opens in a separate Trace Filters tab.
3. Click **Add New** in the lower right corner of the Trace Filters tab.
   The Trace Filter Conditions tab appears.
4. Select a trace filter from the **Trace Filter** list.
5. Select a trace condition that needs to be satisfied for the trace filter to match from the **Condition** list:
   - **Application**—A list of applications appears; select the desired application
   - **Short Number**—A box appears; enter the desired short number of the application (sessions that receive a valid UCP60 with this short number as OAdC will be sent to the trace receiver)
   - **IP Address**—A box appears; enter the desired source IPv4/IPv6 address of the application (all sessions that have packets sent to and received from this IP will be sent to the trace receiver). IP Address value can be IPv4 or IPv6 address.
   - **Unauthenticated sessions**— An outside session has not yet been authenticated
   - **All MXP Traffic**—MXP is a NewNet proprietary protocol used for communication between the HUB, RTR, and AMS. With this condition traffic can be traced between HUB and RTR. This condition is only useful in case the NewNet Technical Assistance Centre (TAC) requested to use this option.
6. Click **Save**.
   The MGR creates the trace filter condition and closes the tab.

Note: You must activate trace filters for them to begin collecting data, but it is not necessary to activate trace filter conditions (they are always active).

15.7 Command-line Tools for Troubleshooting

Tool	Description	More Information
`tp_app_throughput`	Provides the throughput of all applications. **Note:** `tp_app_throughput` is part of the Manager package.	This chapter
`tp_filter`	Allows you to configure the trace filter.	Tools Operator Manual
`tp_session`	Provides session and configuration information for all applications that are connected to the HUB	This chapter
`tp_status`	Provides the operational state and uptime of components.	Tools Operator Manual
`tp_system`	Allows you to: • View software and hardware information;	Tools Operator Manual

Tool	Description	More Information
	• Activate licenses; • Boot the system; • Enable and disable subscriptions to the trap service.	
`tp_manage_user`	Allows you to manages users in a multi-instance setup	Tools Operator Manual
`tp_trace_receiver`	Receives trace data from components and writes it to a PCAP file.	Tools Operator Manual
`tp_walk`	Provides the real-time value of any SNMP attribute.	Tools Operator Manual
`tp_walkall`	Provides the real-time value of all SNMP attributes.	Tools Operator Manual

All commands are scriptable. They exit with value '0' upon successful execution and with value '1' when an error is encountered.

15.7.1 tp_session

The `tp_session` tool:

- Provides session and configuration information for all SMS applications that are connected to the HUB
- Allows manually closing an inside or outside connection to an SMS application

`tp_session` can collect information from an individual HUB device or from all active HUB devices.

Note:

In case of HUB Multi-Instance, before execution of the `tp_session script`, check whether the snmp.conf file is present or not using the following command (this needs to be done for all the configured users).

```
$ ls -ltr /usr/tpuserxx/.snmp/snmp.conf
```

If no such file exists for one or more configured users, then perform the following steps for each of those users.

a. Login as tpuserxx.

b. Create the .snmp directory, if it does not exist

```
$ mkdir -p /usr/tpuserxx/.snmp
```

c. Create the new file snmp.conf as a softlink to the snmp.conf file which comes as part of the Tools package

```
$ ln -s /opt/TextPass/Tools/<verion_number_of_Tools_Package>/var/snmp.conf
/usr/tpuserxx/.snmp/snmp.conf
```

15.7.1.1 Synopsis

```
tp_session [--help] [--device=<host>[:<port>]] [--all-nodes] \
[--detailed] [--app-info] [--noname] [--fullname] [--idx=<index> \
--sel=<sel> | --name=<name> | --close=<id>] [--verbose]
```

15.7.1.2 Options

Option	Description
`--help`	Provides command syntax information.
`--device`	Identifies the HUB device by IP address and port number (only applies when running `tp_session` on a node other than the HUB node). For HUB Multi-Instance setup, it is recommended to provide the HUB External IP address (*see Provisioning HUB External IP*).
`--all-nodes`	Indicates that the session information from all active HUB nodes will be presented. (For HUB Multi-Instance setup, refer point 2 and 3 in the Note mentioned below the table).
`--detailed`	Indicates that `tp_session` will provide the following additional information about the SMS application: • Application name • Service class name • Application group name
`--app-info`	Indicates that `tp_session` will provide the following additional information about the SMS application (only applies to single application queries; the available information depends on the type and service class): • Application index • Application name • Application short number • Application type • Session model • System ID or UID (depends on type) • Maximum number of outside sessions (depends on type) • Outside/inside window size • All service class information
`--noname`	Indicates that IP addresses should not be translated to host names. **Note:** If `--noname` and `--fullname` are not specified, `tp_session` will translate IP addresses to host names (truncated to 18 characters). If both options are specified, `--noname` overrules `--fullname`.

Option	Description
`--fullname`	Indicates that IP addresses should be translated to complete host names (that are not truncated to 18 characters). This option may negatively impact the layout of the output. **Note:** If `--noname` and `--fullname` are not specified, `tp_session` will translate IP addresses to host names (truncated to 18 characters). If both options are specified, `--noname` overrules `--fullname`.
`--idx`	SNMP index of an SMS application. If this parameter is omitted, `tp_session` will present all sessions of all SMS applications.
`--sel`	Defines the field to search for `<name>`: • `short` - Search applicationShortNumber (default) • `name` - Search applicationName • `smpp` - Search applicationSmppSystemId • `cimd` - Search applicationOutsideCimdUserIdentity
`--name`	Value to search for in `<sel>`. If this parameter is omitted, `tp_session` will present all sessions of all SMS applications.
`--close`	Session identifier based on HUB node name and session identifier of the particular session per HUB.
`--verbose`	Indicates that `tp_session` will provide the following additional information about the SMS application: • hubApplicationSessionTimeConnected • hubApplicationSessionMsgReceivedCounter • hubApplicationSessionLastRxActivity • hubApplicationSessionMsgTransmittedCounter • hubApplicationSessionLastTxActivity

Note:

1. To use a `tp_session` command with a special character (such as $), place single quotes (' ') around the command.
2. In case of HUB Multi-Instance setup, ensure that the following configuration is added to the **host-specific configuration** file for the NMM user for whom `tp_session` is being executed.

```
<fxferfile
 localpath="{HOME}/MGRdevices.xml.gz"
 serverpath="/usr/TextPass/etc/MGRdevices.xml.gz"
/>
```

   Here, HOME is the $HOME directory of the NMM user.

3. In case of HUB Multi-Instance setup, either RTR or HUB must be active for the NMM user, for the successful execution of the tp_session script.

15.7.1.3 Examples

The following example runs `tp_session` on a remote node (with the HUB package installed), for the application with short number 202, for all HUB nodes:

```
tp_session --device=10.0.0.20:11261 --all-nodes name=2020
```

The following example runs `tp_session` on the HUB node with detailed information requested for the application with short number 123:

```
tp_session --all-nodes --name=123 --detailed --verbose
```

The following example runs `tp_session` on the HUB node with additional configuration information requested for the application with short number 2020:

```
tp_session --app-info --name=2020
```

The following example shows the output of running `tp_session` (with no options) when the application address type is X.25:

```
sn   ses id               addr_type remote address      local address       type

4444 localhost:00000011   X25       262001              262002              outside
4444 localhost:00000012   IP        cuba.nl.tek*:8001   cuba.nl.tek*:41917  inside
```

The following example shows the output of running `tp_session` (with no options) when the application address type is PSTN:

```
sn   ses id               addr_type remote address      local address       type

5555 localhost:0000001A   PSTN      221                 777778              outside
5555 localhost:0000001B   IP        cuba.nl.tek*:8001   cuba.nl.tek*:43301  inside
```

The following example shows the output of running `tp_session` (with no options) when the application address type is ISDN:

```
sn   ses id               addr_type remote address      local address       type

5555 localhost:0000002B   ISDN      221                 778                 outside
5555 localhost:0000002C   IP        cuba.nl.tek*:8001   cuba.nl.tek*:44426  inside
```

15.7.2 tp_app_throughput

The `tp_app_throughput` command-line tool shows the throughput of different applications in the MGR (total number of messages since the last HUB restart). In continuous mode (`-c` option), each additional report shows the average number of messages per second since the last report.

Note: If, while in continuous mode, data collection takes longer than the configured interval (`-t` option), the results may not be accurate. This condition can occur if the system is slow and/or if there are several thousand applications configured.

15.7.2.1 Synopsis

```
tp_app_throughput [ options ]
```

```
tp_app_throughput -d=<domain> -t=<interval> -a=<applications> -c
```

15.7.2.2 Options

Option	Description
`-t`	Interval time (optional); default is 10 seconds.
`-d`	Domain indication (optional); default is 0 (indicates all domains).
`-a`	Comma-separated list of short codes of applications to monitor (optional); default is “all”.
`-c`	Indicates continuous mode.

`-h`
`-?`
`--help`
	Displays the help message.

15.7.2.3 Operands

Operand	Description
`domain`	Domain to monitor.
`interval`	Interval to monitor.
`applications`	Short code(s) of applications to monitor.

15.8 Operating System Commands for Troubleshooting

15.8.1 Solaris

The following operating system tools are available for troubleshooting on Solaris systems:

Tool	Description
`gcore`	Generates a core file
`hostid`	Provides the host ID of the system (required for a license)
`ifconfig`	Provides an overview of the IP configuration
`netstat`	Provides an overview of IP network statistics
`ping`	An IP diagnostic command
`prstat`	Shows statistics about active processes
`prtconf`	Provides the hardware configuration
`ps`	Provides information about processes

Tool	Description
showrev	Provides the Solaris operating system version
snoop	Trace network traffic on TCP/IP level

Refer to the Solaris documentation at *http://docs.oracle.com*.

15.8.2 Red Hat Enterprise Linux

The following operating system tools are available for troubleshooting on Red Hat Enterprise Linux systems:

Tool	Description
gcore	Generates a core file
hostid	Provides the host ID of the system (required for a license)
ifconfig	Provides an overview of the IP configuration
netstat	Provides an overview of IP network statistics
ping	An IP diagnostic command
top	Shows statistics about active processes
ps	Provides information about processes
sar	Provides performance data
tcpdump	Trace network traffic on TCP/IP level

Refer to the Red Hat Enterprise Linux documentation at *http://www.redhat.com/docs/*.

15.9 Provisioning HUB External IP

The parameter `hubPropIpAddressOwnExternal` and `hubPropIpv6AddressOwnExternal` address will be used as listen addresses for Outside listener(s). Value of these parameters are configurable on the MGR GUI. While adding a new HUB device, the fields **External IPv4 Address** and **External IPv6 Address** appear on the GUI. These IP addresses need to be provisioned on MGR GUI that will be used by the HUB device as its external IP.

The following should be ensured while provisioning the **External IPv4 Address** or **External IPv6 Address** on the MGR GUI:

1. These parameters are mandatory only when using HUB Multi-Instance.
2. If a shared IP address is used between two HUB's (i.e. HUB IP failover) then this field should keep the default value 0.0.0.0.

   Note: This point is valid only for provisioning the"**External IPv4 Address".**
3. When configured, the HUB will accept application connections only on these IP Addresses.
4. The default value for **External IPv4 Address** is '0.0.0.0'.

5. The default value for **External IPv6 Address** is an empty string.
6. Empty string can be configured for both the external addresses. At least one external address must be configured in MGR GUI.

Two HUB devices can be configured on the same server in the same domain if they use different External IPs. However, the HUB devices configured on the same server can be configured using the same external IP addresses if they are part of different domains and the outside listener port is different for them. For more information, refer to the "Configuring Devices" section of the MGR Operator Manual.

Appendix A

Error Mapping and Normalization

Topics:

A.1 Default Error Mapping

This section provides:

- The default forward error mapping from the HUB's internal, normalized errors to SMPP, UCP, and CIMD error codes that are understood by applications and SMSCs
- The default reverse error mapping from the SMPP, UCP, and CIMD error codes generated by applications and SMSCs to the HUB's internal, normalized errors

Use this section to locate the normalized error that corresponds to the SMPP, UCP, or CIMD error that you are interested in. Then, refer to *Error Descriptions* for detailed information about the normalized error.

A.1.1 Default SMPP Forward Error Mapping

Error Code	New Code	Text
intSystemError	8	Internal system error
intShuttingDown	8	Session shutting down
intMxpFailure	8	Error routing message
intMxpTimeout	69	Error routing message
intTxFailure	69	Error sending message
intTemporaryError	100	Temporary error sending message
intPermanentDestError	101	Permanent error sending message
intPermanentMsgError	102	Permanent error sending message
intDestinationNotAvailable	69	Destination not available
intSourceNotAvailable	69	Source not available
intThroughputExceeded	88	Throughput exceeded
intWindowSizeViolation	255	Window size exceeded
intConversionFailed	102	Character set conversion failed
extTransparent	0	-
extResponseTimeout	100	Time out error when sending message
extLoginNoResources	99	Not enough resources
intApplicationNotFound	13	BIND failed, unknown application
intAuthenticationFailed	13	Authentication failure
intApplicationDisabled	13	BIND failed, application disabled
intApplicationGroupDisabled	13	BIND failed
intSubmitMultiPending	69	submit multiple already pending

Error Code	New Code	Text
checksumError	69	Checksum error
syntaxError	69	Syntax error in message
operationNotSupported	3	Operation not supported
operationNotAllowed	101	Operation not allowed
callBarringActive	11	Address blocked
invalidRecipient	11	Invalid recipient
authenticationFailure	14	Invalid login
legitAllCallFailure	101	Operation not allowed
gaNotValid	11	Invalid recipient
repetitionNotAllowed	101	Operation not allowed
legitRepetitionFailure	101	Operation not allowed
priorityCallNotAllowed	101	Operation not allowed
legitCodePrioCallFailure	101	Operation not allowed
urgentMessageNotAllowed	101	Operation not allowed
legitUrgentMessageFailure	101	Operation not allowed
reverseChargingNotAllowed	101	Operation not allowed
legitReverseChargingFailure	101	Operation not allowed
deferredDeliveryNotAllowed	101	Operation not allowed
newAuthCodeNotValid	101	Operation not allowed
newLegitCodeNotValid	101	Operation not allowed
standardTextNotValid	69	Syntax error in message
timePeriodNotValid	97	Invalid time period
messageTypeNotSupported	69	Invalid message type
messageTooLong	69	Invalid message
requestedTextNotValid	69	Syntax error in message
messageTypeNotValidForPager	69	Invalid message type
messageNotFound	12	Message not found
subscriberHangup	11	Recipient error
faxGroupNotSupported	69	Syntax error in message
faxMessageTypeNotSupportted	69	Invalid message type
addressAlreadyInList	10	Invalid address

Error Code	New Code	Text
addressNotInList	10	Invalid address
listFull	101	Operation not allowed
rpidInUse	101	Operation not allowed
deliveryInProgress	100	Delivery in progress
messageForwarded	100	Delivery in progress
invalidMsgLength	1	Message Length is invalid
invalidCommandLength	2	Command Length is invalid
invalidCommandId	3	Invalid Command ID
invalidBindStatusForCmd	4	Incorrect BIND status for given command
alreadyLoggedIn	5	Session already logged in
invalidPriorityFlag	6	Invalid Priority Flag
invalidRegDeliveryFlag	7	Invalid Registered Delivery Flag
systemError	8	System Error
invalidSrcAddress	10	Invalid source address
invalidMsgId	12	Message ID is invalid
bindFailed	13	BIND failed
invalidPassword	14	Invalid password
invalidSystemId	15	Invalid System ID
cancelSmFailed	17	Cancel SM Failed
replaceSmFailed	19	Replace SM Failed
msgQueueFull	20	Message Queue Full
invalidServiceType	21	Invalid Service Type
invalidNumDestinations	51	Invalid number of destinations
invalidDistListName	52	Invalid Distribution List name
invalidDistFlag	64	Destination flag is invalid (submit_multi)
invalidSubmitWithRepRequest	66	Invalid submit with replace request
invalidEsmClassFieldData	67	Invalid esm_class field data
cannotSubmitToDistList	68	Cannot Submit to Distribution List
submitFailed	69	submit_sm, data_sm or submit_multi failed

