ZephyrTel Mobile Messaging AMS R01.13.01

Operator Manual

Release 22.12 Revision A

May 2023

ZephyrTel

CloudForward

Table of Contents

List of Figures

Chapter 1

Introduction

Topics:

1.1 About this Document

This document provides an overview of the ZephyrTel Mobile Messaging Active Message Store (AMS) component. The AMS provides the store-and forward functionality in an SMS network.

The ZephyrTel Mobile Messaging product family is the ZephyrTel product family of core network (SS7 and SIGTRAN) routing and application message routing, storage and mobile network querying products.

The purpose of this document is to inform the reader about the purpose, functionality, architecture, and interfaces of the AMS.

The ZephyrTel Mobile Messaging Routers and HUBs complement the AMS to form a complete SMS network that implements SMSC functionality and more. The Routers provide the SS7 interface toward the PLMN, while the HUB interfaces toward SMS applications (service providers) via TCP/IP.

1.2 Scope

This document discusses the functionality of the ZephyrTel Mobile Messaging AMS component.

1.3 Intended Audience

This document is meant for everyone interested in the functionality offered by the AMS, but specifically for:

- Wireless network operators who want to know which SMS bottlenecks and network issues can be easily be solved with ZephyrTel Mobile Messaging products.
- Original Equipment Manufacturers (OEMs) who intend to make ZephyrTel Mobile Messaging an integrated part of their product or SMS solution.
- System Integrators who need an overview of the ZephyrTel Mobile Messaging functionality and required components for an SMS implementation project.
- SMS Network Engineers who are responsible for the SMS infrastructure of the wireless network and require more knowledge on the powerful possibilities of ZephyrTel Mobile Messaging products.

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 ZephyrTel's Customer Support site is restricted to current ZephyrTel customers only. This section describes how to log into the ZephyrTel 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 ZephyrTel 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

In today's SMS world, the complexity of the SMS environment and the market requirements on network operators are increasing every day. SMS traffic patterns show more and higher peaks, while quality of service demands are growing and the focus on cost-efficiency is getting stronger. Traditional Short Message Service Centres (SMSCs) can no longer cope with performance demands and operators' increasing requirements to route and deliver SMS traffic in an intelligent way.

These challenges can only be solved by a modern SMS network infrastructure.

2.2 SMS Network Context

The true next generation of SMS is the SMS network. As in any well-configured network, the SMS network contains routers, hubs, and storage elements as dictated by modern networking and data processing environments.

The ZephyrTel Mobile Messaging philosophy is that SMS as a data service should be based on a real network structure, as opposed to the central host system on which the traditional SMSC is based.

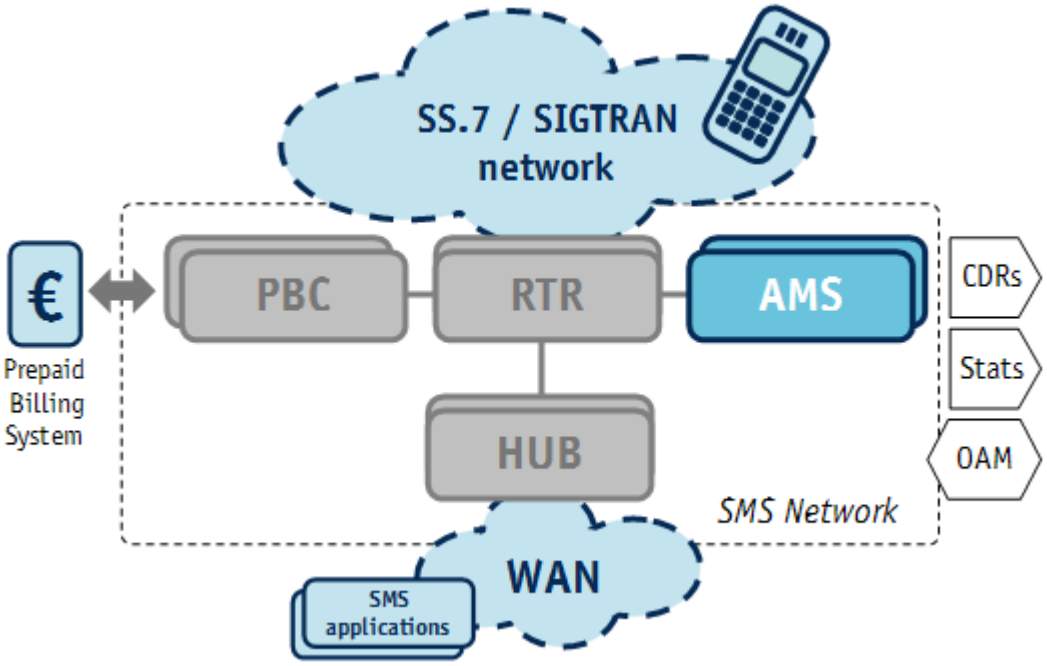


Figure 1: SMS network components

2.2.1 RTR: Interfacing to the PLMN

As depicted in the diagram, the key component is the Router (RTR), which allows routing of SMS traffic from anywhere to virtually everywhere. RTRs can distribute SMS traffic over multiple AMS nodes and/or legacy SMSCs using multi-criteria load distribution and throughput control.

RTRs provide first delivery attempt (FDA) functionality by routing SMS traffic directly to the mobile station (MS) or directly to SMS applications via the HUBs. The latter is especially useful for interactive events, such as SMS voting, in which many SMS votes are sent to the voting application in a very short period of time.

The RTR can be expanded with SMS Firewall (FWL) functionality or a separate FWL platform (not depicted here) can be installed between the SS7/SIGTRAN network and the RTRs. The FWL can monitor and filter all SMS traffic entering the SMS network, that is, the incoming and outgoing SMS MO and MT traffic. In most mobile networks, only incoming foreign MO and MT traffic is monitored and checked for spoofing and spam.

2.2.2 HUB: Interfacing to the Applications

Through the HUB, any SMS application can be connected to the SMS network using SMS protocols such as SMPP, UCP, or CIMD2. Similarly, mobile-originated (MO) SMS traffic that is destined for interactive/voting applications is sent directly to the related application via the HUBs and RTRs. This distributed approach significantly reduces the SMS bottlenecks and improves the overall quality of the SMS services offered.

2.2.3 PBC: Prepaid Billing

The Prepaid Billing Controller (PBC) provides the interface toward prepaid billing systems. In addition to IN interfaces via SS7 (via the RTRs), TCP/IP-based interfaces toward third-party prepaid billing systems are available.

2.2.4 AMS: Message Storage and Delivery Scheduling

The Active Message Store (AMS) is the intelligent store-and-forward component of the ZephyrTel Mobile Messaging Product Family. The AMS has permanent storage facilities on board, and each logical storage entity has its own dedicated delivery scheme. Messages can be buffered in the AMS for delivery at a later time. The AMS supports both store-and-forward and try-and-store behaviour.

The AMS is practically unlimitedly scalable, both in performance and redundancy.

2.3 AMS in the SMS Network

This section discusses the context spaces and their functional interfaces toward the AMS. Here, the AMS is the store-and-forward component in a true SMS network. In this configuration, the AMS interacts with the RTRs in the SMS network.

Note that the AMS will go into an operational state even if no RTR is present in the network (or detected by the AMS).

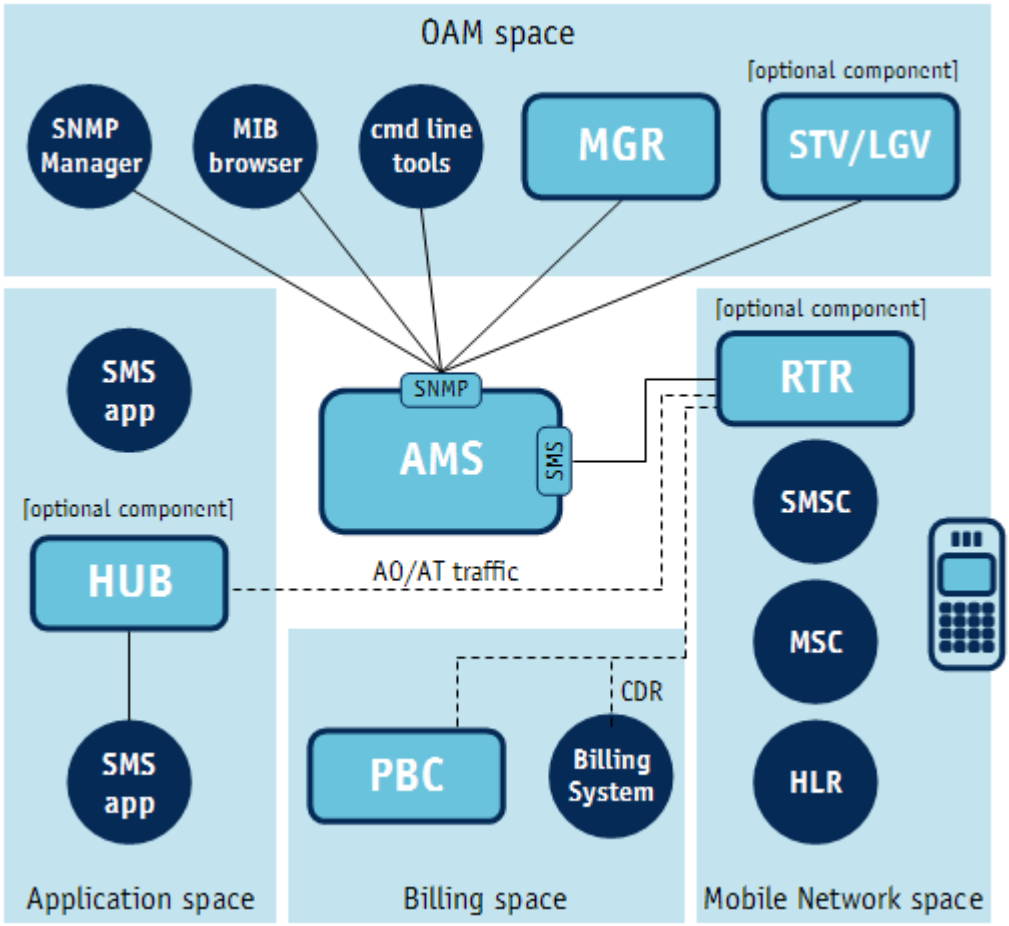


Figure 2: SMS network context

SMS applications (connected to the HUB) forward incoming application-originating (AO) messages to a RTR for immediate delivery or to the AMS for scheduled delivery. The HUB is also responsible for delivering application-terminating (AT) messages (from the RTRs or AMS nodes) to SMS applications.