Error Code	New Code	Text
invalidSrcAddresTon	72	Invalid Source address TON
invalidSrcAddressNpi	73	Invalid Source address NPI
invalidDestAddressTon	80	Invalid Destination address TON
invalidDestAddressNpi	81	Invalid Destination address NPI
invalidSystemType	83	Invalid system_type field
invalidRepIfPresentFlag	84	Invalid replace_if_present flag
invalidNumberOfMessages	85	Invalid number of messages
throttlingError	88	Throttling error
invalidDeliveryTime	97	Invalid Scheduled Delivery Time
invalidValidityPeriod	98	Invalid message validity period
invalidPredefMessage	99	Predefined Message ID is Invalid
receiverTempAppError	100	ESME Receiver Temporary App Error Code
receiverPermAppError	101	ESME Receiver Permanent App Error Code
receiverRejectMessage	102	ESME Receiver Reject Message Error Code
querySmFailed	103	query_sm request failed
errorInOptionalPduPart	192	Error in the optional part of the PDU Body
optionalParamNotAllowed	193	TLV not allowed
invalidParamLength	194	Invalid Parameter Length
expectedOptParamMissing	195	Expected TLV missing
invalidOptParamValue	196	Invalid TLV Value
deliveryFailure	254	Transaction Delivery Failure
unknownError	255	Unknown error
serviceTypeUnauthorized	256	ESME Not authorised to use specified service_type
requestProhibited	257	ESME Prohibited from using specified operation
serviceTypeUnavailable	258	Specified service_type is unavailable
serviceTypeDenied	259	Specified service_type is denied
invalidCodingScheme	260	Invalid Data Coding Scheme

Error Code	New Code	Text
invalidSourceSubAddressUnit	261	Source Address Sub unit is Invalid
invalidDestinationSubAddressUnit	262	Destination Address Sub unit is Invalid
invalidBroadcastFrequencyInterval	263	Broadcast Frequency Interval is invalid
invalidBroadcastAliasName	264	Broadcast Alias Name is invalid
invalidBroadcastAreaFormat	265	Broadcast Area Format is invalid
invalidNumBroadcastAreas	266	Number of Broadcast Areas is invalid
invalidBroadcastContentType	267	Broadcast Content Type is invalid
invalidBroadcastMessageClass	268	Broadcast Message Class is invalid
broadcastFailed	269	broadcast_sm operation failed
queryBroadcastFailed	270	query_broadcast_sm operation failed
cancelBroadcastFailed	271	cancel_broadcast_sm operation failed
invalidNumBroadcastRepetition	272	Number of Repeated Broadcasts is invalid
invalidBroadcastServiceGroup	273	Broadcast Service Group is invalid
invalidBroacastChannelIndicator	274	Broadcast Channel Indicator is invalid
noSystemIdPresent	15	No system ID present
invalidAddressRange	13	Invalid address range
unexpectedOperation	4	Incorrect BIND status for given command
unsupportedParameter	3	Operation not supported
smscConnectionLost	13	No connection to Service Centre
noSmscResponse	100	Time out error when sending message
generalSystemError	8	Internal system error
cannotFindInformation	12	Message not found
parameterFormatingError	102	Permanent error sending message
requestedOperationFailed	101	Operation not allowed
tempCongestionError	100	Temporary error sending message
invalidLogin	14	Invalid login
incorrectAccessType	101	Operation not allowed
tooManyUsersOfSameId	13	Too many sessions for this login
loginRefused	13	Login not allowed
invalidWindowSize	101	Operation not allowed

Error Code	New Code	Text
windowingDisabled	101	Operation not allowed
virtualSmscBarring	101	Operation not allowed
invalidSubAddress	101	Operation not allowed
aliasAccountLoginRefused	101	Operation not allowed
incorrectNumDestAddresses	11	Invalid recipient
syntaxErrorInUserDataParam	69	Syntax error in message
incorrectUdhParameterComb	69	Invalid message type
incorrectDcsParamUsage	69	Syntax error in message
incorrectVPParam	97	Invalid time period
incorrectPID	69	Syntax error in message
incorrectFirstDelivery	97	Invalid time period
incorrectReplyPath	69	Syntax error in message
incorrectStatusReportReq	69	Syntax error in message
incorrectCancelEnabled	69	Syntax error in message
incorrectPriorityParam	69	Syntax error in message
incorrectTariffClassParam	69	Syntax error in message
incorrectServiceDescriptionParam	69	Syntax error in message
incorrectTransportTypeParam	69	Syntax error in message
incorrectMessageTypeParam	69	Invalid message type
incorrectMoreMessagesToSendParam	69	Syntax error in message
incorrectOperationTimerParam	69	Syntax error in message
incorrectDialogueIdParam	69	Syntax error in message
incorrectAlphanumericOrigAddress	69	Syntax error in message
incorrectAlphanumericAddressData	10	Invalid address
incorrectAddressInquiryParam	10	Invalid address
incorrectSctsInquiryParam	97	Invalid time period
incorrectSctsDeliverParam	97	Invalid time period
incorrectModeDeliverParam	69	Syntax error in message
incorrectDeliveryParamComb	69	Syntax error in message
incorrectSctsForCancel	97	Invalid time period
incorrectAddressForCancel	10	Invalid address

Error Code	New Code	Text
incorrectModeForCancel	69	Syntax error in message
incorrectParamCombforCancel	69	Syntax error in message
setChangingPasswordFailed	69	Syntax error in message
setChangingPasswordNotAllowed	69	Syntax error in message
getUnsupportedParam	3	Operation not supported
mtTimeout	100	Time out error when delivering message
mtAbsentSubscriber	100	Time out error when delivering message
mtNoPagingError	100	Time out error when delivering message
mtImsiDetach	100	Time out error when delivering message
mtRoamingRestrictions	100	Time out error when delivering message
mtSystemFailure	100	Time out error when delivering message
mtDataMissing	102	Permanent error sending message
mtUnexpectedDataValue	102	Permanent error sending message
mtFacilityNotSupported	102	Permanent error sending message
mtUnidentifiedSubscriber	102	Permanent error sending message
mtIllegalSubscriber	102	Permanent error sending message
mtIllegalEquipment	102	Permanent error sending message
mtSubscriberBusyForMt	100	Subscriber busy
mtInvalidSmeAddress	102	Permanent error sending message
mtEquipmentProtocolError	102	Permanent error sending message
mtEquipmentNotSmEquipped	102	Permanent error sending message
mtMemoryCapacityExceeded	100	Time out error when delivering message
mtOtherMapError	100	Time out error when delivering message
mtTcapAborted	100	Time out error when delivering message
mtSccpAborted	100	Time out error when delivering message

Error Code	New Code	Text
mtBlockedByMtRule	5	Not accepted - Recipient address is in blacklist
mtShortMsgType0NotSupported	102	Permanent error sending message
mtCanNotReplaceShortMsg	102	Permanent error sending message
mtUnspecifiedProtocolId	102	Permanent error sending message
mtMsgClassNotSupported	102	Permanent error sending message
mtUnspecifiedDataCodingScheme	102	Permanent error sending message
mtTpduNotSupported	102	Permanent error sending message
mtSimStorageFull	100	Temporary error when delivering message
mtNoSmStorageCapabilityInSim	102	Permanent error sending message
mtErrorinMs	102	Permanent error sending message
mtSimApplToolKitBusy	102	Permanent error sending message
mtSimDataDownloadError	102	Permanent error sending message
mtApplSpecificError	102	Permanent error sending message
mtEquipUnspecifiedErrorCause	102	Permanent error sending message
mtUeDeDeregistered	100	Time out error when delivering message
mtNoResponseViaIpsmgw	100	Time out error when delivering message
sriSmTimeout	100	Time out error when delivering message
sriSmSystemFailure	100	Time out error when delivering message
sriSmDataMissing	102	Permanent error sending message
sriSmUnexpectedDataValue	102	Permanent error sending message
sriSmFacilityNotSupported	102	Permanent error sending message
sriSmUnknownSubscriber	102	Permanent error sending message
sriSmAbsentSubscriber	100	Absent subscriber
sriSmCallBarred	102	Permanent error sending message
sriSmTeleserviceNotProvisioned	102	Permanent error sending message
sriSmOtherMapError	102	Permanent error sending message
sriSmTcapAborted	102	Permanent error sending message

Error Code	New Code	Text
sriSmSccpAborted	102	Permanent error sending message
sriSmMsDeregistered	100	Absent subscriber
sriSmMsPurged	100	Absent subscriber
atSystemError	8	Internal system error
atApplicationNotAvailable	69	Destination not available
atBlockedByThroughputControl	88	Throughput exceeded
atBlockedByAtRule	11	Address blocked
atApplicationNotExist	11	Invalid recipient
atTxWindowFull	255	Window size exceeded
amsDeviceNotActive	100	SMSC not available
amsDbError	100	SMSC storage failure
amsStoreFull	20	Message Queue Full
amsQueueFull	20	Message Queue Full
amsRecipientBufferFull	20	Message Queue Full
amsInvalidQueue	102	Permanent error sending message
amsInvalidMessage	102	Permanent error sending message
amsInvalidValidityTime	98	Invalid message validity period
amsInvalidDeferredDelivery	97	Invalid Scheduled Delivery Time
amsStorageRateExceeded	100	Storage rate exceeded
amsDeliveryAttemptsExceeded	102	Permanent error sending message
amsCapabilityNotEnabled	4	Permanent error sending message
rtrBlockedByThroughputControl	88	Throughput exceeded
rtrStorageFailure	100	SMSC storage failure
rtrTimeout	100	SMSC not available
rtrBlockedByRule	11	Address blocked
rtrNoRuleMatching	69	Submit_sm or data_sm failed
rtrLicenseExceeded	100	Temporary error sending message
rtrOriginatorInBlackList	10	Address blocked - Address is in blacklist
rtrOriginatorNotInWhiteList	10	Address blocked - Address not in white list
rtrMessageTooLong	69	Syntax error in message

Error Code	New Code	Text
rtrInternalDecodingFailure	102	Permanent error sending message
rtrInvalidSourceApp	102	Permanent error sending message
rtrInvalidOriginator	102	Permanent error sending message
rtrInvalidRecipient	102	Permanent error sending message
rtrMessageSegmentTooLong	102	Permanent error sending message
rtrMessageSarUdhTooLong	102	Permanent error sending message
capScfUnavailable	8	Internal system error
capUnassignedNumber	102	Permanent error sending message
capUnidentifiedSubscriber	102	Permanent error sending message
capCongestion	100	Temporary error sending message
capFacilityNotSupported	102	Permanent error sending message
capTransferRejected	101	Operation not allowed

A.1.2 Default UCP Forward Error Mapping

Error Code	New Code	Text
intSystemError	4	Internal system error
intShuttingDown	4	Session shutting down
intMxpFailure	3	Error routing message
intMxpTimeout	3	Error routing message
intTxFailure	3	Error sending message
intTemporaryError	30	Temporary error sending message
intPermanentDestError	30	Permanent error sending to destination
intPermanentMsgError	30	Permanent error sending message
intDestinationNotAvailable	30	Destination not available
intSourceNotAvailable	30	Source not available
intThroughputExceeded	4	Throughput exceeded
intWindowSizeViolation	4	Window size exceeded
intConversionFailed	4	Character set conversion failed
extTransparent	0	-
extResponseTimeout	3	Time out error when sending message
extLoginNoResources	99	Not enough resources

Error Code	New Code	Text
intApplicationNotFound	7	Application not found
intAuthenticationFailed	7	Authentication failure
intApplicationDisabled	7	Application disabled
intApplicationGroupDisabled	7	Application group disabled
intSubmitMultiPending	4	submit multiple already pending
checksumError	1	Checksum error
syntaxError	2	Syntax error in message
operationNotSupported	3	Operation not supported
operationNotAllowed	4	Operation not allowed
callBarringActive	5	Call barring active
invalidRecipient	6	Invalid recipient
authenticationFailure	7	Authentication failure
legitAllCallFailure	8	Legitimisation code for all calls, failure
gaNotValid	9	GA not valid
repetitionNotAllowed	10	Repetition not allowed
legitRepetitionFailure	11	Legitimisation code for all calls, failure
priorityCallNotAllowed	12	Priority call not allowed
legitCodePrioCallFailure	13	Legitimisation code for priority call, failure
urgentMessageNotAllowed	14	Urgent message not allowed
legitUrgentMessageFailure	15	Legitimisation code for urgent message, failure
reverseChargingNotAllowed	16	Reverse charging not allowed
legitReverseChargingFailure	17	Legitimisation code for rev. charging, failure
deferredDeliveryNotAllowed	18	Deferred delivery not allowed
newAuthCodeNotValid	19	New AC not valid
newLegitCodeNotValid	20	New legitimisation code not valid
standardTextNotValid	21	Standard text not valid
timePeriodNotValid	22	Time period not valid
messageTypeNotSupported	23	Message type not supported by system
messageTooLong	24	Message too long

Error Code	New Code	Text
requestedTextNotValid	25	Requested standard text not valid
messageTypeNotValidForPager	26	Message type not valid for the pager type
messageNotFound	27	Message not found in smsc
subscriberHangup	30	Subscriber hang-up
faxGroupNotSupported	31	Fax group not supported
faxMessageTypeNotSupportted	32	Fax message type not supported
addressAlreadyInList	33	Address already in list (60 series)
addressNotInList	34	Address not in list (60 series)
listFull	35	List full, cannot add address to list (60 series)
rpidInUse	36	RPID already in use
deliveryInProgress	37	Delivery in progress
messageForwarded	38	Message forwarded
invalidMsgLength	2	Syntax error in message
invalidCommandLength	2	Syntax error in message
invalidCommandId	2	Syntax error in message
invalidBindStatusForCmd	4	Invalid BIND status for operation
alreadyLoggedIn	7	Session already logged in
invalidPriorityFlag	2	Syntax error in message
invalidRegDeliveryFlag	2	Syntax error in message
systemError	4	Internal system error
invalidSrcAddress	33	Invalid originator address
invalidMsgId	24	Invalid message
bindFailed	7	Login not allowed
invalidPassword	7	Invalid password
invalidSystemId	7	Invalid system id
cancelSmFailed	4	Operation not allowed
replaceSmFailed	4	Operation not allowed
msgQueueFull	4	Not accepted - Maximum messages for the address exceeded
invalidServiceType	4	Invalid service type

Error Code	New Code	Text
invalidNumDestinations	2	Invalid num destinations
invalidDistListName	2	Invalid distribution list name
invalidDistFlag	2	Invalid destination flag
invalidSubmitWithRepRequest	4	Operation not allowed
invalidEsmClassFieldData	2	Syntax error in message
cannotSubmitToDistList	4	Operation not allowed
submitFailed	4	Operation not allowed
invalidSrcAddresTon	33	Invalid originator TON
invalidSrcAddressNpi	33	Invalid originator NPI
invalidDestAddressTon	33	Invalid recipient TON
invalidDestAddressNpi	33	Invalid recipient NPI
invalidSystemType	2	Invalid field in request
invalidRepIfPresentFlag	2	Invalid field in request
invalidNumberOfMessages	2	Invalid field in request
throttlingError	4	Throughput exceeded
invalidDeliveryTime	2	Invalid delivery time
invalidValidityPeriod	2	Invalid validity period
invalidPredefMessage	2	Invalid message
receiverTempAppError	30	Temporary error sending message
receiverPermAppError	4	Permanent error sending message
receiverRejectMessage	4	Permanent error sending message
querySmFailed	4	Inquiry operation failed
errorInOptionalPduPart	2	Syntax error in message
optionalParamNotAllowed	2	Syntax error in message
invalidParamLength	2	Syntax error in message
expectedOptParamMissing	2	Syntax error in message
invalidOptParamValue	2	Syntax error in message
deliveryFailure	4	Delivery failure
unknownError	0	Unknown error
serviceTypeUnauthorized	4	Operation not allowed
requestProhibited	4	Operation not allowed

Error Code	New Code	Text
serviceTypeUnavailable	4	Operation not allowed
serviceTypeDenied	4	Operation not allowed
invalidCodingScheme	2	Syntax error in message
invalidSourceSubAddressUnit	2	Syntax error in message
invalidDestinationSubAddressUnit	2	Syntax error in message
invalidBroadcastFrequencyInterval	2	Syntax error in message
invalidBroadcastAliasName	2	Syntax error in message
invalidBroadcastAreaFormat	2	Syntax error in message
invalidNumBroadcastAreas	2	Syntax error in message
invalidBroadcastContentType	2	Syntax error in message
invalidBroadcastMessageClass	2	Syntax error in message
broadcastFailed	4	Operation failed
queryBroadcastFailed	4	Operation failed
cancelBroadcastFailed	4	Operation failed
invalidNumBroadcastRepetition	2	Syntax error in message
invalidBroadcastServiceGroup	2	Syntax error in message
invalidBroacastChannelIndicator	2	Syntax error in message
noSystemIdPresent	2	Syntax error in message
invalidAddressRange	2	Invalid address range
unexpectedOperation	4	Invalid BIND status for operation
unsupportedParameter	3	Operation not supported
smscConnectionLost	4	No connection to Service Centre
noSmscResponse	3	Time out error when sending message
generalSystemError	4	Internal system error
cannotFindInformation	27	Message not found
parameterFormatingError	4	Permanent error sending message
requestedOperationFailed	4	Operation not allowed
tempCongestionError	30	Temporary error sending message
invalidLogin	7	Invalid login
incorrectAccessType	4	Operation not allowed
tooManyUsersOfSameId	4	Too many sessions for this login

Error Code	New Code	Text
loginRefused	7	Login not allowed
invalidWindowSize	4	Operation not allowed
windowingDisabled	4	Operation not allowed
virtualSmscBarring	4	Operation not allowed
invalidSubAddress	4	Operation not allowed
aliasAccountLoginRefused	4	Operation not allowed
incorrectNumDestAddresses	6	Invalid recipient
syntaxErrorInUserDataParam	2	Syntax error in message
incorrectUdhParameterComb	23	Invalid message type
incorrectDcsParamUsage	2	Syntax error in message
incorrectVPParam	22	Invalid time period
incorrectPID	2	Syntax error in message
incorrectFirstDelivery	22	Invalid time period
incorrectReplyPath	2	Syntax error in message
incorrectStatusReportReq	2	Syntax error in message
incorrectCancelEnabled	2	Syntax error in message
incorrectPriorityParam	2	Syntax error in message
incorrectTariffClassParam	2	Syntax error in message
incorrectServiceDescriptionParam	2	Syntax error in message
incorrectTransportTypeParam	2	Syntax error in message
incorrectMessageTypeParam	23	Invalid message type
incorrectMoreMessagesToSendParam	2	Syntax error in message
incorrectOperationTimerParam	2	Syntax error in message
incorrectDialogueIdParam	2	Syntax error in message
incorrectAlphanumericOrigAddress	2	Syntax error in message
incorrectAlphanumericAddressData	33	Invalid address
incorrectAddressInquiryParam	33	Invalid address
incorrectSctsInquiryParam	22	Invalid time period
incorrectSctsDeliverParam	22	Invalid time period
incorrectModeDeliverParam	33	Invalid address
incorrectDeliveryParamComb	33	Invalid address

Error Code	New Code	Text
incorrectSctsForCancel	22	Invalid time period
incorrectAddressForCancel	33	Invalid address
incorrectModeForCancel	2	Syntax error in message
incorrectParamCombforCancel	2	Syntax error in message
setChangingPasswordFailed	2	Syntax error in message
setChangingPasswordNotAllowed	2	Syntax error in message
getUnsupportedParam	3	Operation not supported
mtTimeout	4	Error routing message
mtAbsentSubscriber	30	Temporary error sending message
mtNoPagingError	30	Temporary error sending message
mtImsiDetach	30	Temporary error sending message
mtRoamingRestrictions	30	Temporary error sending message
mtSystemFailure	30	Temporary error sending message
mtDataMissing	4	Permanent error sending message
mtUnexpectedDataValue	4	Permanent error sending message
mtFacilityNotSupported	4	Permanent error sending message
mtUnidentifiedSubscriber	4	Permanent error sending message
mtIllegalSubscriber	4	Permanent error sending message
mtIllegalEquipment	4	Permanent error sending message
mtSubscriberBusyForMt	30	Temporary error sending message
mtInvalidSmeAddress	4	Permanent error sending message
mtEquipmentProtocolError	4	Permanent error sending message
mtEquipmentNotSmEquipped	4	Permanent error sending message
mtMemoryCapacityExceeded	30	Temporary error sending message
mtOtherMapError	30	Temporary error sending message
mtTcapAborted	30	Temporary error sending message
mtSccpAborted	30	Temporary error sending message
mtBlockedByMtRule	5	Not accepted - Recipient address is in blacklist
mtShortMsgType0NotSupported	4	Permanent error sending message
mtCanNotReplaceShortMsg	4	Permanent error sending message
mtUnspecifiedProtocolId	4	Permanent error sending message

Error Code	New Code	Text
mtMsgClassNotSupported	4	Permanent error sending message
mtUnspecifiedDataCodingScheme	4	Permanent error sending message
mtTpduNotSupported	4	Permanent error sending message
mtSimStorageFull	30	Temporary error sending message
mtNoSmStorageCapabilityInSim	4	Permanent error sending message
mtErrorinMs	4	Permanent error sending message
mtSimApplToolKitBusy	4	Permanent error sending message
mtSimDataDownloadError	4	Permanent error sending message
mtApplSpecificError	4	Permanent error sending message
mtEquipUnspecifiedErrorCause	4	Permanent error sending message
mtUeDeDeregistered	30	Temporary error sending message
mtNoResponseViaIpsmgw	30	Temporary error sending message
sriSmTimeout	4	Error routing message
sriSmSystemFailure	30	Temporary error sending message
sriSmDataMissing	4	Permanent error sending message
sriSmUnexpectedDataValue	4	Permanent error sending message
sriSmFacilityNotSupported	4	Permanent error sending message
sriSmUnknownSubscriber	4	Permanent error sending message
sriSmAbsentSubscriber	30	Temporary error sending message
sriSmCallBarred	4	Permanent error sending message
sriSmTeleserviceNotProvisioned	4	Permanent error sending message
sriSmOtherMapError	4	Permanent error sending message
sriSmTcapAborted	4	Permanent error sending message
sriSmSccpAborted	4	Permanent error sending message
sriSmMsDeregistered	30	Temporary error sending message
sriSmMsPurged	30	Temporary error sending message
atSystemError	4	Internal system error
atApplicationNotAvailable	30	Destination not available
atBlockedByThroughputControl	4	Throughput exceeded
atBlockedByAtRule	5	Not accepted - Recipient address is in blacklist
atApplicationNotExist	6	Invalid recipient

Error Code	New Code	Text
atTxWindowFull	4	Window size exceeded
amsDeviceNotActive	4	Permanent error sending message
amsDbError	4	Permanent error sending message
amsStoreFull	4	Not accepted - Maximum messages for the address exceeded
amsQueueFull	4	Not accepted - Maximum messages for the address exceeded
amsRecipientBufferFull	4	Not accepted - Maximum messages for the address exceeded
amsInvalidQueue	4	Permanent error sending message
amsInvalidMessage	4	Permanent error sending message
amsInvalidValidityTime	22	Invalid time period
amsInvalidDeferredDelivery	22	Invalid time period
amsStorageRateExceeded	30	Temporary error sending message
amsDeliveryAttemptsExceeded	4	Permanent error sending message
amsCapabilityNotEnabled	4	Permanent error sending message
rtrBlockedByThroughputControl	4	Throughput exceeded
rtrStorageFailure	30	Temporary error sending message
rtrTimeout	3	Time out error when sending message
rtrBlockedByRule	5	Not accepted - Recipient address is in blacklist
rtrNoRuleMatching	4	Operation not allowed
rtrLicenseExceeded	30	Temporary error sending message
rtrOriginatorInBlackList	5	Not accepted - Recipient address is in blacklist
rtrOriginatorNotInWhiteList	5	Not accepted - Recipient address is in blacklist
rtrMessageTooLong	2	Syntax error in message
rtrInternalDecodingFailure	4	Permanent error sending message
rtrInvalidSourceApp	4	Permanent error sending message
rtrInvalidOriginator	4	Permanent error sending message
rtrInvalidRecipient	4	Permanent error sending message
rtrMessageSegmentTooLong	4	Permanent error sending message

Error Code	New Code	Text
rtrMessageSarUdhTooLong	4	Permanent error sending message
capScfUnavailable	4	Internal system error
capUnassignedNumber	4	Permanent error sending message
capUnidentifiedSubscriber	4	Permanent error sending message
capCongestion	30	Temporary error sending message
capFacilityNotSupported	4	Permanent error sending message
capTransferRejected	4	Operation not allowed

A.1.3 Default CIMD Forward Error Mapping

Error Code	New Code	Text
intSystemError	6	General system error
intShuttingDown	4	Connection to SMS Center lost
intMxpFailure	9	Error routing message
intMxpTimeout	9	Error routing message
intTxFailure	9	Error sending message
intTemporaryError	10	Temporary congestion error
intPermanentDestError	9	Permanent error sending to destination
intPermanentMsgError	9	Permanent error sending message
intDestinationNotAvailable	9	Destination not available
intSourceNotAvailable	4	Source not available
intThroughputExceeded	10	Throughput exceeded
intWindowSizeViolation	10	Window size exceeded
intConversionFailed	9	Character set conversion failed
extTransparent	0	-
extResponseTimeout	10	Time out error when sending message
extLoginNoResources	103	Not enough resources
intApplicationNotFound	100	Application not found
intAuthenticationFailed	100	Authentication failure
intApplicationDisabled	100	Application disabled
intApplicationGroupDisabled	100	Application group disabled
intSubmitMultiPending	9	submit multiple already pending

Error Code	New Code	Text
checksumError	8	Checksum error
syntaxError	2	Syntax error in message
operationNotSupported	3	Unsupported parameter error
operationNotAllowed	9	Operation not allowed
callBarringActive	9	Not accepted - Recipient address is in blacklist
invalidRecipient	300	Incorrect destination address
authenticationFailure	100	Invalid login
legitAllCallFailure	9	Operation not allowed
gaNotValid	300	Incorrect destination address
repetitionNotAllowed	9	Operation not allowed
legitRepetitionFailure	9	Operation not allowed
priorityCallNotAllowed	9	Operation not allowed
legitCodePrioCallFailure	9	Operation not allowed
urgentMessageNotAllowed	9	Operation not allowed
legitUrgentMessageFailure	9	Operation not allowed
reverseChargingNotAllowed	9	Operation not allowed
legitReverseChargingFailure	9	Operation not allowed
deferredDeliveryNotAllowed	9	Operation not allowed
newAuthCodeNotValid	9	Operation not allowed
newLegitCodeNotValid	9	Operation not allowed
standardTextNotValid	2	Syntax error in message
timePeriodNotValid	305	Incorrect validity/firstdel parameter usage
messageTypeNotSupported	316	Incorrect message type parameter usage
messageTooLong	303	Incorrect bin/head/normal user data parameter combination
requestedTextNotValid	2	Syntax error in message
messageTypeNotValidForPager	316	Incorrect message type parameter usage
messageNotFound	7	Cannot find information
subscriberHangup	300	Incorrect destination address
faxGroupNotSupported	2	Syntax error in message

Error Code	New Code	Text
faxMessageTypeNotSupportted	316	Incorrect message type parameter usage
addressAlreadyInList	306	Incorrect address usage
addressNotInList	306	Incorrect address usage
listFull	9	Operation not allowed
rpidInUse	9	Operation not allowed
deliveryInProgress	9	Delivery in progress
messageForwarded	9	Delivery in progress
invalidMsgLength	2	Syntax error in message
invalidCommandLength	2	Syntax error in message
invalidCommandId	2	Syntax error in message
invalidBindStatusForCmd	1	Unexpected operation
alreadyLoggedIn	9	Session already logged in
invalidPriorityFlag	2	Syntax error in message
invalidRegDeliveryFlag	2	Syntax error in message
systemError	6	General system error
invalidSrcAddress	306	Incorrect address usage
invalidMsgId	303	Incorrect bin/head/normal user data parameter combination
bindFailed	103	Login refused by SMS Center
invalidPassword	100	Invalid login
invalidSystemId	100	Invalid login
cancelSmFailed	9	Operation not allowed
replaceSmFailed	9	Operation not allowed
msgQueueFull	10	Not accepted - Maximum messages for the address exceeded
invalidServiceType	9	Operation not allowed
invalidNumDestinations	2	Syntax error in message
invalidDistListName	2	Syntax error in message
invalidDistFlag	2	Syntax error in message
invalidSubmitWithRepRequest	9	Operation not allowed
invalidEsmClassFieldData	2	Syntax error in message
cannotSubmitToDistList	9	Operation not allowed

Error Code	New Code	Text
submitFailed	9	Operation not allowed
invalidSrcAddresTon	306	Incorrect address usage
invalidSrcAddressNpi	306	Incorrect address usage
invalidDestAddressTon	306	Incorrect address usage
invalidDestAddressNpi	306	Incorrect address usage
invalidSystemType	2	Syntax error in message
invalidRepIfPresentFlag	2	Syntax error in message
invalidNumberOfMessages	2	Syntax error in message
throttlingError	10	Throughput exceeded
invalidDeliveryTime	2	Invalid delivery time
invalidValidityPeriod	2	Invalid validity period
invalidPredefMessage	2	Invalid message
receiverTempAppError	10	Temporary congestion error
receiverPermAppError	9	Permanent error sending message
receiverRejectMessage	9	Permanent error sending message
querySmFailed	9	Enquire message status failed
errorInOptionalPduPart	2	Syntax error in message
optionalParamNotAllowed	2	Syntax error in message
invalidParamLength	2	Syntax error in message
expectedOptParamMissing	2	Syntax error in message
invalidOptParamValue	2	Syntax error in message
deliveryFailure	9	Delivery failure
unknownError	0	Unknown error
serviceTypeUnauthorized	9	Operation not allowed
requestProhibited	9	Operation not allowed
serviceTypeUnavailable	9	Operation not allowed
serviceTypeDenied	9	Operation not allowed
invalidCodingScheme	2	Syntax error in message
invalidSourceSubAddressUnit	2	Syntax error in message
invalidDestinationSubAddressUnit	2	Syntax error in message
invalidBroadcastFrequencyInterval	2	Syntax error in message