The RTRs are connected to the SS7 or SIGTRAN network. The RTRs receive MO traffic from the network and deliver MT messages to the network (using the HLR and MSC to deliver to the mobile station).

The RTRs deliver mobile-terminating (MT) messages, while the HUB delivers application-terminating (AT) messages. Both deliveries are based on the applicable AMS delivery scheme for each message. In this context, the RTR typically delivers MO-AT messages directly to the application via the HUB and only uses the AMS when delivery to the destination application fails.

Similarly, the RTRs can deliver MO-MT messages directly to the mobile station and only use the AMS if the delivery failed. The failed message will be placed in the appropriate message queue in the AMS, which will perform delivery attempts according the delivery scheme.

2.3.1 Mobile Network Space

The PLMN interface (that is, the SS7 interface toward the mobile network) is arranged entirely by the hardware and router system software. The following components are part of the mobile network space:

- RTR—RTR handles all MO and MT messages and routes AO/AT messages from the HUB.
- SMSC—The legacy SMSC can serve as a destination for MO or AO traffic and can deliver MT or AT traffic.
- MSC—The MSC or SGSN is the originating and terminating point in the PLMN of MO and MT messages, respectively.
- HLR—The RTR uses the Home Location Register (HLR) to query subscriber information (that is, the destination MSC for an SMS delivery).

2.3.2 Application Space

The application space contains the applications and HUB interfacing with the RTR. The application space contains the following components (at a minimum):

- HUB—An SMPP/UCP/CIMD2 load distributor/concentrator acting as a proxy for all sessions from SMS applications and handling authentication and session control of all sessions toward the RTRs and AMS systems.
- SMS Application—The application responsible for sending (AO) and/or receiving (AT) traffic for a short number, using the SMPP/UCP/CIMD2 protocol over TCP/IP.

2.3.3 Operations and Maintenance and Provisioning Space

The operations and maintenance provisioning space (OAM) represents the interface with operations and maintenance software, including the Manager (MGR), Statistics Viewer (STV), Log Viewer (LGV), 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—Represents SNMP-based network management tools that may be available. This is the system that is responsible for capturing and processing all SNMP alarms.
- MIB Browser—Application that can be used to view and modify the contents of various SNMP variables defined in the MIB file.
- Command-line tools—Miscellaneous tools and utilities that can be used on the command line of the ZephyrTel Mobile Messaging host machine.
- MGR—Web-based management application of the ZephyrTel Mobile Messaging product suite, which is used to manage the configuration of all ZephyrTel Mobile Messaging systems in the SMS network configuration.
- STV—Application that stores, processes, and display statistical output of all available ZephyrTel Mobile Messaging components for real-time and archived viewing.

2.3.4 Billing Space

The billing space is the interface toward the prepaid (IN) triggers and the ASN.1 formatted call detail records (CDRs) for any billing system or mediation device. ZephyrTel Mobile Messaging can produce SMSC- or MSC-compatible CDRs.

The following components may reside in the billing space:

- Prepaid Billing Controller (PBC)—The PBC provides the interface between the ZephyrTel Mobile Messaging components and any third-party IN or other prepaid billing systems. For IN, the Routers generate IN-triggers via SS7 to the IN billing systems.
- Billing System—The generic term for any system that performs charging of subscribers. Sometimes a mediation system is used; that is, a system that performs post-processing (reformatting) of CDRs as a preparation for the processing by the billing system.
- Fully SNMP manageable—Statistics counters, system status, settings, queues, and delivery schemes are all manageable via SNMP.

2.4 Node Discovery

The elements in the SMS network discover one another dynamically. This means that a new AMS, RTR, or HUB can be added to the SMS network without configuring the address of the new node on the existing nodes.

Addition of extra AMS nodes can provide additional redundancy by increasing the replication level, storage capacity, or delivery capacity without stopping or starting existing nodes.

The network discovery protocol is implemented over IP multicast. Other communication among nodes is all accomplished via IP, which allows the nodes in the SMS network to be connected by any IP network.

2.5 Intermediate Cache

When a message passes through the Mobile Messaging system and is then forwarded to an external SMSC, the system can store the message's state and certain parameters in a record in the Intermediate Cache (Icache). The Icache's contents represent the set of messages that are handled and/or stored on an external SMSC.

When the Mobile Messaging system forwards a message, it instructs the external SMSC to report back with the message's final delivery status: successful delivery, permanent failure, expiry, or deletion. The Mobile Messaging system then:

1. Uses the delivery notification to look up the corresponding record in the Icache
2. Applies the appropriate action, based on the status
3. Deletes the record from the Icache

The Icache stores the message state that is required to:

- Create a final delivery CDR

- Complete two-stage online charging via the PBC
- Route delivery notifications to the appropriate recipient

Icache records are stored in the AMS. You can select queues for different types of Icache records. The selected queue determines the records' validity periods.

The `amsstoragemode` parameter in the AMS semi-static configuration file controls whether the AMS acts as a message store and Icache, as message store only, or as an Icache only. The Icache settings are configured in the MGR.

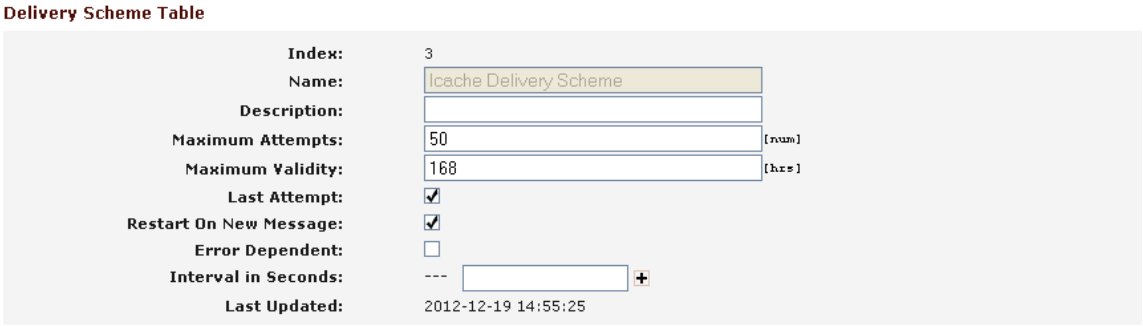
Note: The Icache requires that, when SMPP is used, the `deliver_sm` SMSC delivery receipt must include the `receipted_message_id` parameter.

To support the scenario in which the external SMSC's report about the final delivery status of a message does not reach the Mobile Messaging system, the AMS supports the expiration of Icache records. If an Icache record expires, the AMS signals this event to the RTR, which can then take appropriate action.

2.5.1 iCache Queue Settings

To configure the iCache, the following steps needs to be taken:

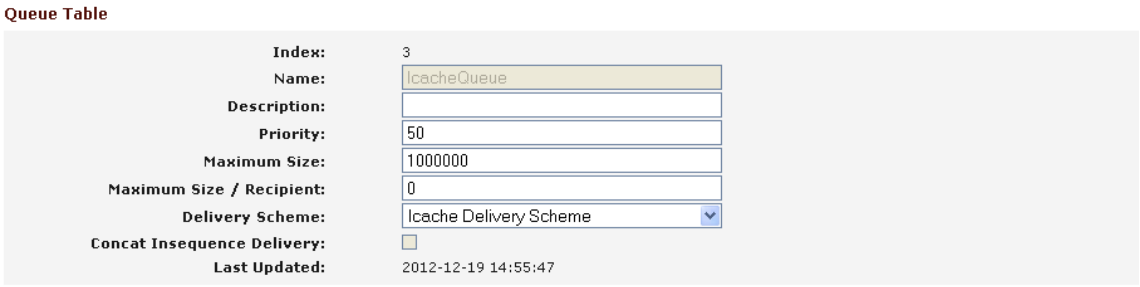
1. First configure the AMS delivery scheme for iCache as shown below:



   Make sure that the configured delivery scheme is activated.

   Note: The iCache functionality uses **Maximum Validity** of messages given in the delivery scheme. The configuration setting `amsmaximumvalidityperiod` and `amsdefaultvalidityperiod` will not be used.

2. Then configure the AMS queue table for iCache, as shown below:



Make sure that the configured queue table is activated.