Error Code	New Code	Text
invalidBroadcastAliasName	2	Syntax error in message
invalidBroadcastAreaFormat	2	Syntax error in message
invalidNumBroadcastAreas	2	Syntax error in message
invalidBroadcastContentType	2	Syntax error in message
invalidBroadcastMessageClass	2	Syntax error in message
broadcastFailed	9	Operation failed
queryBroadcastFailed	9	Operation failed
cancelBroadcastFailed	9	Operation failed
invalidNumBroadcastRepetition	2	Syntax error in message
invalidBroadcastServiceGroup	2	Syntax error in message
invalidBroacastChannelIndicator	2	Syntax error in message
noSystemIdPresent	2	Syntax error in message
invalidAddressRange	2	Invalid address range
unexpectedOperation	1	Unexpected operation
unsupportedParameter	3	Unsupported parameter
smscConnectionLost	4	Source not available
noSmscResponse	5	No response from SMSC
generalSystemError	6	General system error
cannotFindInformation	7	Cannot find information
parameterFormatingError	8	Parameter formatting error
requestedOperationFailed	9	Requested operation failed
tempCongestionError	10	Temporary congestion error
invalidLogin	100	Invalid login
incorrectAccessType	101	Incorrect access type
tooManyUsersOfSameId	102	Too many users with this login ID
loginRefused	103	Login refused by SMS Center
invalidWindowSize	104	Invalid window size
windowingDisabled	105	Windowing disabled
virtualSmscBarring	106	Virtual SMS Center-based barring
invalidSubAddress	107	Invalid subaddr
aliasAccountLoginRefused	108	Alias account, login refused

Error Code	New Code	Text
incorrectNumDestAddresses	301	Incorrect number of destination addresses
syntaxErrorInUserDataParam	302	syntax error in user data parameter
incorrectUdhParameterComb	303	Incorrect bin/head/normal user data parameter combination
incorrectDcsParamUsage	304	Incorrect DCS parameter usage
incorrectVPParam	305	Incorrect validity parameter usage
incorrectPID	307	Incorrect PID parameter usage
incorrectFirstDelivery	308	Incorrect first delivery parameter usage
incorrectReplyPath	309	Incorrect reply path usage
incorrectStatusReportReq	310	Incorrect status report request parameter usage
incorrectCancelEnabled	311	Incorrect cancel enabled parameter usage
incorrectPriorityParam	312	Incorrect priority parameter usage
incorrectTariffClassParam	313	Incorrect tariff class parameter usage
incorrectServiceDescriptionParam	314	Incorrect service description parameter usage
incorrectTransportTypeParam	315	Incorrect transport type parameter usage
incorrectMessageTypeParam	316	Incorrect message type parameter usage
incorrectMoreMessagesToSendParam	318	Incorrect MMs parameter usage
incorrectOperationTimerParam	319	Incorrect operation timer parameter usage
incorrectDialogueIdParam	320	Incorrect dialogue ID parameter usage
incorrectAlphanumericOrigAddress	321	Incorrect alpha originator address usage
incorrectAlphanumericAddressData	322	Incorrect data for alpha numeric originator
incorrectAddressInquiryParam	400	Incorrect address parameter usage
incorrectSctsInquiryParam	401	Incorrect scts parameter usage
incorrectSctsDeliverParam	500	Incorrect scts parameter usage
incorrectModeDeliverParam	501	Incorrect mode parameter usage
incorrectDeliveryParamComb	502	Incorrect parameter combination
incorrectSctsForCancel	600	Incorrect scts parameter usage

Error Code	New Code	Text
incorrectAddressForCancel	601	Incorrect address parameter usage
incorrectModeForCancel	602	Incorrect mode parameter usage
incorrectParamCombforCancel	603	Incorrect parameter combination
setChangingPasswordFailed	800	Changing password failed
setChangingPasswordNotAllowed	801	Changing password not allowed
getUnsupportedParam	900	Unsupported item requested
mtTimeout	10	Temporary congestion error
mtAbsentSubscriber	10	Temporary congestion error
mtNoPagingError	10	Temporary congestion error
mtImsiDetach	10	Temporary congestion error
mtRoamingRestrictions	10	Temporary congestion error
mtSystemFailure	10	Temporary congestion error
mtDataMissing	9	Permanent error sending message
mtUnexpectedDataValue	9	Permanent error sending message
mtFacilityNotSupported	9	Permanent error sending message
mtUnidentifiedSubscriber	9	Permanent error sending message
mtIllegalSubscriber	9	Permanent error sending message
mtIllegalEquipment	9	Permanent error sending message
mtSubscriberBusyForMt	10	Temporary congestion error
mtInvalidSmeAddress	9	Permanent error sending message
mtEquipmentProtocolError	9	Permanent error sending message
mtEquipmentNotSmEquipped	9	Permanent error sending message
mtMemoryCapacityExceeded	10	Temporary congestion error
mtOtherMapError	10	Temporary congestion error
mtTcapAborted	10	Temporary congestion error
mtSccpAborted	10	Temporary congestion error
mtBlockedByMtRule	9	Not accepted - Recipient address is in blacklist
mtShortMsgType0NotSupported	9	Permanent error sending message
mtCanNotReplaceShortMsg	9	Permanent error sending message
mtUnspecifiedProtocolId	9	Permanent error sending message
mtMsgClassNotSupported	9	Permanent error sending message

Error Code	New Code	Text
mtUnspecifiedDataCodingScheme	9	Permanent error sending message
mtTpduNotSupported	9	Permanent error sending message
mtSimStorageFull	10	Temporary congestion error
mtNoSmStorageCapabilityInSim	9	Permanent error sending message
mtErrorinMs	9	Permanent error sending message
mtSimApplToolKitBusy	9	Permanent error sending message
mtSimDataDownloadError	9	Permanent error sending message
mtApplSpecificError	9	Permanent error sending message
mtEquipUnspecifiedErrorCause	9	Permanent error sending message
mtUeDeDeregistered	10	Temporary congestion error
mtNoResponseViaIpsmgw	10	Temporary congestion error
sriSmSystemFailure	10	Temporary congestion error
sriSmDataMissing	9	Permanent error sending message
sriSmUnexpectedDataValue	9	Permanent error sending message
sriSmFacilityNotSupported	9	Permanent error sending message
sriSmUnknownSubscriber	9	Permanent error sending message
sriSmAbsentSubscriber	10	Absent subscriber
sriSmCallBarred	9	Permanent error sending message
sriSmTeleserviceNotProvisioned	9	Permanent error sending message
sriSmOtherMapError	9	Permanent error sending message
sriSmTcapAborted	9	Permanent error sending message
sriSmSccpAborted	9	Permanent error sending message
sriSmMsDeregistered	10	Absent Subscriber
sriSmMsPurged	10	Absent Subscriber
atSystemError	6	General system error
atApplicationNotAvailable	9	Destination not available
atBlockedByThroughputControl	10	Throughput exceeded
atBlockedByAtRule	9	Not accepted - Recipient address is in blacklist
atApplicationNotExist	300	Incorrect destination address
atTxWindowFull	10	Window size exceeded
amsDeviceNotActive	9	Permanent error sending message

Error Code	New Code	Text
amsDbError	9	Permanent error sending message
amsStoreFull	10	Not accepted - Maximum messages for the address exceeded
amsQueueFull	10	Not accepted - Maximum messages for the address exceeded
amsRecipientBufferFull	10	Not accepted - Maximum messages for the address exceeded
amsInvalidQueue	9	Permanent error sending message
amsInvalidMessage	9	Permanent error sending message
amsInvalidValidityTime	305	Incorrect validity parameter usage
amsInvalidDeferredDelivery	308	Incorrect first delivery parameter usage
amsStorageRateExceeded	10	Temporary congestion error
amsDeliveryAttemptsExceeded	9	Permanent error sending message
amsCapabilityNotEnabled	9	Permanent error sending message
rtrBlockedByThroughputControl	10	Temporary congestion error
rtrStorageFailure	10	Temporary congestion error
rtrTimeout	10	Temporary congestion error
rtrBlockedByRule	9	Not accepted - Recipient address is in blacklist
rtrNoRuleMatching	9	Requested operation failed
rtrLicenseExceeded	10	Temporary congestion error
rtrOriginatorInBlackList	9	Not accepted - Recipient address is in blacklist
rtrOriginatorNotInWhiteList	9	Not accepted - Recipient address not in white list
rtrMessageTooLong	2	Message too long
rtrInternalDecodingFailure	9	Permanent error sending message
rtrInvalidSourceApp	9	Permanent error sending message
rtrInvalidOriginator	9	Permanent error sending message
rtrInvalidRecipient	9	Permanent error sending message
rtrMessageSegmentTooLong	9	Permanent error sending message
rtrMessageSarUdhTooLong	9	Permanent error sending message
capScfUnavailable	6	General system error

Error Code	New Code	Text
capUnassignedNumber	9	Permanent error sending message
capUnidentifiedSubscriber	9	Permanent error sending message
capCongestion	10	Temporary congestion error
capFacilityNotSupported	9	Permanent error sending message
capTransferRejected	9	Operation not allowed

A.1.4 Default SMPP Reverse Error Mapping

Error	Normalized	Class
1	invalidMsgLength	Message permanent error
2	invalidCommandLength	Message permanent error
3	invalidCommandId	Message permanent error
4	invalidBindStatusForCmd	Message permanent error
5	alreadyLoggedIn	Destination temporary error
6	invalidPriorityFlag	Message permanent error
7	invalidRegDeliveryFlag	Message permanent error
8	systemError	Destination temporary error
10	invalidSrcAddress	Message permanent error
11	invalidRecipient	Message permanent error
12	invalidMsgId	Message permanent error
13	bindFailed	Destination temporary error
14	invalidPassword	Destination temporary error
15	invalidSystemId	Destination temporary error
17	cancelSmFailed	Message permanent error
19	replaceSmFailed	Message permanent error
20	msgQueueFull	Destination temporary error
21	invalidServiceType	Message permanent error
51	invalidNumDestinations	Message permanent error
52	invalidDistListName	Message permanent error
64	invalidDistFlag	Message permanent error
66	invalidSubmitWithRepRequest	Message permanent error
67	invalidEsmClassFieldData	Message permanent error

Error	Normalized	Class
68	cannotSubmitToDistList	Message permanent error
69	submitFailed	Destination temporary error
72	invalidSrcAddresTon	Message permanent error
73	invalidSrcAddressNpi	Message permanent error
80	invalidDestAddressTon	Message permanent error
81	invalidDestAddressNpi	Message permanent error
83	invalidSystemType	Message permanent error
84	invalidRepIfPresentFlag	Message permanent error
85	invalidNumberOfMessages	Message permanent error
88	throttlingError	Destination temporary error
97	invalidDeliveryTime	Message permanent error
98	invalidValidityPeriod	Message permanent error
99	invalidPredefMessage	Message permanent error
100	receiverTempAppError	Destination temporary error
101	receiverPermAppError	Message permanent error
102	receiverRejectMessage	Message permanent error
103	querySmFailed	Message permanent error
192	errorInOptionalPduPart	Message permanent error
193	optionalParamNotAllowed	Message permanent error
194	invalidParamLength	Message permanent error
195	expectedOptParamMissing	Message permanent error
196	invalidOptParamValue	Message permanent error
254	deliveryFailure	Destination temporary error
255	unknownError	Destination temporary error
256	serviceTypeUnauthorized	Message permanent error
257	requestProhibited	Message permanent error
258	serviceTypeUnavailable	Message permanent error
259	serviceTypeDenied	Message permanent error
260	invalidCodingScheme	Message permanent error
261	invalidSourceSubAddressUnit	Message permanent error
262	invalidDestinationSubAddressUnit	Message permanent error

Error	Normalized	Class
263	invalidBroadcastFrequencyInterval	Message permanent error
264	invalidBroadcastAliasName	Message permanent error
265	invalidBroadcastAreaFormat	Message permanent error
266	invalidNumBroadcastAreas	Message permanent error
267	invalidBroadcastContentType	Message permanent error
268	invalidBroadcastMessageClass	Message permanent error
269	broadcastFailed	Message permanent error
270	queryBroadcastFailed	Message permanent error
271	cancelBroadcastFailed	Message permanent error
272	invalidNumBroadcastRepetition	Message permanent error
273	invalidBroadcastServiceGroup	Message permanent error
274	invalidBroacastChannelIndicator	Message permanent error

A.1.5 Default UCP Reverse Error Mapping

Error	Normalized	Class
1	checksumError	Message permanent error
2	syntaxError	Message permanent error
3	operationNotSupported	Destination temporary error
4	operationNotAllowed	Destination temporary error
5	callBarringActive	Message permanent error
6	invalidRecipient	Message permanent error
7	authenticationFailure	Destination temporary error
8	legitAllCallFailure	Message permanent error
9	gaNotValid	Message permanent error
10	repetitionNotAllowed	Message permanent error
11	legitRepetitionFailure	Message permanent error
12	priorityCallNotAllowed	Message permanent error
13	legitCodePrioCallFailure	Message permanent error
14	urgentMessageNotAllowed	Message permanent error
15	legitUrgentMessageFailure	Message permanent error
16	reverseChargingNotAllowed	Message permanent error

Error	Normalized	Class
17	legitReverseChargingFailure	Message permanent error
18	deferredDeliveryNotAllowed	Message permanent error
19	newAuthCodeNotValid	Message permanent error
20	newLegitCodeNotValid	Message permanent error
21	standardTextNotValid	Message permanent error
22	timePeriodNotValid	Message permanent error
23	messageTypeNotSupported	Message permanent error
24	messageTooLong	Message permanent error
25	requestedTextNotValid	Message permanent error
26	messageTypeNotValidForPager	Message permanent error
27	messageNotFound	Message permanent error
28	extTransparent	Destination temporary error
29	extTransparent	Destination temporary error
30	subscriberHangup	Destination temporary error
31	faxGroupNotSupported	Message permanent error
32	faxMessageTypeNotSupportted	Message permanent error
33	addressAlreadyInList	Message permanent error
34	addressNotInList	Message permanent error
35	listFull	Message permanent error
36	rpidInUse	Message permanent error
37	deliveryInProgress	Message permanent error
38	messageForwarded	Message permanent error
99	extLoginNoResources	Destination temporary error

A.1.6 Default CIMD Reverse Error Mapping

Error	Normalized	Class
1	unexpectedOperation	Message permanent error
2	syntaxError	Message permanent error
3	unsupportedParameter	Destination temporary error
4	smscConnectionLost	Destination temporary error
5	noSmscResponse	Destination temporary error

Error	Normalized	Class
6	generalSystemError	Destination temporary error
7	cannotFindInformation	Message permanent error
8	parameterFormatingError	Message permanent error
9	requestedOperationFailed	Message permanent error
10	tempCongestionError	Destination temporary error
100	invalidLogin	Destination temporary error
101	incorrectAccessType	Destination temporary error
102	tooManyUsersOfSameId	Destination temporary error
103	loginRefused	Destination temporary error
104	invalidWindowSize	Destination temporary error
105	windowingDisabled	Destination temporary error
106	virtualSmscBarring	Destination temporary error
107	invalidSubAddress	Destination temporary error
108	aliasAccountLoginRefused	Destination temporary error
300	invalidRecipient	Message permanent error
301	incorrectNumDestAddresses	Message permanent error
302	syntaxErrorInUserDataParam	Message permanent error
303	incorrectUdhParameterComb	Message permanent error
304	incorrectDcsParamUsage	Message permanent error
305	incorrectVPParam	Message permanent error
306	invalidSrcAddress	Message permanent error
307	incorrectPID	Message permanent error
308	incorrectFirstDelivery	Message permanent error
309	incorrectReplyPath	Message permanent error
310	incorrectStatusReportReq	Message permanent error
311	incorrectCancelEnabled	Message permanent error
312	incorrectPriorityParam	Message permanent error
313	incorrectTariffClassParam	Message permanent error
314	incorrectServiceDescriptionParam	Message permanent error
315	incorrectTransportTypeParam	Message permanent error
316	incorrectMessageTypeParam	Message permanent error

Error	Normalized	Class
317	syntaxError	Message permanent error
318	incorrectMoreMessagesToSendParam	Message permanent error
319	incorrectOperationTimerParam	Message permanent error
320	incorrectDialogueIdParam	Message permanent error
321	incorrectAlphanumericOrigAddress	Message permanent error
322	incorrectAlphanumericAddressData	Message permanent error
400	incorrectAddressInquiryParam	Message permanent error
401	incorrectSctsInquiryParam	Message permanent error
500	incorrectSctsDeliverParam	Message permanent error
501	incorrectModeDeliverParam	Message permanent error
502	incorrectDeliveryParamComb	Message permanent error
600	incorrectSctsForCancel	Message permanent error
601	incorrectAddressForCancel	Message permanent error
602	incorrectModeForCancel	Message permanent error
603	incorrectParamCombforCancel	Message permanent error
800	setChangingPasswordFailed	Message permanent error
801	setChangingPasswordNotAllowed	Message permanent error
900	getUnsupportedParam	Message permanent error

A.2 Error Descriptions

This section describes the HUB's internal, normalized errors in detail.