Note: The maximum value of '**Maximum Size**' can be 40000000 (40 million).

When using iCache in combination with message store the total number of iCache records and stored messages together is limited to maximum storage size of the AMS.

2.6 Quality Characteristics

ZephyrTel Mobile Messaging provides carrier grade design and behaviour. This quality behaviour is a result of the ZephyrTel Mobile Messaging architecture and design.

The most important quality design aspects are:

- High 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 ZephyrTel Mobile Messaging product suite
- Flexibility, ensuring it is easily adjustable to changing market requirements
- Reliability, ensuring correctness, completeness, consistency, and no loss of data
- 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 ZephyrTel Mobile Messaging systems
- Usability, providing easy-to-use GUI and command-line access
- Traceability and audit-ability, providing the ability to diagnose all system activity
- Accuracy, ensuring correct billing and revenue assurance

These design aspects are based on ISO 9126 quality attributes and can be recognized when working with ZephyrTel Mobile Messaging systems.

2.7 Multi-Instance Support

Multi instance feature allows multiple ZMM 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 AMS.

Note: A separate LICENSE is required for each ZMM user.

Chapter 3

Platform Specification

Topics:

3.1 Introduction

This chapter discusses the specifications of theAMS platform, including the capacity and hardware platform requirements of a single AMS node.

The seamless scalability of ZephyrTel Mobile Messaging ensures that capacity and performance are easily upgraded by adding more AMS nodes.

3.2 Storage Capacity

A single AMS node can store between 1 and 40 million SMS messages and Icache records, depending on the hardware configuration and AMS license. This limitation includes all status reports, notifications, replica messages, and Icache records that must be stored on the AMS node.

There is no direct limit on the number of SMS messages and Icache records. You can indirectly limit each type by configuring specific queues and setting limits on the queues.

3.3 Throughput Capacity

The throughput capacity of the AMS mainly depends on the following factors:

- Type of DB storage engine used (i.e. Btree or Hash)
- Certain configuration parameter settings, particularly `amsmaxdeliveryrate` and `amsmaxreplicationrate`
- Network characteristics such as the message delivery success rate
- System hardware (Processor, RAM, Disk) configuration.

A single AMS node using the Hash storage engine for the DB can provide a peak performance of 1600 to 2000 SMS messages and Icache records per second, depending on the hardware configuration. If the Btree storage engine is used for the DB, then the peak throughput performance would be approximately 1600 to 1800 messages per second.

The `amsmaxdeliveryrate` parameter should be configured such that it is at least equal to the value of the expression given on the right-hand side below:

```
amsmaxdeliveryrate >= (estimated peak incoming traffic rate, i.e. messages per
                      second / average success rate for message delivery attempts)
```

However, for optimum performance it is recommended that `amsmaxdeliveryrate` is always configured approximately 10% higher than the minimum allowed value indicated above.

Also, the `amsmaxreplicationrate` parameter should be set as equal to the estimated peak incoming traffic rate.

For example, iIf the estimated peak incoming load is 2000 messages per second and the average success rate for delivery attempts is 80%, then you should set `amsmaxdeliveryrate="2750"` (i.e. 10% higher than the minimum value, which is 2000*100/80 = 2500).

Similarly, in the above example you should set `amsmaxreplicationrate="2000"`.

Note: The average success rates are typically different for MT and AT delivery attempts.

For MT messages, a typical first delivery success rate is about 80%. In such a case, only 20% of incoming MO-MT and AO-MT messages must be stored for further delivery attempts. Therefore, only one-fifth of the total mobile network capacity arrives in the AMS.

For AT messages the delivery success rate is typically much higher, about 98%-100%. This increases the average success rate for overall delivery attempts (i.e. for all messages) and reduces the effective incoming traffic load on the AMS even further.

3.4 Queue Capacity

There is no practical limit to the amount of user-defined queues, though the added total size of all queues can never exceed the maximum message storage capacity.

3.5 Delivery Scheme Capacity

Up to 200 delivery schemes can be defined, and each delivery scheme can contain up to 100 delivery intervals.

3.6 Hash Storage

Note: The configuration is for normal AMS usage with HDD disks. For high capacity configuration with SSD disks, please refer to section *Configuration for High Capacity* .

To achieve a better performance of the AMS it can be considered to use a HASH store type instead of the old btree.

To change this the following steps must be taken:

1. Shutdown the AMS.
2. Change `common_config` file.
3. Convert the master and replica databases.
4. Start the AMS.
5. OS related settings (section *OS Related Settings*).

3.6.1 Change `common_config` File

1. To ensure that the AMS is using a Hash storage type, the `amsdbstorageengine` configuration setting needs to be added and it needs to be set to `hash`:

   ```
   amsdbstorageengine="hash"
   ```

   This will ensure that the AMS uses a HASH storage type for the replica and master database.

2. In addition the following settings can be applied to achieve a better performance:

```
amsdbcheckpointsize="10000000"
amsdbcheckpointtime="480"
amsdbcachesize="2000"
amsschedulerholdofftime="1"
amsdbwritepause="5"
amsdbtransactionclustersize="200"
```

   Also, the `amsmaxdeliveryrate` and `amsmaxreplicationrate` parameters should be set as described in *Throughput Capacity*.

3.6.2 Convert the Master and Replica Databases

Before converting the master and replica database it is recommended to make a backup of the database:

1. The master database can be converted via the following command:

```
cd /dbamsstore/master/
db_dump /dbamsstore/master/MainDB |db_load -t hash -c \
   db_pagesize=16384 -c h_ffactor=48 -c h_nelem=3000000 -c db_lorder=4321 tmp_DB
mv tmp_DB MainDB
```

2. A replica database can be converted via the following commands:

```
cd /dbamsstore/replica/
db_dump /dbamsstore/replica/ReplicaDB_0 |db_load -t hash -c \
   db_pagesize=16384 -c h_ffactor=48 -c h_nelem=3000000 -c db_lorder=4321
tmp_ReplicaDB_0
mv tmp_ReplicaDB_0 ReplicaDB_0
```

   Note: Do the same for all the replica databases in the directory.

3.7 Hardware Requirements

Because the AMS is a disk I/O-intensive application, it is required that the AMS contain enough hard disks. For a production configuration, at least four disks in a RAID-1 configuration are required.

The AMS supports two types of disk configuration:

1. SAS disks of size 300GB and RPM must be 10K RPM or more (on HP GEN8 and GEN9 15K RPM)
2. Solid State Disks 200GB 12G SAS WRITE INTENSIVE SFF 2.5IN SC 3YR WTY. Supported on HP DL 360p G9 servers running RHEL OS.

The amount of main memory (RAM) also determines the amount of messages that can be stored. AMS systems require at least 1.75 KB per message stored. This figure consists of both internal buffers of the AMS and the AMS BerkeleyDB caches set by the `amsdbcachesize`. It should be added to the needs of other processes and the operating system

As an example, the following memory figures are needed for systems running an AMS only:

- 1,000,000 messages: 2 GB, including an `amsdbcachesize` of 350
- 3,000,000 messages: 6 GB, typically rounded up to 8 GB, including an `amsdbcachesize` of 999
- 6,000,000 messages: 12 GB, including an `amsdbcachesize` of 2000.
- 25,000,000 messages: 30 GB, including an `amsdbcachesize` of 16384.

- 30,000,000 messages: 34 GB, including an `amsdbcachesize` of 16384.

Note:

1. Due to rapid hardware evolution, please contact your ZephyrTel Mobile Messaging account manager for an up-to-date hardware recommendation if you plan to extend your AMS capacity.
2. Store capacity of more than 6 million messages is currently supported on RHEL OS only and requires Solid State Disks.

3.8 OS Related Settings

For RHEL7 the following settings are recommended:

3.8.1 Generic checks

Note: The user should login as `root` to apply these changes.

Before proceeding with the next optimization, you should first verify that the battery or capacitor for the disk cache is working properly. To do so, execute the following command:

```
/opt/hp/hpssacli/bld/hpssacli 'ctrl all show config detail' | egrep -i -e cache
 -e battery
```

If the system is working correctly, the sample output should be as follows:

```
   Cache Serial Number: PAAVPID10330J4B
   Wait for Cache Room: Disabled
   Cache Board Present: True
   Cache Status: OK
   Cache Ratio: 25% Read / 75% Write
   Drive Write Cache: Disabled
   Total Cache Size: 512 MB
   Total Cache Memory Available: 400 MB
   No-Battery Write Cache: Disabled
   Cache Backup Power Source: Batteries
   Battery/Capacitor Count: 1
   Battery/Capacitor Status: OK
```

The parameters with bold output above indicate that the battery or capacitor is installed and it is working correctly. In case you get a different output for the corresponding parameters, it means that either the battery/capacitor is not installed or it is installed but not working correctly. It is important that you resolve any battery or capacitor issues to prevent poor AMS performance.

3.8.2 AMS with SAS Disks

Note: The user should login as `root` to apply these changes.

1. First recommended change will be the increase of network buffers:

```
net.core.wmem_default = 2097152
net.core.wmem_max     = 2097152
net.core.rmem_default = 2097152
net.core.rmem_max     = 2097152
```

These settings can be changed with the following commands:

```
echo 2097152 > /proc/sys/net/core/wmem_default
echo 2097152 > /proc/sys/net/core/wmem_max
echo 2097152 > /proc/sys/net/core/rmem_default
echo 2097152 > /proc/sys/net/core/rmem_max
```

2. Ensure that '`noatime`' and '`barrier=0`' are set in the file system properties. This is particularly important for the `/dbamsstore` and `/dbamslog` disk partitions.

   Example:

   ```
   LABEL=/data /data ext3 barrier=0,noatime 0 0
   LABEL=/dbamsstore /dbamsstore ext3 barrier=0,noatime 0 0
   LABEL=/dbamslog /dbamslog ext3 barrier=0,noatime 0 0
   ```

   Reboot the system.

   Note: This change should not be applied to operating system related partitions.

3.8.3 AMS with SSD Disks

Note: These settings are supported on RHEL servers only.

1. `/dbamsstore` and `/dbamslog` disk partitions should be with ext4 fs
2. Change `/etc/fstab` to ensure that '`noatime`', '`barrier=0`','`data=writeback`' and `commit=1`' are set in the file system properties. This is particularly important for the `/dbamsstore` and `/dbamslog` disk partitions.

   Example:

   ```
   LABEL=/dbamsstore /dbamsstore ext4 barrier=0,noatime,data=writeback,commit=1 0
   0
   LABEL=/dbamslog /dbamslog ext4 barrier=0,noatime,data=writeback,commit=1 0 0
   ```

   Reboot the system.

   Note: This change should not be applied to operating system related partitions.

3. Verify that the "Estimated Life Remaining" of SSD disks should be at least 180 days.

   To do so, execute the following command:

   ```
   hpssacli ctrl slot=0 show ssdinfo detail | egrep -i -e 'Estimated Life Remaining'
    -e ' Total' -e 'physicaldrive'
   ```

   If the system is working correctly, the sample output should be as follows:

   ```
   Total Solid State Drives with Wearout Status: 0
   Total Smart Array Solid State Drives: 3
   Total Solid State SAS Drives: 3
   Total Solid State Drives: 3
      physicaldrive 1I:1:3
         Estimated Life Remaining based on workload to date: 16951 days
      physicaldrive 1I:1:4
         Estimated Life Remaining based on workload to date: 16195 days
      physicaldrive 2I:1:5
         Estimated Life Remaining based on workload to date: 11018 days
   ```

   The parameters with bold output above indicate the estimated life remaining of each of the SSD disks.

4. Add to `/etc/sysctl.conf`:

```
# Change SCTP heartbeat interval to 1 second
net.sctp.hb_interval = 1000
net.core.rmem_max=8388608
net.core.wmem_max=8388608
net.core.rmem_default=1310710
net.core.wmem_default=1310710

#VM Tuning
vm.dirty_expire_centisecs=200
vm.dirty_writeback_centisecs=100
```

5. For faster AMS startup, execute following command:

```
blockdev --setra 32768 /dev/sdb
```

3.9 Configuration for High Capacity

For the complete details about how to configure the SSD disks, please refer to the ZMM Installation Manual for RHEL7 chapter "Configure AMS for High Capacity".

3.9.1 Change the `common_config` File

The following settings can be applied to achieve a better performance:

```
amsnumberofreplicas="1"
amsmasterstoragetype="nonvolatile"
amsreplicastoragetype="nonvolatile"
amsstatusupdatemaster="always"
amsstatusupdatereplica="never"
amsdefaultvalidity="720"
amsmaximumvalidity="720"
amsdbstorageengine="hash"
amsdbwritepause="30"
amsdbmaxsequentialwrites="16"
amsdbcachesize="16384"
amsdbcheckpointsize="9500000"
amsdbcheckpointtime="1800"
amsmaxdeliveryrate="1500"
amsmaxnumberofmessages="25000000"
```

Note:

1. `amsmaxnumberofmessages` must be set to max AMS messages.
2. `amsmaxdeliveryrate` must be set as per the customer network.
3. The above settings are applicable for RHEL servers only. A capacity of more than 6 million is supported on RHEL only.

Chapter 4

Functional Overview

Topics:

4.1 Introduction

The AMS provides store-and-forward functionality in an SMS network. The AMS is a high-speed storage medium for SMS messages that provides an intelligent mechanism to deliver messages to their destinations.

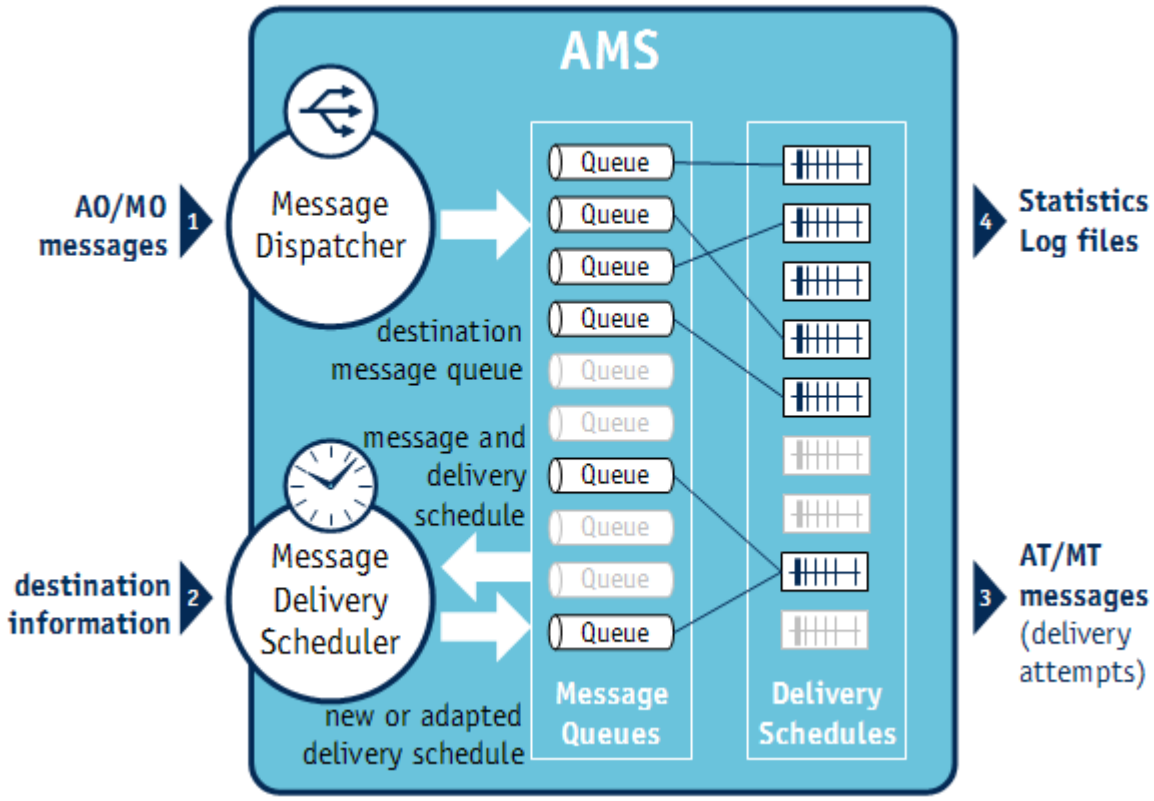


Figure 3: Message dispatcher and delivery scheduler

As depicted above, the AMS message dispatcher handles incoming application-originated (AO) and mobile-originated (MO) traffic. The message dispatcher is responsible for placing messages in the proper message queues, based on the defined queue settings. RTRs may overrule the message dispatcher's queue choice by indicating a specific queue for a message.

Once a message is in the queue, the message delivery scheduler monitors it. The message delivery scheduler is responsible for delivering a message at a specific time, based on the active delivery schedule for that message.

Note: Each message queue always has one delivery scheme attached to it, which may be altered based on the destination information.

The AMS delivers application-terminated (AT) and mobile-terminated (MT) messages via a RTR.

Optionally, the message delivery scheduler can also adapt the delivery schedule for a message; for example, if the message destination is temporarily not available. This process is depicted in the diagram above in flow 2 (destination information).

4.2 AMS Feature Summary

The AMS provides the following features:

- Reliable high-capacity message storage - A single-end AMS node can store up to 1 or 40 million SMS messages, depending on the hardware configuration.
- Multiple message interfaces - The AMS can accept SMS traffic from RTRs directed toward the mobile network and directed toward the application interface via the HUB (using SMPP, UCP, and CIMD2). For system management and statistics, the SNMP interface is available.
- True message prioritisation - The AMS allows the definition of priority queues for messages that will always be delivered before any other messages. It also allows definition of low-priority queues for messages that will be delivered in the background when there is capacity available (always after any higher priority messages).