The HUB's trace framework can help you troubleshoot many errors. For information about tracing, refer to *Tracing Application Traffic*.

A.2.1 General Error Descriptions

intSystemError

Description	Internal system error
Caused by	An unexpected system event (this error is generated internally by the HUB)
Triggered by	Not triggered by a specific operation
How to resolve	Contact NewNet support; updated software may be required

intShuttingDown

Description	Session shutting down
Caused by	The HUB received a message for a session that is shutting down (this error is generated internally by the HUB)
Triggered by	Not triggered by a specific operation
How to resolve	Check why the session was lost; resend the message over a new session

intMxpFailure

Description	Error routing message
Caused by	The HUB is unable to forward the request to the RTR (this error is generated internally by the HUB)
Triggered by	The HUB attempting to forward a request to the RTR
How to resolve	Check the status of the RTR; check if there is congestion on the connection to the RTR

intMxpTimeout

Description	Error routing message
Caused by	The RTR did not respond to the MXP request in time (this error is generated internally by the HUB)
Triggered by	The HUB attempting to forward a request to the RTR
How to resolve	Check the status of the RTR; check if there is congestion on the connection to the RTR

intTxFailure

Description	Error sending message
Caused by	The HUB cannot deliver a message to an application or SMSC (this error is generated internally by the HUB)
Triggered by	The HUB attempting to deliver a message to an application or SMSC
How to resolve	Check syslog for more information

intTemporaryError

Description	Temporary error sending message
Caused by	The HUB cannot send/forward a message to an application or SMSC (this error is generated internally by the HUB)
Triggered by	The HUB attempting to send/forward a message to an application or SMSC

How to resolve	Check syslog for more information; check if there is congestion; ensure that the application/SMSC is not overloaded (the session window/buffer may be full)

intPermanentDestError

The RTR indicated that there was a permanent destination error (by sending MXIP_RTG_REJECT_PERM_DEST to the HUB). This error is generated internally by the HUB.

intPermanentMsgError

Description	Permanent error sending message
Caused by	Permanent error that is related to the received request; for example, a permanent message error response to a modify request when modifications are not allowed (this error is generated internally by the HUB)
Triggered by	Not related to a specific operation
How to resolve	Check if the requested operation is allowed by the RTR

intDestinationNotAvailable

Description	Destination not available
Caused by	The specified destination is no longer available; for example, a session was closed and all pending messages are NACKed with this error (this error is generated internally by the HUB)
Triggered by	A session was closed
How to resolve	Check if the related session was closed

intSourceNotAvailable

The source session is no longer available (this message is mainly reported to the RTR). This error is generated internally by the HUB.

intThroughputExceeded

This error is never reported externally to an application or SMSC. This error is generated internally by the HUB.

intWindowSizeViolation

This error is never reported externally to an application or SMSC. This error is generated internally by the HUB.

intConversionFailed

Description	Character set conversion failed
Caused by	Character set conversion failed (this error is generated internally by the HUB)

Triggered by	The converted message exceeds the maximum message size
How to resolve	Check the failed message and check the related character conversion table

extTransparent

This error is never reported externally to an application or SMSC.

extResponseTimeout

Description	Time out error when sending message
Caused by	A timeout occurred on a pending message
Triggered by	No response was received within the configured timeout
How to resolve	Check the timeout settings

extLoginNoResources

This error is never reported externally to an application or SMSC.

intApplicationNotFound

Description	Application not found
Caused by	The application specified in the log-in request is not found (this error is generated internally by the HUB)
Triggered by	The HUB sent a bind request to a non-existent application
How to resolve	Check if the application specified in the log-in request is configured in the MGR

intAuthenticationFailed

Description	Authentication failure
Caused by	The specified password is incorrect (this error is generated internally by the HUB)
Triggered by	The HUB sent a bind request with an incorrect password
How to resolve	Check if the password used matches the configuration

intApplicationDisabled

Description	A bind request for a disabled application arrived
Caused by	The specified application is disabled (this error is generated internally by the HUB)
Triggered by	The HUB sent a bind request to a disabled application
How to resolve	Ensure that the application specified in the bind request is enabled

intApplicationGroupDisabled

Description	A bind request for a disabled application group arrived
Caused by	The application group of the specified application is disabled (this error is generated internally by the HUB)
Triggered by	The HUB sent a bind request to an application that is in an application group that is disabled
How to resolve	Ensure that the application group is enabled

intSubmitMultiPending

Description	Submit multiple already pending
Caused by	The HUB is busy processing a UCP02 request and received a new request (this error is generated internally by the HUB)
Triggered by	The HUB is busy and received a new request
How to resolve	Ensure that the application using UCP02 only sends a new request when pending requests are finished

A.2.2 Protocol Error Descriptions

checksumError

Description	Checksum error
Caused by	A UCP message contains a checksum error
Triggered by	Not triggered by a specific operation
How to resolve	Ensure that the application or SMSC generates UCP messages with correct checksum values

syntaxError

Description	Syntax error in message
Caused by	Syntax of a UCP message is incorrect
Triggered by	Syntax of a UCP message is incorrect
How to resolve	Ensure that UCP messages are generated according to the specification; check syslog for more information

operationNotSupported

Description	Operation not supported
Caused by	The operation is not supported by the system
Triggered by	The HUB received an operation that it does not support

How to resolve	Check the HUB documentation to verify whether the operation is supported by the HUB

operationNotAllowed

Description	Operation not allowed
Caused by	The operation is not allowed by the system
Triggered by	An application sent a deliver or notification request to the HUB, and the operation is not allowed for the application
How to resolve	Determine which operation caused the error; verify if the operation is allowed for the call party (application or SMSC)

callBarringActive

Description	Address blocked
Caused by	Call barring is active
Triggered by	In a UCP02 multi-submit, all of the divided submit requests failed
How to resolve	Not applicable

invalidRecipient

Description	Invalid recipient
Caused by	AdC is invalid
Triggered by	An invalid recipient address is specified
How to resolve	Check if a valid recipient address was used

authenticationFailure

Description	Authentication failure
Caused by	Authentication failure
Triggered by	A log-in or bind request failed
How to resolve	Check syslog for more information

legitAllCallFailure

Description	Operation not allowed
Caused by	The legitimization code for all calls failed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

gaNotValid

Description	Invalid recipient
Caused by	GA not valid
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

repetitionNotAllowed

Description	Operation not allowed
Caused by	Repetition not allowed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

legitRepetitionFailure

Description	Operation not allowed
Caused by	Legitimization code for repetition failed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

priorityCallNotAllowed

Description	Operation not allowed
Caused by	Priority call not allowed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

legitCodePrioCallFailure

Description	Operation not allowed
Caused by	Legitimization code for priority call failed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

urgentMessageNotAllowed

Description	Operation not allowed
Caused by	Urgent message not allowed
Triggered by	This error is not reported by the HUB

How to resolve	Not applicable

legitUrgentMessageFailure

Description	Operation not allowed
Caused by	Legitimization code for urgent message failed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

reverseChargingNotAllowed

Description	Operation not allowed
Caused by	Reverse charging is not allowed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

legitReverseChargingFailure

Description	Operation not allowed
Caused by	Legitimization code for reverse charging failed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

deferredDeliveryNotAllowed

Description	Operation not allowed
Caused by	Deferred delivery not allowed
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

newAuthCodeNotValid

Description	Operation not allowed
Caused by	New AC not valid
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

newLegitCodeNotValid

Description	Operation not allowed

Caused by	New legitimization code not valid
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

standardTextNotValid

Description	Syntax error in message
Caused by	Standard text is not valid
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

timePeriodNotValid

Description	Invalid time period
Caused by	Time period is not valid
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

messageTypeNotSupported

Description	Invalid message type
Caused by	The system does not support the message type
Triggered by	The HUB does not support the specified message type (for example, MT = 1 or 5)
How to resolve	Verify whether the message type is supported by the UCP specification; if not, contact the application creator

messageTooLong

Description	Invalid message
Caused by	The message is too long
Triggered by	The payload of a UCP message exceeded the maximum length (for example, a UCP51 operation with an Amsg of more than 640 characters)
How to resolve	Verify that the relevant application or SMSC generates valid requests

requestedTextNotValid

Description	Syntax error in message
Caused by	The requested standard text is not valid
Triggered by	This error is not reported by the HUB

How to resolve	Not applicable

messageTypeNotValidForPager

Description	Invalid message type
Caused by	Message type is not valid for the pager type
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

messageNotFound

Description	Message not found
Caused by	Message not found in SMSC
Triggered by	Returned in the case of a modified response in which the specified message cannot be found in the SMSC or AMS
How to resolve	In the case of the AMS, verify that the related message is present in the AMS

subscriberHangup

Description	Recipient error
Caused by	Subscriber hang-up
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

faxGroupNotSupported

Description	Syntax error in message
Caused by	Fax group not supported
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

faxMessageTypeNotSupported

Description	Invalid message type
Caused by	Fax message type not supported
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

addressAlreadyInList

Description	Invalid address
Caused by	Address already in list (60 series)
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

addressNotInList

Description	Invalid address
Caused by	Address not in list (60 series)
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

listFull

Description	Operation not allowed
Caused by	List full, cannot add address to list (60 series)
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

rpidInUse

Description	Operation not allowed
Caused by	RPID already in use
Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

deliveryInProgress

Description	Delivery in progress
Caused by	Delivery in progress
Triggered by	A modified request (such as UCP 54) was forwarded to the RTR/AMS and failed because the RTR/AMS was busy delivering the message
How to resolve	Verify that the related message is present in the AMS

messageForwarded

Description	Delivery in progress
Caused by	Message forwarded

Triggered by	This error is not reported by the HUB
How to resolve	Not applicable

invalidMsgLength

Description	Message length is invalid
Caused by	The payload field in an SMPP message is too long; the SMPP message is invalid
Triggered by	The payload of an SMPP message exceeded the maximum length
How to resolve	Verify that the application or SMSC can generate valid SMPP messages

invalidCommandLength

Description	Command length is invalid
Caused by	The relation between SMPP and total message length is incorrect
Triggered by	This error can occur when the HUB receives a submit request with a total message length of less than 20 bytes
How to resolve	Verify that the application or SMSC can generate valid SMPP messages

invalidCommandId

Description	Invalid command ID
Caused by	The command ID used in an SMPP message is invalid
Triggered by	The command ID used in an SMPP message is invalid
How to resolve	Verify that the application or SMSC can generate valid SMPP messages

invalidBindStatusForCmd

Description	Incorrect BIND status for given command
Caused by	The application sent a submit request before binding
Triggered by	The HUB received a request that is not allowed because the related application did not log in
How to resolve	Check why the application did not log in

alreadyLoggedIn

Description	Session already logged in
Caused by	An application that is already logged in sent a second log-in (bind) request
Triggered by	An application that is already logged in sent a second log-in (bind) request
How to resolve	Check why the application sent a second log-in requests

invalidPriorityFlag

Description	Invalid priority flag
Caused by	Invalid priority flag
Triggered by	This error is not directly reported by the HUB, but can be reported indirectly via the error mapping feature
How to resolve	Check why the application sent a second log-in requests

invalidRegDeliveryFlag

Invalid registered delivery flag.

systemError

System error.

invalidSrcAddress

Description	Invalid source address
Caused by	The HUB received an SMPP, UCP, or CIMD request with an invalid source address
Triggered by	The HUB received a request with a source address that was the wrong length or contained invalid content (for example, an alphanumeric address with non-ASCII characters, or the TON indicates a numerical address but the address contains invalid digits)
How to resolve	Verify that the application can generate valid messages

invalidMsgId

Description	Message ID is invalid
Caused by	The HUB received an SMPP request with an invalid message ID
Triggered by	The HUB received an SMPP request with an invalid message ID
How to resolve	Verify that the application can generate valid SMPP messages

bindFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidPassword

Description	Password too long or contains illegal characters
Caused by	The password in an SMPP, CIMD, or UCP message is not configured according to protocol definitions
Triggered by	An invalid password was specified

How to resolve	Create a trace and verify whether the application makes a valid bind request

invalidSystemId

Description	System ID too long or contains illegal characters
Caused by	The HUB received an SMPP message with an invalid SMPP system ID
Triggered by	The HUB received an SMPP message with an invalid SMPP system ID
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

cancelSmFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

replaceSmFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

msgQueueFull

Description	Message queue full
Caused by	An application used the AO delay time setting and the HUB message queue is full
Triggered by	An application used the AO delay time setting and the HUB message queue is full
How to resolve	Check if the AoDelayTime setting is in line with the number of messages sent by the application

invalidServiceType

Description	Invalid service type
Caused by	The HUB received an SMPP message with an invalid SERVICE_TYPE
Triggered by	The HUB received an SMPP message with an invalid SERVICE_TYPE
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidNumDestinations

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDistListName

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDistFlag

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidSubmitWithRepRequest

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidEsmClassFieldData

The HUB received an SMPP request with an ESM class message type "conversation abort" (0x18).

cannotSubmitToDistList

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

submitFailed

Description	submit_sm, data_sm or submit_multi failed
Caused by	The HUB received an SMPP message with an invalid UserDataHeader
Triggered by	The HUB received an SMPP message with an invalid UserDataHeader
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidSrcAddresTon

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidSrcAddresNpi

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDestAddressTon

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDestAddressNpi

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidSystemType

Description	System type too long or contains illegal characters
Caused by	The HUB received an SMPP message with an invalid SYSTEM_TYPE field
Triggered by	The HUB received an SMPP message with an invalid SYSTEM_TYPE field
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidRepIfPresentFlag

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidNumberOfMessages

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

throttlingError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDeliveryTime

Description	Invalid scheduled delivery time
Caused by	The HUB received an SMPP message with an invalid ScheduledDelivery field
Triggered by	The HUB received an SMPP message with an invalid ScheduledDelivery field
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidValidityPeriod

Description	Invalid message validity period
Caused by	The HUB received an SMPP message with an invalid ValidityPeriod field
Triggered by	The HUB received an SMPP message with an invalid ValidityPeriod field
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidPredefMessage

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

receiverTempAppError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

receiverPermAppError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

receiverRejectMessage

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

querySmFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

errorInOptionalPduPart

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

optionalParamNotAllowed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidParamLength

Description	Invalid parameter length
Caused by	An invalid optional parameter was specified in an SMPP request; optional parameters are specified in TLV (tag, length, value) format, and the HUB will return this error when the value part is incorrect
Triggered by	An invalid optional parameter was specified in an SMPP request
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

expectedOptParamMissing

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidOptParamValue

Description	Invalid TLV value

Caused by	An invalid optional parameter was specified in an SMPP request; optional parameters are specified in TLV (tag, length, value) format, and the HUB will return this error when the value part is incorrect
Triggered by	An invalid optional parameter was specified in an SMPP request
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

deliveryFailure

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

unknownError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

serviceTypeUnauthorized

ESME not authorised to use the specified service_type, hence specified service_type has been denied for use by the given ESME. This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

requestProhibited

ESME is prohibited from using the specified operation, i.e. the PDU request was recognised but is denied to the ESME. This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

serviceTypeUnavailable

Specified service_type is unavailable due to a service outage within the Message Centre. This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

serviceTypeDenied

Specified service_type is denied due to inappropriate message content with respect to the selected service_type. This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidCodingScheme

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidSourceSubAddressUnit

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidDestinationSubAddressUnit

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastFrequencyInterval

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastAliasName

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastAreaFormat

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidNumBroadcastAreas

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastContentType

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastMessageClass

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

broadcastFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

queryBroadcastFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

cancelBroadcastFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidNumBroadcastRepetition

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroadcastServiceGroup

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidBroacastChannelIndicator

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

noSystemIdPresent

Description	system_id field has a length of 0
Caused by	The HUB received an SMPP bind request without a SystemId field
Triggered by	The HUB received an SMPP bind request without a SystemId field
How to resolve	Create a trace and verify whether the application makes a valid SMPP request

invalidAddressRange

Description	A bind was sent that had an address_range that does not correspond to the range defined for the application
Caused by	The HUB received an SMPP bind request that is not in line with the configured APC_app_outside_smpp_address_range in the application
Triggered by	The HUB received an SMPP bind request that is not in line with the configured APC_app_outside_smpp_address_range in the application
How to resolve	Create a trace and verify whether the received address range is in line with the configured address range

unexpectedOperation

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

unsupportedParameter

Description	Operation not supported
Caused by	The HUB received a CIMD message with an invalid message parameter
Triggered by	The HUB received a CIMD message with an invalid message parameter
How to resolve	Create a trace and verify whether the application makes a valid CIMD request

smscConnectionLost

Description	No connection to service center

Caused by	The HUB rejected an outside session; this can happen when the HUB is unable to connect to a related inside session
Triggered by	The HUB rejected an outside session
How to resolve	Check syslog for information about why the outside session was rejected

noSmscResponse

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

generalSystemError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

cannotFindInformation

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

parameterFormatingError

Description	Permanent error sending message
Caused by	The content of a message parameter is incorrect (only applies for a CIMD request); for example, a CIMD request with invalid user data
Triggered by	The content of a message parameter is incorrect
How to resolve	Create a trace and verify whether the application makes a valid CIMD request

requestedOperationFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

tempCongestionError

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidLogin

Description	Invalid login
Caused by	The CLI authentication failed or the configured protocol in the application does not match the used protocol
Triggered by	The CLI authentication failed or the configured protocol in the application does not match the used protocol
How to resolve	Check syslog to determine if the problem is related to CLI

incorrectAccessType

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

tooManyUsersOfSameId

Description	Too many sessions for this login
Caused by	An application used too many outside sessions
Triggered by	An application used too many outside sessions
How to resolve	Use the `tp_session` command-line tool to determine the number of sessions used

loginRefused

Description	Login not allowed
Caused by	The application is not allowed to log in; for example, the application is not active, the session model does not allow outside sessions, or the application exceeded the maximum number of service classes
Triggered by	The application is not allowed to log in
How to resolve	Check syslog for more information

invalidWindowSize

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

windowingDisabled

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

virtualSmscBarring

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

invalidSubAddress

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

aliasAccountLoginRefused

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectNumDestAddresses

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

syntaxErrorInUserDataParam

Description	Syntax error in message
Caused by	The HUB received a CIMD request with invalid user data
Triggered by	The HUB received a CIMD request with invalid user data
How to resolve	Create a trace and verify whether the application makes a valid CIMD request

incorrectUdhParameterComb

Description	Invalid message type
Caused by	The HUB received a CIMD request with a parameter field that has an incorrect length
Triggered by	The HUB received a CIMD request with a parameter field that has an incorrect length
How to resolve	Create a trace and verify whether the application makes a valid CIMD request

incorrectDcsParamUsage

Description	Invalid message type
Caused by	The HUB received a CIMD or UCP request with an invalid DCS field
Triggered by	The HUB received a CIMD or UCP request with an invalid DCS field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD/UCP request

incorrectVPParam

Description	Invalid time period
Caused by	The HUB received a CIMD or UCP request with an invalid validity period (VP) field
Triggered by	The HUB received a CIMD or UCP request with an invalid VP field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD/UCP request

incorrectPID

Description	Syntax error in message

Caused by	The HUB received a CIMD or UCP request with an invalid protocol identifier (PID) field
Triggered by	The HUB received a CIMD or UCP request with an invalid PID field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD/UCP request

incorrectFirstDelivery

Description	Invalid time period
Caused by	The HUB received a CIMD or UCP request with an invalid first delivery timestamp field
Triggered by	The HUB received a CIMD or UCP request with an invalid first delivery timestamp field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD/UCP request

incorrectReplyPath

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid reply path field
Triggered by	The HUB received a CIMD request with an invalid reply path field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectStatusReportReq

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid status report request field
Triggered by	The HUB received a CIMD request with an invalid status report request field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectCancelEnabled

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid "cancel enabled" field
Triggered by	The HUB received a CIMD request with an invalid "cancel enabled" field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectPriorityParam

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid priority field
Triggered by	The HUB received a CIMD request with an invalid priority field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectTariffClassParam

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid tariff class field
Triggered by	The HUB received a CIMD request with an invalid tariff class field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectServiceDescriptionParam

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid service description field
Triggered by	The HUB received a CIMD request with an invalid service description field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectTransportTypeParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectMessageTypeParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectMoreMessagesToSendParam

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid "more messages to send" field
Triggered by	The HUB received a CIMD request with an invalid "more messages to send" field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectOperationTimerParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectDialogueIdParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectAlphanumericOrigAddress

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid alphanumeric originator address field
Triggered by	The HUB received a CIMD request with an invalid alphanumeric originator address field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectAlphanumericAddressData

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectAddressInquiryParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectSctsInquiryParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectSctsDeliverParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectModeDeliverParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectDeliveryParamComb

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectSctsForCancel

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectAddressForCancel

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

incorrectModeForCancel

Description	Syntax error in message
Caused by	The HUB received a CIMD request with an invalid "cancel mode" field
Triggered by	The HUB received a CIMD request with an invalid "cancel mode" field
How to resolve	Create a trace and verify whether the application/SMSC makes a valid CIMD request

incorrectParamCombforCancel

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

setChangingPasswordFailed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

setChangingPasswordNotAllowed

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

getUnsupportedParam

This error is not directly reported by the HUB, but it can be reported indirectly via the error mapping feature.

A.2.3 MT Error Descriptions

mtTimeout

The RTR did not receive an MtFwdSm response message within the time set by the `tcapmaxlongresponsetime` semi-static configuration attribute. MT timeouts are typically caused by signaling problems. Check the per-country and per-network MT timeout counters to identify if the timeouts only occur for a specific country or network. When resolving this issue, it can also be helpful to create an MT counting rule for specific MSC or MSC groups to narrow down the timeouts to specific MSCs.

mtAbsentSubscriber

An MT delivery attempt resulted in an "absent subscriber" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtNoPagingError

An MT delivery attempt resulted in a "No Paging" error.

mtImsiDetach

An MT delivery attempt resulted in an "IMSI Detached" error.

mtRoamingRestrictions

An MT delivery attempt resulted in a "Roaming Restriction" error.

mtSystemFailure

An MT delivery attempt resulted in a "system failure" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtDataMissing

An MT delivery attempt resulted in a "data missing" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtUnexpectedDataValue

An MT delivery attempt resulted in an "expected data value" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtFacilityNotSupported

An MT delivery attempt resulted in a "facility not supported" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtUnidentifiedSubscriber

An MT delivery attempt resulted in an "unidentified subscriber" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtIllegalSubscriber

An MT delivery attempt resulted in an "illegal subscriber" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtIllegalEquipment

An MT delivery attempt resulted in an "illegal equipment" error, as reported in the user error field of the MtFwdSm response message. See 3GPP TS 29.002 section 7.6.1.

mtSubscriberBusyForMt

An MT delivery attempt resulted in a "subscriber busy for MT SMS" error. See 3GPP TS 29.002 section 7.6.1.

mtInvalidSmeAddress

An MT delivery attempt resulted in an "invalid SME address" error.

mtEquipmentProtocolError

An MT delivery attempt resulted in an "equipment protocol" error.

mtEquipmentNotSmEquipped

An MT delivery attempt resulted in an "equipment not SM-equipped" error.

mtMemoryCapacityExceeded

An MT delivery attempt resulted in a "memory capacity exceeded" error. See 3GPP TS 23.040 section 3.3.2.

mtOtherMapError

An MT delivery attempt resulted in another MAP layer error.

mtTcapAborted

An MT delivery attempt resulted in a TCAP layer abort.

mtSccpAborted

An MT delivery attempt resulted in an SCCP layer UDTS message.

mtBlockedByMtRule

An MT delivery attempt was blocked by an MT rule in the RTR.

mtShortMsgType0NotSupported

An MT delivery attempt resulted in a "Short Message Type 0 Not Supported " error.

mtCanNotReplaceShortMsg

An MT delivery attempt resulted in a "Cannot replace short message" error.

mtUnspecifiedProtocolId

An MT delivery attempt resulted in a "Unspecified TP-PID " error.

mtMsgClassNotSupported

An MT delivery attempt resulted in a "Message Class Not Supported" error.

mtUnspecifiedDataCodingScheme

An MT delivery attempt resulted in a "Unspecified TP-DCS" error.

mtTpduNotSupported

An MT delivery attempt resulted in a "TPDU Not supported" error.

mtSimStorageFull

An MT delivery attempt resulted in a "(U)SIM SMS Storage Full" error.

mtNoSmStorageCapabilityInSim

An MT delivery attempt resulted in a "No SMS Storage Capability in (U)SIM" error.

mtErrorinMs

An MT delivery attempt resulted in a "Error in MS" error.

mtSimApplToolKitBusy

An MT delivery attempt resulted in a "(U)SIM Application Tool Kit Busy" error.

mtSimDataDownloadError

An MT delivery attempt resulted in a "(U)SIM Data Download" error.

mtApplSpecificError

An MT delivery attempt resulted in a "Application Specific" error.

mtEquipUnspecifiedErrorCause

An MT delivery attempt resulted in a "Unspecified Error Cause" error.

mtUeDeregistered

An MT delivery attempt to UE in IMS domain resulted in an "UE Deregistered" error.

mtNoResponseViaIpsmGw

An MT delivery attempt to UE in IMS domain resulted in a "No Response Via IPSM-GW" error.

A.2.4 SRI-SM Error Descriptions

sriSmTimeout

The RTR did not receive an SendRoutingInfoForSm response message within the time set by the `tcapmaxresponsetime` semi-static configuration attribute.

sriSmSystemFailure

The location query preceding the MT delivery attempt produced this error.

sriSmDataMissing

The location query preceding the MT delivery attempt produced this error.

sriSmUnexpectedDataValue

The location query preceding the MT delivery attempt produced this error.

sriSmFacilityNotSupported

The location query preceding the MT delivery attempt produced this error.

sriSmUnknownSubscriber

The location query preceding the MT delivery attempt produced this error.

sriSmAbsentSubscriber

The location query preceding the MT delivery attempt produced this error.

sriSmMsDeregistered

The location query preceding the MT delivery attempt produced "Ms Deregistered" error.

sriSmMsPurged

The location query preceding the MT delivery attempt produced "Ms Purged" error.

sriSmCallBarred

The location query preceding the MT delivery attempt produced this error.

sriSmTeleserviceNotProvisioned

The location query preceding the MT delivery attempt produced this error.

sriSmOtherMapError

The location query preceding the MT delivery attempt produced another MAP error.

sriSmTcapAborted

The location query preceding the MT delivery attempt produced a TCAP layer abort.

sriSmSccpAborted

The location query preceding the MT delivery attempt produced a SCCP layer abort.

Note: The 'mtAbsentSubscriber' and 'sriSmAbsentSubscriber' errors have been deprecated. RTR will not be sending these two error codes to HUB anymore. Instead, it will send either 'mtNoPagingError', 'mtImsiDetach' or 'mtRoamingRestrictions' in case it is a MT-FSM error; if it is a SRI-SM error it will send either 'sriSmMsDeregistered' or 'sriSmMsPurged'.

A.2.5 AT Error Descriptions

atSystemError

An AT delivery attempt produce a system error.

atApplicationNotAvailable

An AT delivery attempt is impossible, as the destination application is not available.

atBlockedByThroughputControl

An AT delivery attempt is blocked by a throughput constraint of the destination application.

atBlockedByAtRule

An AT delivery attempt is blocked by an AT rule on the RTR.

atApplicationNotExist

An AT delivery attempt is impossible, as the destination application is unknown.

atTxWindowFull

An AT delivery attempt is impossible, as the transmission window toward the destination application is full.