  For example, a high-priority queue can be defined for an emergency message application, stock information, or premium customers. The “season’s greetings” messages can be set to low priority.
- Prioritized throughput control - In a peak traffic situation, the AMS will dynamically give high priority to the acceptance of incoming messages and will scale down replication and, eventually, the delivery rate.
- Multiple queues - The AMS includes automatically generated queues for standard traffic and operator-defined message queues that require specific delivery schemes, queue size, or priorities.
- Flexible delivery schemes - Whereas a traditional SMSC only allows one retry scheme for standard traffic, the AMS allows for differentiation among delivery schemes based on the queue that the messages are in. This functionality allows assignment of a specific retry scheme to be used for certain subscriber groups, specific date and time periods, and so on.
- Optimized delivery - Error-dependent delivery schemes take network errors into account. These schemes are especially important during busy hours when people are sending many messages to one another at a specific date and time.
- Real-time queue viewing - The Queue Query tool is delivered with the AMS as part of the Manager. The Queue Viewer provides real-time insight in the current status of the queue, including how many messages are in the queue, the average amount of deliveries, and average message details.
- Queue management - Queue content can be monitored and managed by purging messages for a specific destination, reassigning priorities, and adding or removing queues. Queues can also be queried to inspect waiting messages for a specific subscriber, which is typically used for Customer Care purposes. Alternatively, an XML interface is also available for a Customer Care application to directly query and manipulate message queue content, such as to view which message are waiting for delivery.
- Detailed statistics - As with all ZephyrTel Mobile Messaging products, the AMS provides an abundance of statistical counters that are made real-time available via SNMP or via the Web-based graphical Statistics Viewer (STV).
- Parallel AT deliveries - It is now possible to configure applications based on the submitted service type. The AMS enables the enhanced parallel delivery method for such applications, if the "`service type`" matches the one in the notification. This will initiate the configurable amount of initial deliveries. The rate of injecting new message is also configurable. The feature is applicable to both AT delivery in case of AO-ST-AT and AT notification in case of AO-ST-MT, AO-MT and AO-ST-AT. This feature is only applicable to SMPP protocol. Refer to section *amsparalleldelivery Entity* for more details.

AT Extrusion feature uses the recipient address in the incoming message as the recipient address to create the recipient queue. Therefore, if this feature is enabled, multiple messages with different recipients in the incoming message, but directed to the same terminating application, will have different queues instead of a single queue for the same configured terminating application. This feature is only applicable when AT parallel delivery is enabled.

AT Extrusion reuse last interval feature is an extension of the AT Extrusion feature. This feature reuse the last configured interval until the message reaches its validity period. This feature is only applicable when the *amsatextrusion* is enabled. Refer to the semi-static parameter *amsatextrusionreuselastinterval* to enable the feature.

4.3 Routing Paths

The AMS provides reliable and persistent store-and-forward functionality for SMS messages. The message flow can be controlled in the RTR by defining the appropriate routing rules to enforce specific routing paths for a message. The diagram below illustrates the concept of the RTR's MO/AO routing engine, which defines filters and parameters used and the actual routing path.

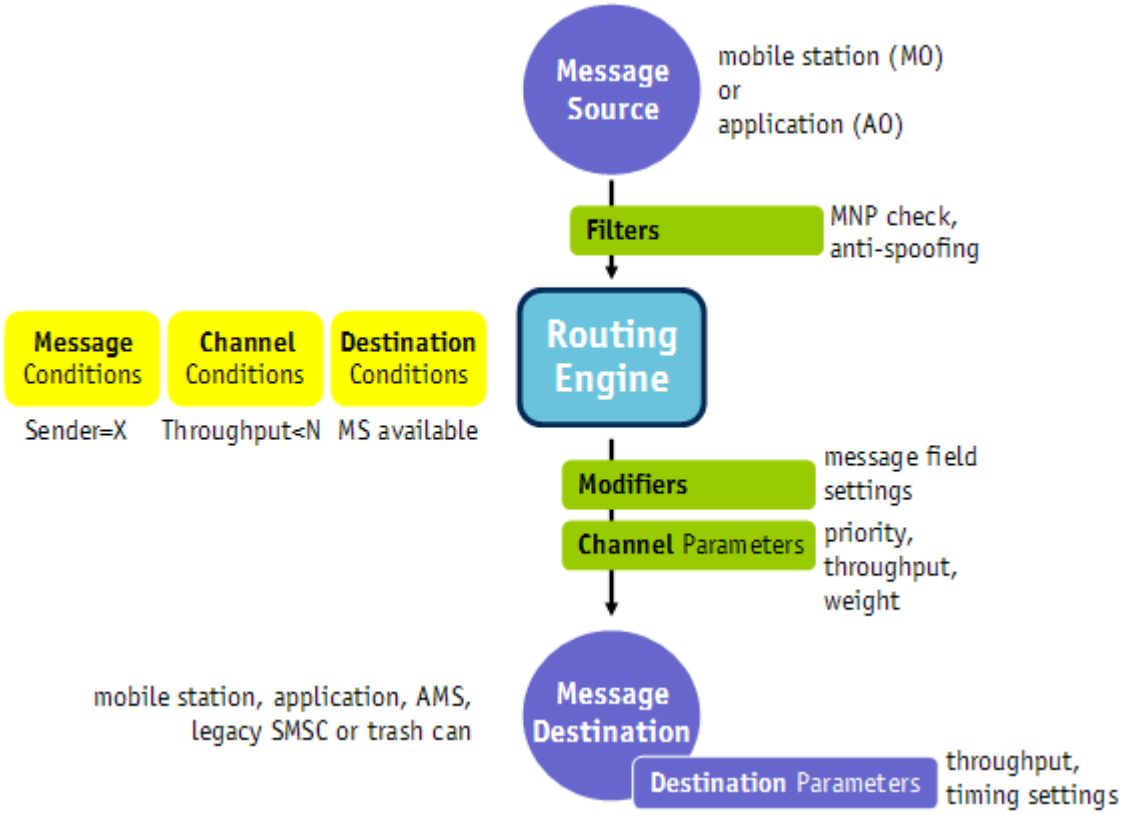


Figure 4: AMS rule context

4.4 Message Flow

When a routing rule instructs the RTR to route a message to the AMS, the following steps are taken:

1. The RTR selects the appropriate AMS node as the Master (the AMS node primarily responsible for performing delivery attempts for that message) based on the recipient and/or originator address for MT or AT messages.
2. The RTR forwards the message to the AMS (over SCTP)
3. After the AMS confirms the successful storage action, the message is acknowledged toward the originator (optionally indicating the assigned unique time stamp)
4. The AMS replicates the message for reliability purposes (once or multiple times)

Now the message is stored safely in the AMS, and message delivery according to the defined or selected delivery schemes begins:

1. The AMS performs one or more delivery attempts according to the corresponding delivery scheme
2. These delivery attempts are routed via the RTR toward the message's destination (either a mobile station or an application)
3. If the delivery is successful, the AMS triggers a notification or status report (when requested)
4. The old replicas of the messages are purged from all AMS nodes
5. The RTR creates a CDR upon successful delivery (or upon final attempt) and the message is removed from the delivery queues
6. All statistical counters are updated accordingly

The AMS continuously determines whether queued messages should be scheduled for delivery.

4.5 Message Queues and Delivery Schemes

The AMS provides a queue-based storage system. Each incoming SMS message is assigned a message queue, and the message will be delivered according to the delivery schedule that is attached to that queue. Delivery schedules are defined separately and can be set to any number of delivery attempts with variable delivery intervals.

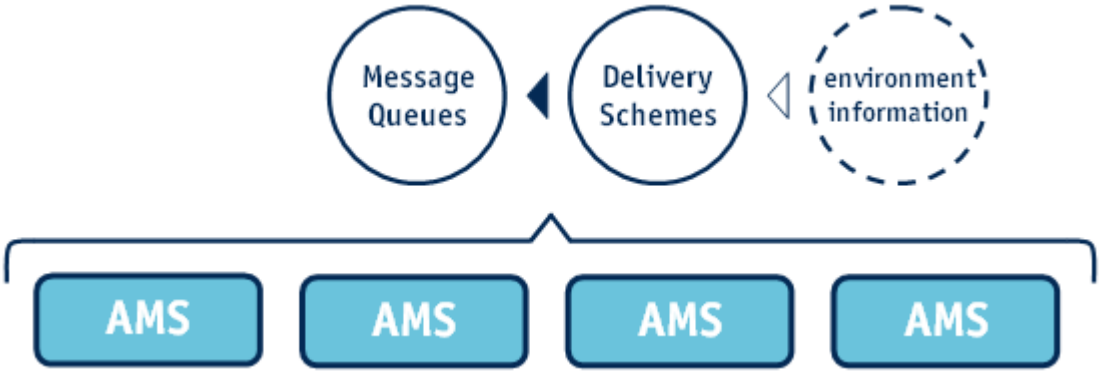


Figure 5: Delivery schedule

Message queues are defined system-wide and are distributed across all available AMS nodes in the system. As the diagram above illustrates, each message queue always has a delivery scheme attached to it.

Note: A service centre alert or application alert message for a certain recipient causes an immediate delivery attempt for this recipient.

4.6 Notifications and Status Reports

The AMS can trigger a status report (for MO messages) or a notification (for AO messages), if the originator requests. The RTR will generate the applicable status report or notification when the AMS triggers it.

The following notifications are supported:

- Buffered—Indicates that the message was successfully stored in the AMS
- Delivered—Indicates that the message was successfully delivered to its destination (typically a Mobile Station or an application)
- Deleted—Indicates that the message was deleted or blocked
- Expired—Indicates that the message validity period has expired and the message has been deleted from the AMS systems

If the delivery of the status report or notification fails, it is queued in the AMS for later delivery. Status reports and notifications are stored in the applicable queue just like regular messages.

Note: In the case of concatenated messages, status reports and notifications could be generated for each message segment.

Note: The RTR can generate MAP Phase 2 status reports or Phase 1 status reports if requested using configured scan tags.

4.7 Enhanced Character Conversion

4.7.1 Introduction

The Enhanced Character Conversion functionality allows applying customer specific coding rules on stored messages. The conversion can be applied both on single messages and concatenated messages. It therefore relies on the later described in the *In-Sequence Message Delivery* feature. Single messages could be extended to concatenated and concatenated could increase the amount of segments. A converted message with more than 1 segments will always get the 8-bit concatenated info in the UDH with IEIE 0x0.