A.2.6 AMS Error Descriptions

amsDeviceNotActive

Description	SMSC not available
Caused by	The AMS device is not active
Triggered by	The AMS device did not start properly
How to resolve	Ensure that the AMS database is configured

amsDbError

Description	SMSC storage failure
Caused by	Unknown AMS database error
Triggered by	There is a problem with disk recovery
How to resolve	Contact the NewNet Support administrator

amsStoreFull

Description	Message queue full

Caused by	The specified amount of messages is exceeded
Triggered by	Too many messages in the AMS
How to resolve	Ensure that messages are being delivered; check the configuration parameters and the licensed maximum number of messages; contact your NewNet account manager to increase the licensed maximum capacity

amsQueueFull

Description	Message queue full
Caused by	An AMS queue is full
Triggered by	Too many messages in a specific queue
How to resolve	Check the queue size in the MGR and increase, if needed

amsRecipientBufferFull

Description	Message queue full
Caused by	A recipient buffer is full
Triggered by	Too many messages for a specific recipient
How to resolve	Check the allowed number of messages per recipient in the queue maximum size/recipient in the MGR

amsInvalidQueue

Description	Permanent error sending messages
Caused by	The RTR referenced a queue that does not exist
Triggered by	The RTR indicated the wrong queue
How to resolve	Check the RTR properties and rules; ensure that the AMS is configured correctly

amsInvalidMessage

Description	Permanent error sending messages
Caused by	The message is invalid
Triggered by	The RTR sent an invalid message
How to resolve	Trace the message to obtain more information

amsInvalidValidityTime

Description	Invalid message validity period
Caused by	The validity period is in the past

Triggered by	Invalid validity period
How to resolve	Verify that NTP is correctly configured and running on all systems; verify that the incoming AO message has a correct validity period

amsInvalidDeferredDelivery

Description	Invalid scheduled delivery time
Caused by	The deferred delivery time is incorrect
Triggered by	The deferred delivery time is incorrect
How to resolve	Verify that NTP is correctly configured and running on all systems; verify that the incoming AO message has a correct scheduled delivery time

amsStorageRateExceeded

Description	Storage rate exceeded
Caused by	There is too much database action
Triggered by	The input speed is too high
How to resolve	Check the `maxingress` configuration parameter of AMS; a value that is too high will make the system unstable, but a value that is too low will cause rejections

amsDeliveryAttemptsExceeded

Description	Permanent error sending message
Caused by	The AMS is performing too many delivery attempts
Triggered by	The message has already expired by the time it is stored
How to resolve	Check if the maximum retries setting in the delivery scheme is too low or if the delivery scheme time is too short, which can cause the message to expire if the RTR already made a delivery attempt

amsCapabilityNotEnabled

Description	Permanent error sending message
Caused by	The AMS cannot perform the requested action (message store or Icache)
Triggered by	The system configuration (license) is incorrect
How to resolve	Correct the AMS license

A.2.7 RTR Error Descriptions

rtrBlockedByThroughputControl

Throughput limitations are blocking the AO message.

rtrStorageFailure

Storing a message in the AMS failed.

rtrTimeout

The RTR timed out while waiting for a response from the HUB (only applies to AO-AO and AT-AT routing).

rtrBlockedByRule

An AO rule blocked the message on the RTR.

rtrNoRuleMatching

No AO rule matched the message on the RTR, therefore the message gave an error.

rtrLicenseExceeded

The throughput license of the RTR is exceeded. Check the STV license statistics and request and contact your NewNet account manager for a license or system capacity upgrade.

rtrOriginatorInBlackList

An AO message was rejected due to the provisioned originator blacklist for the originating application. See the MGR SMS application originator lists.

rtrOriginatorNotInWhiteList

An AO message was rejected due to the provisioned originator whitelist for the originating application. See the MGR SMS application originator lists.

rtrMessageTooLong

An AO message was rejected, as it contained too many characters to be delivered to an MT destination. Check the application's long message handling settings.

rtrInternalDecodingFailure

An AO message was rejected, due to internal decoding failure of the corresponding MXP message sent from HUB to RTR.

rtrInvalidSourceApp

An AO message was rejected, as RTR and HUB were out-of-sync when the originating application was removed.

rtrInvalidOriginator

An AO message was rejected, as it contained some invalid character in the originator address or the alphanumeric originator address was too long.

rtrInvalidRecipient

An AO message was rejected, as it contained some invalid character in the recipient address.

rtrMessageSegmentTooLong

An AO message, which was actually a segment of a concatenated SMPP message, was rejected as the user data part was too long to be included in a single GSM message.

rtrMessageSarUdhTooLong

A long AO message (i.e. whose user data part was longer than a single GSM message) was rejected, because the total user data size was too long and it could not be properly split up into multiple concatenated GSM message segments.

A.2.8 CAMEL Error Descriptions

capScfUnavailable

An AO message was rejected due to the CAMEL charging failure "SCF Unavailable".

capUnassignedNumber

An AO message was rejected due to the CAMEL charging failure "Unassigned Number".

capUnidentifiedSubscriber

An AO message was rejected due to the CAMEL charging failure "Unidentified Subscriber".

capCongestion

An AO message was rejected due to the CAMEL charging failure "Congestion".

capFacilityNotSupported

An AO message was rejected due to the CAMEL charging failure "Facility Not Supported".

capTransferRejected

An AO message was rejected due to the CAMEL charging failure "Transfer Rejected".

Appendix B

Reason Codes in Delivery Receipts

Topics:

B.1 UCP 53 Notification Reason Codes

If the delivery status of a UCP 53 notification operation is 1 (buffered) or 2 (not delivered), the reason code indicates the type of error that occurred. This section lists the reason codes that the HUB supports when it receives a notification from the AMS.

Reason Code	Description
0	Unknown subscriber
10	Network time-out
101	Unknown subscriber
103	Call barred
106	Facility not supported
107	Absent subscriber
108	Delivery fail
111	MS not equipped
115	MS not a subscriber
118	System fail
120	HLR system failure
126	System failure
127	Unexpected data value

When the HUB is transparently passing AT-AT traffic, it is possible that messages could contain reason codes other than those listed here.

B.2 SMPP Delivery Receipts

The `network_error_code` field of the SMPP DeliverSM delivery receipt indicates the error that occurred in the AGW during the processing or delivery attempt of a message.

The following table lists only the GSM error codes, i.e. which are applicable for mobile-terminated messages.

Note: The value of the `network_error_code` field consists of 3 octets. The first octet indicates the network type, which is '3' for GSM. The remaining two octets contain the actual error code in hexadecimal format, which in the case of GSM would correspond to one of the decimal error values listed in the table below.

Error Code	Description
1	Unknown subscriber

Error Code	Description
5	Unidentified subscriber
9	Illegal subscriber
11	Teleservice not provisioned
12	Illegal MS
13	Call barred
21	Facility not supported
27	Absent subscriber
31	Subscriber busy
32	Delivery failure, caused by the GSM error: • SC congestion • Protocol error • MS not equipped • Unknown service center • Memory capacity exceeded • Delivery failure
34	Delivery failure, caused by the GSM error: • Network timeout • System failure • TCAP error • SCCP error
35	Data missing
36	Unexpected data value

B.3 SMPP Notification Text Error Codes

The HUB maps internal MXP error codes to '`Err`' field values in the SMPP Notification text, which is then sent to the originating application in the '`short_message`' field of the Deliver_SM carrying the delivery receipt.

The table below lists the Notification text error values and the corresponding internal errors:

Notification Text Error	Internal Error
001	MT TIMEOUT
002	MT ABSENT SUBSCRIBER

Notification Text Error	Internal Error
003	MT SYSTEM FAILURE
004	MT DATA MISSING
005	MT UNEXPECTED DATA VALUE
006	MT FACILTY NOT SUPPORTED
007	MT UNIDENTIFIED SUBSCRIBER
008	MT ILLEGAL SUBSCRIBER
009	MT ILLEGAL EQUIPMENT
010	MT SUBSCRIBER BUSY FOR MT SM
011	MT INVALID SME ADDRESS
012	MT EQUIPMENT PROTOCOL ERROR
013	MT EQUIPMENT NOT SM EQUIPPED
014	MT MEMORY CAPACITY EXCEEDED
015	MT OTHER MAP ERROR
016	MT TCAP ABORTED
017	MT SCCP ABORTED
018	MT NO PAGING RESP
019	MT IMSI DETACH
020	MT ROAM RESTRICT
021	MT SMTYPE0 NOT SUPPORTED ERROR
022	MT CANNOT REPLACE SM ERROR
023	MT UNSPECIFIED PROTID ERROR

Notification Text Error	Internal Error
024	MT MESSSAGE CLASS NOT SUPPORTED ERROR
025	MT UNSPECIFIED DATA CODING SCHEME ERROR
026	MT TPDU NOT SUPPORTED ERROR
027	MT SIM SMS STORAGE FULL ERROR
028	MT NO SMS STORAGE CAPABILITY IN SIM ERROR
029	MT ERROR IN MS
030	MT SIM APPLICATION TOOL KIT BUSY ERROR
031	MT SIM DATA DOWNLOAD ERROR
032	MT APPLICATION SPECIFIC ERROR
033	MT EQUIPMENT UNSPECIFIED ERROR CAUSE
034	MT USER EQUIPMENT DEREGISTERED
035	MT NO RESPONSE VIA IPSMGW
255	MT BLOCKED BY MT RULE
101	SRISM TIMEOUT
102	SRISM_SYSTEM_FAILURE
103	SRISM_DATA_MISSING
104	SRISM_UNEXPECTED_DATA_VALUE
105	SRISM_FACILITY_NOT_SUPPORTED
106	SRISM_UNKNOWN_SUBSCRIBER
107	SRISM_ABSENT_SUBSCRIBER
108	SRISM_CALL_BARRED

Notification Text Error	Internal Error
109	SRISM_TELESERVICE_NOT_PROVISIONED
110	SRISM_OTHER_MAP_ERROR
111	SRISM_TCAP_ABORTED
112	SRISM_SCCP_ABORTED
113	SRISM_MS_DEREGISTERED
114	SRISM_MS_PURGED
201	AT SYSTEM ERROR
202	AT SHUTTING DOWN
203	AT MXP FAILURE
204	AT MXP TIMEOUT
205	AT TX FAILURE
206	AT TEMPORARY ERROR
207	AT PERMANENT DEST ERROR
208	AT PERMANENT MSG ERROR
209	AT APP NOT AVAILABLE
210	AT SOURCE NOT AVAILABLE
211	AT BLOCKED BY THROUGHPUT CONTROL
212	AT LOGIN INVALID
213	AT LOGIN NOT ALLOWED
214	AT LOGIN TOO MANY SESSIONS
215	AT LOGIN NO SMSC CONNECTION

Notification Text Error	Internal Error
216	AT ALREADY LOGGED IN
217	AT OPERATION NOT ALLOWED
218	AT OPERATION NOT SUPPORTED
219	AT RESPONSE TIMEOUT
220	AT INVALID SYNTAX
221	AT INVALID CHECKSUM
222	AT BLOCKED BY AT RULE
223	AT RECIPIENT ERROR
224	AT APP NOT EXISTING
225	AT INVALID MSG TYPE
226	AT INVALID MSG
227	AT INVALID TIME PERIOD
228	AT INVALID ADDRESS
229	AT MSG NOT FOUND
230	AT DELIVERY IN PROGRESS
231	AT TRANSPARENT
232	AT LOGIN NO RESOURCES
233	AT TX WINDOW FULL
234	AT MAX MESSAGES BUFFERED
235	AT INVALID BIND STATUS
236	AT NO ROUTING RULE

Notification Text Error	Internal Error
237	AT CONVERSION FAILED
301	AMS DEVICE NOT ACTIVE
302	AMS DB ERROR
303	AMS STORE FULL
304	AMS QUEUE FULL
305	AMS RECIP BUFFER FULL
306	AMS INVALID QUEUE
307	AMS INVALID MESSAGE
308	AMS INVALID VALIDITY TIME
309	AMS INVALID DEFERRED DELIVERY
310	AMS STORAGE RATE EXCEEDED
311	AMS DELIVERY ATTEMPTS EXCEEDED
312	AMS CAPABILITY NOT ENABLED
400	RTR BLOCKED BY THROUGHPUT CONTROL
401	RTR STORAGE FAILURE
402	RTR TIMEOUT
403	RTR BLOCKED BY RULE
404	RTR NO RULE MATCHING
405	RTR LICENSE EXCEEDED
406	RTR ORIGINATOR IN BLACK LIST
407	RTR ORIGINATOR NOT IN WHITE LIST

Notification Text Error	Internal Error
408	RTR MESSAGE TOO LONG
409	RTR INTERNAL DECODING FAILURE
410	RTR INVALID SOURCE APP
411	RTR INVALID ORIGINATOR
412	RTR INVALID RECIPIENT
413	RTR MESSAGE SEGMENT TOO LONG
414	RTR MESSAGE SAR UDH_TOO_LONG
700	CAP SCF UNAVAILABLE
701	CAP UNASSIGNED NUMBER
702	CAP UNIDENTFIED SUBSCRIBER
703	CAP CONGESTION
704	CAP FACILITY NOT SUPPORTED
705	CAP SM TRANSFER REJECTED

Note: As per the current functionality implemented in the HUB and RTR, the RTR, AMS and CAP internal errors in the above table are not relevant as far as the generation of Notification text error values is concerned. This is because whenever any internal error belonging to the above three categories is returned, RTR rejects the corresponding AO messages with NACK and no separate notification is generated.

B.4 CIMD2 Delivery Response

This section lists the CIMD2 delivery response error codes.

Error Code	Description
0	No error
1	Unexpected operation
2	Syntax error

Error Code	Description
3	Unsupported parameter error
4	Connection to SMS center lost
5	No response from SMS center
6	General system error
7	Cannot find information
8	Parameter formatting error
9	Requested operation failed
10	Temporary congestion error

For backward compatibility, the congestion error can be configured using the semi-static configuration attribute `aothroughputerrforcimd`. The error code can be a numeric value between 0 and 10.

Appendix C

Sample Configuration File

Topics:

C.1 Sample Common Configuration File

```
<tpconfig
    networkdiscoverymulticastaddress="{239.255.xx.yy}"
    networkdiscoverynetworkaddress="{network-address}"
    networkdiscoverynetworkmask="{network-mask}"
    hubsmpppluspprefixhandling="false"
    >


    <!-- Configurable notification text for UCP53
    -->

    <hubnotificationtext notificationstatus="deliverySuccessful"
       formatstring="Message for %MRAD, with identification %SDD%SDM%SDY%STH%STM%STS
 has been delivered on %DDM/%DDD/%DDX at %DTI:%DTM%DTP."
        usageofdefaults="always" defaulterrorcode="19" defaulterrortext="no error"
 defaultucpreasoncode="12" />
    <hubnotificationtext notificationstatus="inProgress"
       formatstring="Message for %MRAD, with identification %SDD%SDM%SDY%STH%STM%STS
 has been buffered."
        usageofdefaults="usageIncaseOfNoError" defaulterrorcode="1404"
defaulterrortext="Still in progress"
        defaultucpreasoncode="244" />


   <!-- Configurable notification error mapping, this sample contains the complete
 set of
        default mapped notification errors. See HUB operator manual for more info.
    -->

   <hubnotificationerrormapping id="1"  internalerror="defaultNotificationMapping"

        errorcode="108" errortext="Absent subscriber" ucpreasoncode="108" />
    <hubnotificationerrormapping id="2"  internalerror="mtTimeout"
        errorcode="10" errortext="Network time-out" ucpreasoncode="10" />
    <hubnotificationerrormapping id="3"  internalerror="sriSmTimeout"
        errorcode="10" errortext="Network time-out" ucpreasoncode="10" />
    <hubnotificationerrormapping id="4"  internalerror="sriSmUnknownSubscriber"
        errorcode="101" errortext="Unknown subscriber" ucpreasoncode="101" />
    <hubnotificationerrormapping id="5"  internalerror="mtAbsentSubscriber"
        errorcode="107" errortext="Absent subscriber" ucpreasoncode="107" />
    <hubnotificationerrormapping id="6"  internalerror="sriSmAbsentSubscriber"
        errorcode="107" errortext="Absent subscriber" ucpreasoncode="107" />
    <hubnotificationerrormapping id="7"  internalerror="mtFacilityNotSupported"
        errorcode="106" errortext="Facility not supported" ucpreasoncode="106" />
    <hubnotificationerrormapping id="8"  internalerror="sriSmFacilityNotSupported"

        errorcode="106" errortext="Facility not supported" ucpreasoncode="106" />
    <hubnotificationerrormapping id="9"  internalerror="sriSmSystemFailure"
        errorcode="120" errortext="HLR system failure" ucpreasoncode="120" />
    <hubnotificationerrormapping id="10"  internalerror="mtSystemFailure"
        errorcode="118" errortext="System fail" ucpreasoncode="118" />
    <hubnotificationerrormapping id="11"  internalerror="sriSmCallBarred"
        errorcode="103" errortext="Call barred" ucpreasoncode="103" />
    <hubnotificationerrormapping id="12"  internalerror="mtOtherMapError"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
    <hubnotificationerrormapping id="13"  internalerror="sriSmOtherMapError"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
    <hubnotificationerrormapping id="14"  internalerror="mtTcapAborted"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
```

```
    <hubnotificationerrormapping id="15"  internalerror="mtSccpAborted"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
    <hubnotificationerrormapping id="16"  internalerror="sriSmTcapAborted"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
    <hubnotificationerrormapping id="17" internalerror="sriSmSccpAborted"
        errorcode="126" errortext="System failure" ucpreasoncode="126" />
    <hubnotificationerrormapping id="18"  internalerror="mtUnexpectedDataValue"
        errorcode="127" errortext="Unexpected data value" ucpreasoncode="127" />
    <hubnotificationerrormapping id="19"  internalerror="mtEquipmentProtocolError"

        errorcode="111" errortext="MS not equipped" ucpreasoncode="111" />
    <hubnotificationerrormapping id="20" internalerror="mtEquipmentNotSmEquipped"
        errorcode="111" errortext="MS not equipped" ucpreasoncode="111" />
    <hubnotificationerrormapping id="21" internalerror="mtIllegalSubscriber"
        errorcode="115" errortext="MS not a subscriber" ucpreasoncode="115" />
    <hubnotificationerrormapping id="22" internalerror="mtUnidentifiedSubscriber"
        errorcode="115" errortext="MS not a subscriber" ucpreasoncode="115" />

</tpconfig>
```