4.7.2 Triggering Conversion

The conversion is triggered in case all of the following pre-requisites:

- A queue needs to be assigned to a proper Unicode Character Conversion table. See MGR OM for more information.

- It applies only to specific DCS values (UCS2 like). See the RTR OM for the full list of DCS values.
- The UserDataHeader may only contain IEI 0x0 (concatenated 8-bit) or IEI 0x8 (concatenated 16-bit).
- A conversion could lead to the same message text. If that is the case, nothing is converted.

4.7.3 UDH Concat 16-bit to 8-bit Conversion

This feature is upon special request of the customer and enabled by the Enhanced Character Conversion feature (thus CharacterMap defined in queue). It will translate the UDH with IEI 0x8 (16-bit concatenated) into a an IEI 0x0. The high byte of the message reference is dropped. A converted message will always have

This UDH conversion is only triggered in case all if the following pre-requisites:

- A queue needs to be assigned to a proper Unicode Character Conversion table. See MGR OM for more information.
- The UserDataHeader has only IEI 0x8 (16-bit concatenated).

4.7.4 Unsplit Surrogate Pairs

Existing behavior of UCC conversion may separate the pairs into 2 segments by fully using the space in the segment. This leads to issues in certain handsets (android phones where the software version is less than 6.0) from not being able to display messages when multi Unicode characters/surrogate pairs are split into 2 segments.

This feature prevents certain Unicode characters (surrogate pairs and multi-Unicode characters) on the SMSC from splitting into 2 segments of a concatenated message, thus this feature will move the whole pair of Unicode characters to one of the segments. Consequently, the whole surrogate pair is unsplit.

This feature, "Unsplit Surrogate Pairs", is license controlled and configured on the AMS and the MGR.

It also allows configuration of multiple Unicode characters using the *pairedunicodecharlist* semi-static parameter. For example, multiple Unicode characters, 0x0023 0x20E3.

For more information about configuring this feature, refer to the MGR Operator Manual (Section 8.3 Configuring Message Queues, Step 13).

The following points should be considered for this feature:

- **Adjustment regardless of the value of entries of UCC:** Segments boundary adjustment is executed regardless of UCC entries values when a surrogate pair character is split into two consecutive segments. This means that even if Emoji Conversion is not executed, this adjustment shall be executed.
- **Consecutive segment not received:** When the former of a split surrogate pair is received, but the latter of it is not received (and vice versa), a part of a surrogate pair remains split as it is.
- If "UCC conversion" and this feature are enabled the result will be free of split surrogate character pairs.
- Pre-requisites of this feature are the same as those mentioned in *Triggering Conversion*.

4.7.5 Unique Service Center Timestamps for Additional Segments

If the AMS applies the enhanced character conversion to a single or concatenated message and generates additional segments, the unique service center timestamp shall be maintained while delivering the additional segments.

This section presents the explanations and limitations of maintaining the unique service center timestamps for additional segments:

- If AMS receives the messages from the RTR for submission, either as a single message or concatenated messages, and the enhanced character conversion is applicable for the received messages, then the AMS will reserve the next few unique service center timestamp (i.e. sc_timestamp) for the additional segments.
- The reservation of few unique 'sc_timestamp' is based on either 'total no. segments' or 'single message', which tells the AMS that how many next 'sc_timestamp' need to be reserved.
- If the message is single and the enhanced character conversion is applicable, then next 4 (i.e. 4 x single message) unique sc_timestamp shall be reserved for the additional segments.

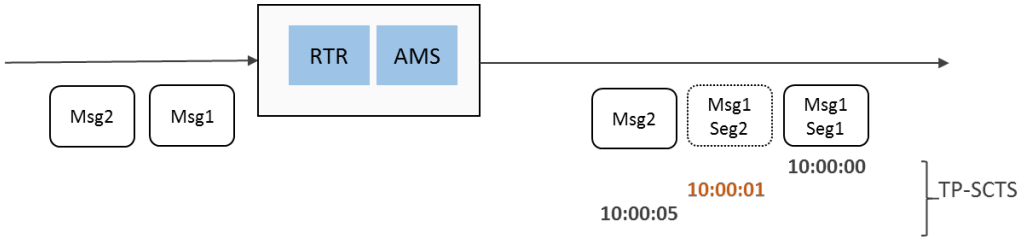


Figure 6: Unique Timestamp when Single Message Is Converted into 2 Segments

In the above figure, 4 unique sc_timestamp (i.e. 10:00:01 - 10:00:04) will be reserved. The timestamp 10:00:01 will be used for one additional segment and the rest of the reserved timestamps will be released when the message gets successful/permanent failure/expired before receiving the second message (i.e. Msg2), Otherwise, the timestamp range 10:00:02 - 10:00:04 will never be used.

- If all segments of the concatenated messages are received within the configured timeout for the AMS message queue and the enhance character conversion is applicable, then the next 4 X no. of segment' unique sc_timestamp shall be reserved for the additional segments.

  For example, if the number of segment is 2, the next 8 (i.e. 4 x 2 segments) unique sc_timestamp shall be reserved for the additional segments.

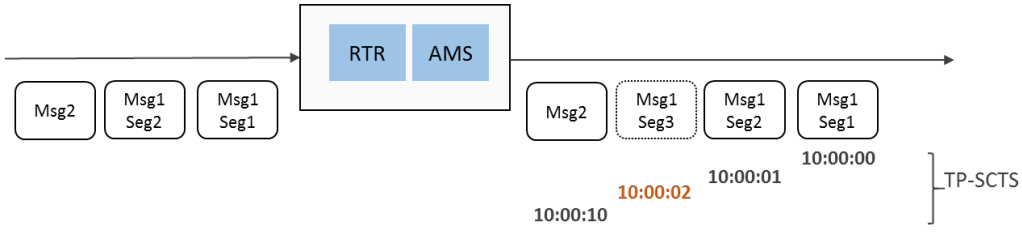


Figure 7: Unique Timestamp when 2 Segments Concatenated Message Is Converted into 3 Segments

In the above figure, 8 unique sc_timestamp (i.e. 10:00:02 - 10:00:09) will be reserved. 10:00:02 will be used for one additional segment and the rest of the reserved timestamp will be released when

the message gets successful/permanent failure/expired before receiving the second message (i.e. Msg2). Otherwise, the timestamp range 10:00:03 - 10:00:09 will never be used.

Note: The character conversion will not be applicable if all segments are not received within the configured time for the AMS message queue.

- The semi-static parameter `amsmediatedservicecentretimestampsenabled` must be set to 'true' on RTR to maintain the unique sc_timestamp generation.
- The sc_timestamp for additional segments will be one higher than the current highest timestamp stored for that recipient.
- Another message to the same recipient will not disrupt the processes and uniqueness of sc_timestamp.
- While encoding the sc_timestamp for the additional segments over MXP interface for delivery, AMS shall use the reserved unique sc_timestamp and assign the same for the additional segments.

  Note: Make sure that sc_timestmp for the stored (i.e. original) message should not be changed. Please refer to *Figure 7: Unique Timestamp when 2 Segments Concatenated Message Is Converted into 3 Segments* .

- When the AMS has delivered all converted messages (i.e. original + additional) and realized that some of the reserved sc_timestamps are unused, and there are no extra messages present in the AMS for this recipient after this just converted message, only then it will set the next 'sc_timestamp' to the 1 higher of the highest just transmitted sc_timestamp or to the current time whichever is higher.

  Note: While reducing the timestamp make sure that it will not create a duplicate sc_timestamp (uniqueness of sc_timestamp must be maintained).

Limitations:

- There is a possibility that some of the reserved timestamps are lost and will never be recovered. Please refer to *Figure 7: Unique Timestamp when 2 Segments Concatenated Message Is Converted into 3 Segments* and its explanation.
- The reserved timestamp can be lost on the AMS restart since this reserved timestamp is maintained in the 'Recipient buffer' (i.e. in memory) not in the Databases. To prevent issues after AMS restart, it sets the next_sc_timestamp for the recipient as well. Reserved timestamp is recalculated when AMS rereads the database.

4.8 In-Sequence Message Delivery

The In-Sequence Message Delivery will only be used in conjunction with the Enhanced Message Conversion functionality because the AMS needs to have the full concatenated message information before it can do any conversion.

The In-Sequence Message delivery will allow the AMS to deliver messages from the same originator to a single recipient (that is, concatenated messages) in sequence, provided these messages are stored in the same queue. The AMS guarantees that the concatenated messages are delivered in the correct order toward the destination mobile-station(MS) or application.

Because the AMS provides true message prioritization, high priority messages always have precedence over low priority messages; this is also true for concatenated messages.

4.8.1 Concatenated In-Sequence Message Delivery

To further increase the message delivery success rate for concatenated messages, the AMS provides functionality to ensure in-sequence delivery of concatenated messages with a minimum delay in between.

The **Character Conversion Map** and **Concat In-Sequence Delivery Timeout** settings in the MGR control this behavior:

- When the character conversion map is chosen then the In-Sequence Delivery Timeout has to be set to a non-zero value. The default value will be 15, which means the next segment has to be received within 15 seconds. Otherwise, the already received message will be processed as a single message.
- Non concatenated messages will always be handled as single messages.
- When disabled, the AMS simulates regular SMSC behavior. This is the default setting.