C.2 Sample Host-Specific Configuration File

```
<tpconfig
    ipaddress="127.0.0.1"
    runtextpassprocess="false"
    runtexthubprocess="true"
    runtpfclientprocess="true"
    hubipaddressowninternal="10.2.3.12"
    hubipv6addressowninternal="fd5b:bd9d:c3e0:100::135"
    hubsmppplusprefixhandling="false"
  >


    <fxferfile
       localpath="/usr/TextPass/etc/common_config.txt"
       serverpath="/usr/TextPass/etc/common_config.txt"
       validate="/usr/TextPass/bin/tp_config --validatecommonconfig SERVERFILE"
    />
    <fxferfile
        localpath="/usr/TextPass/etc/MGRdata.xml.gz"
        serverpath="/usr/TextPass/etc/MGRdata.xml.127.0.0.1.gz"
    />

    <trapreceiver ipaddress="127.0.0.1" udpport="11173" />

</tpconfig>
```

Appendix D

References

Topics:

D.1 References

1. GSM 09.02 version 7.5.0 Release 1998; Digital cellular telecommunications systems (Phase 2+); Mobile Application Part (MAP) specification
2. 3GPP TS 23.066 version 4.0.0 Release 4; Digital cellular telecommunications systems (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Support of Mobile Number Portability (MNP); Technical Realisation; Stage 2
3. 3GPP TS 29.002 version 3.20 Release 1999; digital cellular telecommunications systems (Phase 2+); Mobile Application Part (MAP) specification
4. Short Message Service Centre EMI - UCP Specification version 5.1, Document version 4.8, April 2005
5. Short Message Peer to Peer Protocol Specification v3.4 Document version: 12-Oct-1999 Issue 1.2 at *http://smsforum.net*
6. The Simple Network Management Protocol IETF - RFC 1157
7. Sun Security Blueprints at *http://wwws.sun.com/software/security/blueprints/index.html*
8. Solaris documentation at *http://docs.oracle.com*
9. Solaris Tunable Parameters Reference Manual at *http://download.oracle.com/docs/cd/E19253-01/817-0404/*
10. RedHat documentation at *http://www.redhat.com/docs*
11. NewNet Mobile Messaging UCP Protocol Implementation Compliance Statement
12. NewNet Mobile Messaging SMPP Protocol Implementation Compliance Statement
13. NewNet Mobile Messaging Traffic Element Installation Manual
14. NewNet Mobile Messaging RTR Operator Manual
15. NewNet Mobile Messaging Tools Operator Manual
16. NewNet Mobile Messaging MGR Operator Manual
17. NewNet Mobile Messaging AMS Operator Manual
18. NewNet Mobile Messaging SNMP Trap Reference Guide

Glossary

#

3GPP	3rd Generation Partnership Project

A

ACK	Data Acknowledgement
AGW	Application Gateway A gateway between SMS applications and service centres provided by the Router, HUB, and AMS components.
AMS	Active Message Store Provides store-and-forward functionality for SMS messages.
AO	Application Originated Short message traffic that is originated by an application.
application	The telecommunications software that is hosted on the platform. A service provided to subscribers to a network; for example, voice over IP (VoIP), video on demand (VoD), video conferencing, or gaming.
architecture	Used to conceptually describe the function, interaction, and connectivity of hardware, software, and/or system components within a network.
ARP	Address Resolution Protocol ARP monitoring uses the Address Resolution Protocol to determine

A

	whether a remote interface is reachable. Auto Reply service Personalized SMS auto reply service. This service is provided by the Mobile Messaging XS-ARP component.
ASCII	American Standard Code for Information Interchange
AT	Application Terminated Short message traffic that terminates at an application.
ATIC	Incoming application-terminated counting Counting rule that operates on incoming application-terminated (AT) messages.
ATIR	Incoming application-terminated routing Routing rule that operates on incoming application-terminated (AT) messages.
ATIX	Incoming application-terminated eXternal condition External condition rule that operates on incoming application-originated (AO) messages.
ATOC	Outgoing application-terminated counting Counting rule that operates on outgoing application-terminated (AT) messages.

A

ATOR	Outgoing application-terminated routing Routing rule that operates on outgoing application-terminated (AT) messages.
ATOX	Outgoing application-terminated eXternal condition External condition rule that operates on outgoing application-originated (AO) messages.

B

bandwidth	The data rate supported by a network connection or interface; most commonly expressed in terms of bytes per second (bps).

C

CDR	Call Detail Record This refers to the recording of all connections in a database to permit activities such as billing connection charges or network analysis. CDR files are used in public switched networks, IP networks, for IP telephony, and mobile communications networks. Charging Data Record Used for user billing: a telecom provider transfers them from time to time in order to send bills to their users.
Checksum	Provides protection against data corruption in the network. The sender of a packet computes a checksum according to an algorithm. The receiver then re-computes the checksum, using the same algorithm. The packet is accepted if the checksum is valid; otherwise, the packet is discarded.

C

CIMD	Computer Interface for Message Distribution Proprietary SMSC protocol developed by Nokia.
CLI	Custom LSMS Interface Command-line interface Calling Line Identification
CRC	Cyclic Redundancy Check A number derived from, and stored or transmitted with, a block of data in order to detect corruption. By recalculating the CRC and comparing it to the value originally transmitted, the receiver can detect some types of transmission errors.

D

DCS	Data Coding Scheme
Diameter	Protocol that provides an Authentication, Authorization, and Accounting (AAA) framework for applications such as network access or IP mobility. Diameter works in both local and roaming AAA situations. Diameter can also be used as a signaling protocol for mobility management which is typically associated with an IMS or wireless type of environment. Diameter is the successor to the RADIUS protocol. The MPE device supports a range of Diameter interfaces, including Rx, Gx, Gy, and Ty.

E

EC	External Condition

E

Condition that is passed on the external condition interface.

ECI

External condition interface

Interface for communicating with external condition applications.

EMI

External Machine Interface

Protocol used to connect to SMSCs, developed by LogicaCMG.

ESME

External Short Message Entity

The remote-destination entities on the IP network that is connected to using SMPP protocol.

F

failover

The capability to automatically switch to a redundant or backup server, system, or network when the previously active server, system, or network fails or terminates abnormally. In certain instances, however, automatic failover may not be desirable, and human intervention may be required to initiate the failover manually.

FCDR

SMSC-compatible ASN.1 CDR format

FDA

First Delivery Attempt

Approximately 85 to 90 percent of SMS traffic gets through on first delivery attempt (FDA). That means that all of the initial processing that the SMSC does to store, query and forward messages is to a certain extent a waste of

F

	processing power — it would be much more cost-effective for an operator if a less expensive piece of equipment could first attempt to deliver the message.

G

GSM	Global System for Mobile Communications
GT	Global Title Routing Indicator
GUI	Graphical User Interface The term given to that set of items and facilities which provide the user with a graphic means for manipulating screen data rather than being limited to character based commands.

H

HPLMN	Home Public Land Mobile Network
HUB	Works in combination with the Router to manage traffic to and from SMS applications.

I

Icache	Intermediate Cache Enables the Mobile Messaging system to store the state and certain parameters of a short message while it is being processed by an external SMSC.
ID	Identity, identifier
IETF	Internet Engineering Task Force

I

IMSI	International Mobile Subscriber Identity
IP	Internet Protocol IP specifies the format of packets, also called datagrams, and the addressing scheme. The network layer for the TCP/IP protocol suite widely used on Ethernet networks, defined in STD 5, RFC 791. IP is a connectionless, best-effort packet switching protocol. It provides packet routing, fragmentation and re-assembly through the data link layer.
ISDN	Integrated Services Digital Network Integrates a number of services to form a transmission network. For example, the ISDN network integrates, telephony, facsimile, teletext, Datex-J, video telephony and data transfer services, providing users with various digital service over a single interface: voice, text, images, and other data.
ISO	International Standards Organization

M

MAC	Media Access Control Address The unique serial number burned into the Ethernet adapter that identifies that network card from all others.
MAP	Mobile Application Part

M

max	maximum
MC	Message Copy A feature that provides the ability to forward a copy of a Diameter Request message received by or routed through the Diameter Signaling Router to a Diameter Application Server (a DAS peer). This capability is triggered based on configuration or can be dictated by a Diameter Agent Application (DAA).
MGR	A Web-based interface for managing NewNet Mobile Messaging components. Prior to Suite 6, the Configuration Manager (CM) provided this functionality.
MIB	Management Information Database
MNP	Mobile Number Portability
MO	Mobile Originated Refers to a connection established by a mobile communication subscriber. Everything initiated by the mobile station is known as mobile originated.
MS	Mobile Station The equipment required for communication with a wireless telephone network.
MSC	Mobile Switching Center

M

MSISDN

Mobile Station International Subscriber Directory Number

The MSISDN is the network specific subscriber number of a mobile communications subscriber. This is normally the phone number that is used to reach the subscriber.

MT

Mobile Terminated

All transmissions that reach the mobile station and are accepted by it, such as calls or short messages.

MXP

Message eXchange Protocol

NewNet proprietary protocol used for communication between the Mobile Messaging HUB, RTR, and AMS components.

N

NPI

Number Plan Indicator

O

OS

Operating System

P

PBC

Prepaid Billing Controller

Performs prepaid charging using the Diameter, CAMEL, or SMPP+ interface.

PC

Point Code

The identifier of a signaling point or service control point in a network. The format of the point code can be one of the following types:

- ANSI point codes in the format network indicator-network

P

	cluster-network cluster member (**ni-nc-ncm**). • Non-ANSI domestic point codes in the format network indicator-network cluster-network cluster member (**ni-nc-ncm**). • Cluster point codes in the format network indicator-network cluster-* or network indicator-*-*. • ITU international point codes in the format **zone-area-id**. • ITU national point codes in the format of a 5-digit number (**nnnnn**), or 2, 3, or 4 numbers (members) separated by dashes (**m1-m2-m3-m4**) as defined by the Flexible Point Code system option. A group code is required (**m1-m2-m3-m4-gc**) when the ITUDUPPC feature is turned on. • 24-bit ITU national point codes in the format main signaling area-subsignaling area-service point (**msa-ssa-sp**). • 16-bit Japanese SS7 international point codes in the format of a 5-digit decimal number (nnnnn), or 3 numbers separated by dashes, i.e. Main number area - Sub number area - Unit number (M-S-U).
PID	Password ID Process ID Protocol ID
ping	A network tool used to determine if a target host can be reached across an IP network. Ping estimates the round-trip time and

P

	packet loss (if any) rate between hosts.
Provisioning	Static and longer-term management tasks. These may include selection of network equipment, replacement of network equipment, interface additions or deletions, link speed modifications, topology changes, and capacity planning. This term is often used interchangeably with configuration.
PSTN	Public Switched Telephone Network.

Q

QoS	Quality of Service Control mechanisms that guarantee a certain level of performance to a data flow.

R

RFC	Request for Comment RFCs are standards-track documents, which are official specifications of the Internet protocol suite defined by the Internet Engineering Task Force (IETF) and its steering group the IESG.
RTR	Router Routes all types of SMS traffic.
RX	Receive

S

SC	Site Collector System Controller

S

SCCP	Signaling Connection Control Part
SCTP	Stream Control Transmission Protocol
SCTP	Stream Control Transmission Protocol An IETF transport layer protocol, similar to TCP that sends a message in one operation. The transport layer for all standard IETF-SIGTRAN protocols. SCTP is a reliable transport protocol that operates on top of a connectionless packet network such as IP and is functionally equivalent to TCP. It establishes a connection between two endpoints (called an association; in TCP, these are sockets) for transmission of user messages.
SGSN	Serving GPRS Support Node
SIGTRAN	The name given to an IETF working group that produced specifications for a family of protocols that provide reliable datagram service and user layer adaptations for SS7 and ISDN communications protocols. The most significant protocol defined by the SIGTRAN group was the Stream Control Transmission Protocol (SCTP), which is used to carry PSTN signalling over IP. The SIGTRAN group was significantly influenced by telecommunications engineers intent on using the new protocols for adapting VoIP networks to the PSTN with special regard to signaling applications. Recently, SCTP is finding applications beyond its

S

	original purpose wherever reliable datagram service is desired.
SM	Short Message
SMPP	Short Message Peer-to-Peer Protocol An open, industry standard protocol that provides a flexible data communications interface for transfer of short message data.
SMS	Short Message Service
SMSC	Short Message Service Center
SMT	Scroll (area) Message Text
SNMP	Simple Network Management Protocol. An industry-wide standard protocol used for network management. The SNMP agent maintains data variables that represent aspects of the network. These variables are called managed objects and are stored in a management information base (MIB). The SNMP protocol arranges managed objects into groups.
SS7	Signaling System #7
SSH	Secure Shell A protocol for secure remote login and other network services over an insecure network. SSH encrypts and authenticates all EAGLE 5 ISS IPUI and MCP traffic, incoming and

S

	outgoing (including passwords) to effectively eliminate eavesdropping, connection hijacking, and other network-level attacks.

T

TCP	Transfer Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLV	Type/Length/Value
TON	Type of Number
Tools	A collection of command-line tools for managing and troubleshooting NewNet Mobile Messaging components.
trap	A mechanism used in the context of SNMP (Simple Network Management Protocol) for one-way event notification.
TS	Test Strategy Traffic Server Technical Specification Teleservices
TX	Transmit

U

UCP	Universal Computer Protocol Protocol used to connect to SMSCs.

U

UDH	User Data Header
UDP	User Datagram Protocol
UDT	Unitdata Transfer
UDTS	Unitdata Transfer Service An error response to a UDT message.
UID	User ID
UMTS	Universal Mobile Telecommunications System The standard for 3G used by GSM service providers. UMTS includes voice and audio services, for fast data, graphic and text transmissions, along with transmission of moving images and video.
UTF-8	Variable-length character encoding for Unicode that is backward-compatible with ASCII.

X

XML	eXtensible Markup Language A version of the Standard Generalized Markup Language (SGML) that allows Web developers to create customized tags for additional functionality.
XS	eXternal Service Value-adding component that communicates with the Router to provide a service.