4.8.2 Recommendations

The following practices are recommended when using in-sequence message delivery:

- Make all segments of the concatenated message use the same message queue to ensure in-sequence and uninterrupted delivery of concatenated messages.
- In the RTR, configure a routing rule that first stores messages in the AMS, then performs a delivery attempt. This will ensure that the AMS receives all segments of a concatenated message before it begins delivery of the first segment of the message.

4.9 Message Distribution

To achieve an approximately equal load on all AMS nodes, the RTRs will evenly distribute messages across all available AMS nodes.

The distribution of MT messages is based on the recipient address, to enable assignment of unambiguous timestamps. All RTRs use the same distribution scheme, which ensures that each message for a certain recipient is stored in a particular AMS node, independent of the RTR handling that message.

The distribution of AT messages is based on both originator address and recipient address.

Maximum Number of Store Request per Second

The maximum number of store request messages that a (set of) RTR(s) may send per second to a (set of) AMS(s) is configured on the RTR using the semi-static `maxstorerequestspersecond` attribute (refer to RTR Operator Manual). Setting this threshold helps preventing the RTR(s) from flooding AMSs with messages to be stored. This only works reliably if all RTRs have this attribute set to the same value.

Example: Assume that there are 2 RTRs and 2 AMSs, and that the `maxstorerequestspersecond` attribute is set to 500. Then, each of the two RTRs is restricted to distribute 250 SMs per second over the two AMSs. If one AMS goes down, each RTR will send all 250 SMs to the remaining AMS.

4.10 Message Validity and Expiry

The AMS assigns a unique timestamp to messages that it accepts. As of the first intended delivery attempt, the AMS assigns the appropriate validity period to the message.

The validity period can be based on:

- For MO messages:
  - Specified in number of hours
  - Specified to use the AMS maximum validity period
- For AO messages:
  - Specified in an absolute date and time
  - Not specified, resulting in the AMS default validity period

The AMS will schedule delivery attempts for the accepted messages according to the appropriate delivery scheme. If the message is not yet successfully delivered after the validity period expires, the AMS will delete the message. The RTR will then generate a CDR and a status report or notification (if one has been requested for this message).

The default validity period and the maximum validity period are configurable parameters in AMS, within the range of 1 to 2232 hours (equal to 3 months).

Note: If the single shot indicator is set for a message, it expires after the first delivery attempt.

4.11 Message Expiry on Completion of Delivery Intervals

Normally the AMS considers a message as expired once all delivery intervals configured for the relevant delivery scheme have been completed, even if the message validity period has not yet elapsed. All such messages are removed from the AMS immediately, without any further retries.

However, the AMS now also allows a configurable option to prevent messages from getting expired on completion of delivery intervals. If the necessary configuration parameter (**Retain up to Validity**) is enabled for a delivery scheme, then all messages following that scheme will be retained in their respective delivery queues once all configured delivery intervals have been used up. In such a case the retained messages will not expire until their respective validity periods get elapsed or the number of delivery attempts exceeds the configured maximum number of attempts for the delivery scheme.

When all messages for a given recipient are retained in the delivery queue, the AMS will not attempt any further periodic retries for such messages (since no delivery intervals are left) but will wait until the message having the earliest validity period expires. However, in case a new/replaced message becomes available for the same recipient (in the same queue) or an alert is received for the same recipient in the meantime, then the AMS will trigger a delivery attempt for the first available retained message.

Refer to *Delivery Scheme and Message Retention* for more details. For information about configuring delivery schemes, refer to the MGR Operator Manual.

4.12 Deferred Message Delivery

Messages can be submitted to AMS with an indication to defer the first delivery attempt. Depending on the properties of the interface used to submit the message, the defer period can be based on:

- MO messages—Specified in number of hours using configured scan tags (relative to the receipt of the message)
- AO messages—Specified in an absolute or relative date and time

The validity period of a deferred message starts at the scheduled time of the first delivery attempt. The deferred delivery time must be within the:

- Message's validity period, and
- AMS's maximum deferred delivery time (`amsmaximumdeferreddelivery`)

Otherwise, the message is rejected.

Note: The routing path applied to deferred delivery messages should not include a first delivery attempt (FDA) performed by the RTR.

4.13 Prioritised Throughput Control

In peak traffic situations, the AMS dynamically gives high priority to the acceptance of incoming messages. These are handled immediately, while message delivery and replication are throttled to ensure that as many incoming messages as possible can be accepted (i.e. received from RTR and stored in the database).

Priority is as follows:

1. Acceptance
2. Delivery
3. Replication

Note: The message delivery and replication rates are always subject to configured upper limits, as specified by the parameters `amsmaxdeliveryrate` and `amsmaxreplicationrate`.

Chapter 5

Reliability and Availability

Topics:

5.1 Introduction

The AMS is designed to be highly reliable and to never lose a message once it is positively acknowledged.

AMS reliability is optimized by:

- Transactional database capabilities to ensure persistent storage and integrity
- Acknowledgement toward the RTR after commit to disk
- AMS node redundancy and failover (a minimum configuration should consist of two nodes)
- Message replication, as every message is replicated to a configured replica on another node
- Database checkpoints and log records to rebuild the database and queues after restart
- Checksum verification on message replication

Furthermore, the AMS is designed to gracefully handle contingency situations that pertain to related network elements, the related network, the AMS hardware platform, or any related software component.
.

5.2 Message Replication

The AMS uses a replication mechanism to achieve a very high and scalable redundancy. The RTRs distribute messages and Icache records equally across the available AMS nodes. An incoming message is stored on the receiving AMS node (the master) and in *N* other AMS nodes (the replica(s)). The number *N* can be defined by the operator and determines the grade of redundancy required; typically *N* will be 1, meaning one backup of each message. The master AMS node is responsible for assigning the message an identifier and forwarding it to the replica(s).

Note: The AMS instance(s) to which a message or Icache record would be replicated is determined independently for each message or record, when it is first stored in the master AMS database. Hence in case one or more AMS instances become unavailable, new messages or Icache records that are subsequently stored in the master AMS may be replicated to different AMS instance(s) as compared to the existing messages or records (which were stored previously).

In an environment where the AMS is being used as both a message store and an Icache (transaction store), the AMS will not replicate messages or Icache records between incompatible nodes. For example, if AMS-1 is only a message store and AMS-2 is only an Icache, records will not be replicated between them. However, if AMS-1 and AMS-2 are both message stores and Icaches, records will be replicated between them.

This example illustrates the typical configuration of two AMS nodes with three system-wide message queues (queue 1, 2 and 3). Each AMS node serves as both master and replica for different queues: AMS-1 is master (M) for queue 1 and queue 2 and replica (R) for queue 3.

Note: A queue will never be just a master or just a replica. A queue will contain both master and replica messages, depending on which AMS the RTR chooses for a specific message.

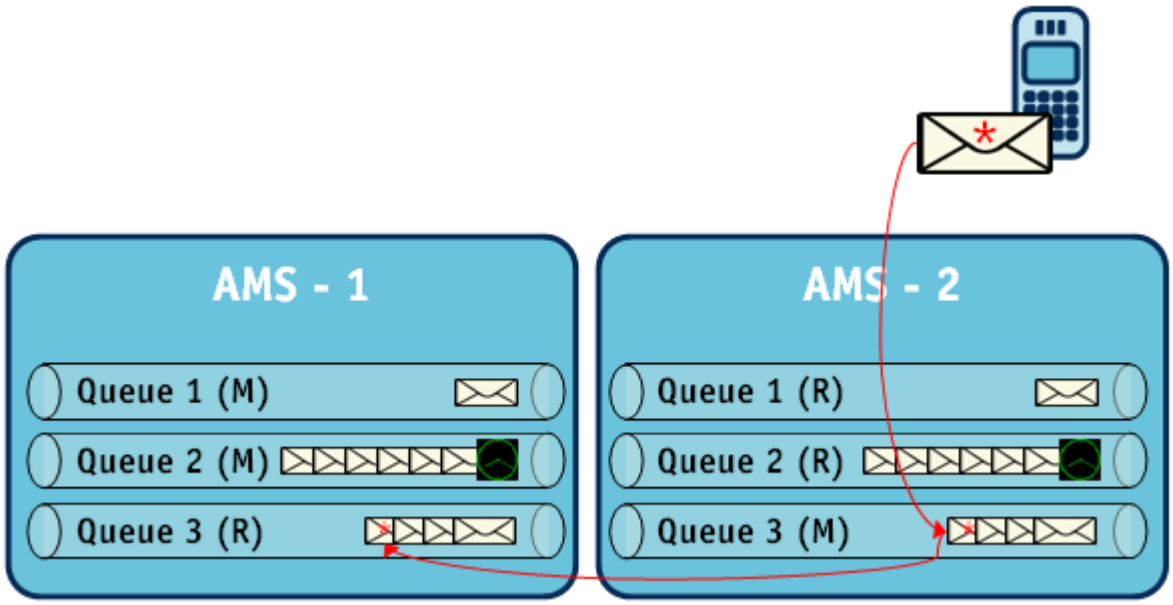


Figure 8: Sample AMS configuration with two nodes

If a replica node is temporarily unavailable, the message will be replicated to it when the node becomes available again (delayed replication). The delayed replication functionality also supports concatenated messages.

If more than two AMS nodes are present, the AMS system can be configured to store multiple replicas of the same message for additional redundancy (if the operator requires it).

5.2.1 Message Replication in different types of AMS Deployment

AMS can be deployed in a network/site in the following ways:

- Standalone Deployment, i.e. Single instance of AMS running on each physical server.
- Multi-instanced Deployment, i.e. Multiple instances of AMS running on each physical server.
- VM-based Deployment, i.e. Single instance of AMS running on a VM, with multiple VMs running on each physical server.

Note: For Multi-instanced or VM-based deployments, it is recommended that the same number of AMS instances/VMs with mutually compatible capabilities are installed on each physical server.

AMS Replication algorithm is designed to ensure that an AMS instance and any of its replicas are not co-located on the same physical server. Moreover, since no AMS instance can ever be its own replica, replication is not performed at all in case only a single AMS instance is available.

Note that the software also attempts to assign the replicas in such a manner that each AMS instance receives replication traffic from the same number of other AMS instances. However, it is still possible that under certain scenarios a given AMS instance may end up serving as the replica for more number of AMS instances (as compared to its peers); this is referred to as an "overloaded replica". Such a scenario is more likely to occur when:

- There is a “mixed” deployment configuration, e.g. some AMS instances running on VM and some others running directly on physical server(s).
- The capabilities of the deployed AMS instances are not mutually compatible.

- The number of AMS instances/VMs on each physical server are different.

The following sections explain the replication mechanism for each of the deployment scenarios mentioned above.

5.2.2 Replication for Standalone Deployment

In this scenario, single instance of AMS is running on a physical server.

Replication algorithm is as follows:

1. AMS list is sorted based on Host-ID of the AMS node.
2. Each AMS replicates to the AMS running on next Host-ID in the sorted list. If the target AMS is down (temporarily unavailable) then the next AMS in the sorted list is selected, if it exists.
3. If the configured Number of Replicas is more than 1, then repeat step 2 till either the requisite number of replicas are selected or no more physical servers are left.
4. If no other physical server exists (after step 2), then replication will not happen.

Note: Do not set `amsPropPhyServerId` for this scenario; otherwise replication behavior might get impacted.

Example:

Consider the following scenario where there are four physical servers each running one AMS:

- Physical Server 1 , Host ID A, AMS ID 1
- Physical Server 2 , Host ID B, AMS ID 3
- Physical Server 3 , Host ID C, AMS ID 2
- Physical Server 4 , Host ID D, AMS ID 4

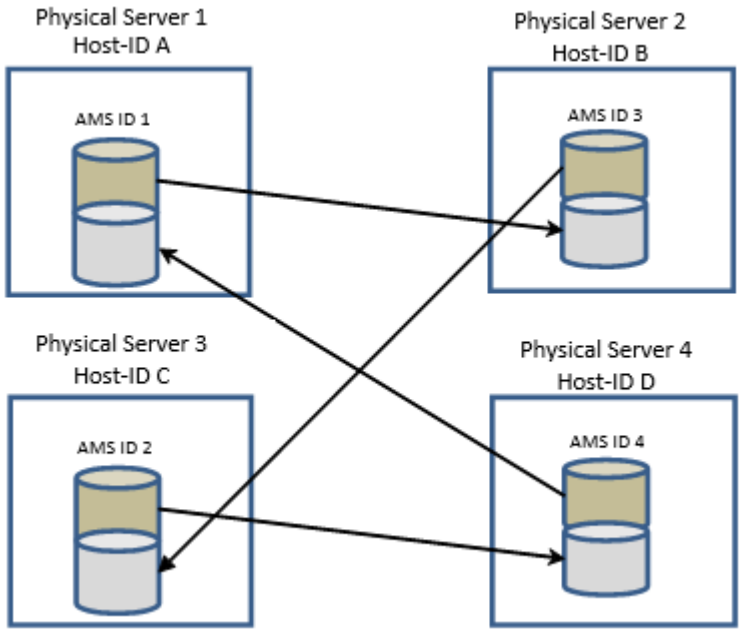


Figure 9: Standalone AMS Replication (Number of Replicas =1)

If the number of Replicas is configured as '1', the replication logic in this case will be as follows:

- To calculate replica, AMS list is sorted based on Host-ID:
  - A: AMS ID 1
  - B: AMS ID 3
  - C: AMS ID 2
  - D: AMS ID 4

- Replication will be:
  - AMS ID 1 will replicate to AMS ID 3
  - AMS ID 3 will replicate to AMS ID 2
  - AMS ID 2 will replicate to AMS ID 4
  - AMS ID 4 will replicate to AMS ID 1

Note: Each AMS will be replica of one AMS.

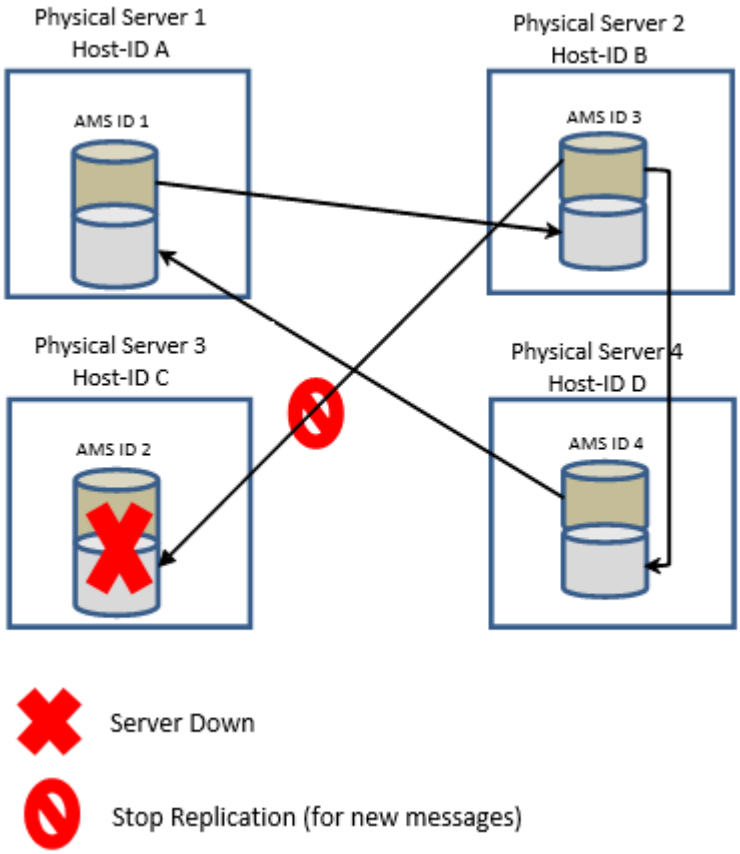


Figure 10: Standalone AMS Replication (Number of Replicas=1) when one AMS is down

- If AMS ID 2 goes down, then replication will be:
  - AMS ID 1 will replicate to AMS ID 3
  - AMS ID 3 will replicate to AMS ID 4
  - AMS ID 4 will replicate to AMS ID 1

Note: AMS ID 4 will act as active replica for AMS ID 2 and will take over the responsibility of delivering all messages on behalf of the failed instance as well as updating the database.

- When AMS ID 2 comes up again, the initial replication pattern will get restored:
  - AMS ID 1 will replicate to AMS ID 3
  - AMS ID 3 will replicate to AMS ID 2
  - AMS ID 2 will replicate to AMS ID 4
  - AMS ID 4 will replicate to AMS ID 1

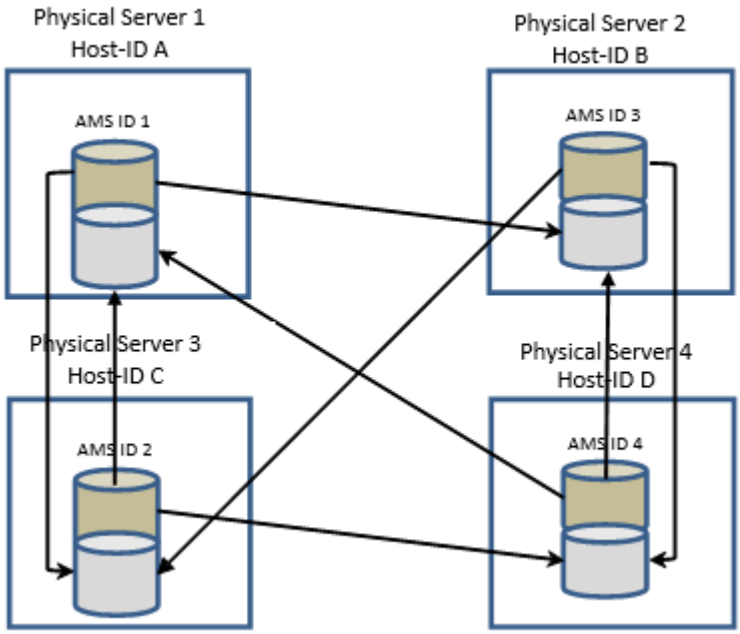


Figure 11: Standalone AMS Replication (Number of Replicas=2)

As a further example, consider the following replication logic when the number of Replicas is configured as '2':

- AMS list will be sorted based on Host-id (identical to the previous case with number of Replicas = 1).
- Replication will be:
  - AMS ID 1 will replicate to AMS ID 3 and AMS ID 2
  - AMS ID 3 will replicate to AMS ID 2 and AMS ID 4
  - AMS ID 2 will replicate to AMS ID 4 and AMS ID 1
  - AMS ID 4 will replicate to AMS ID 1 and AMS ID 3

Note: Each AMS will be replica of two other AMS.

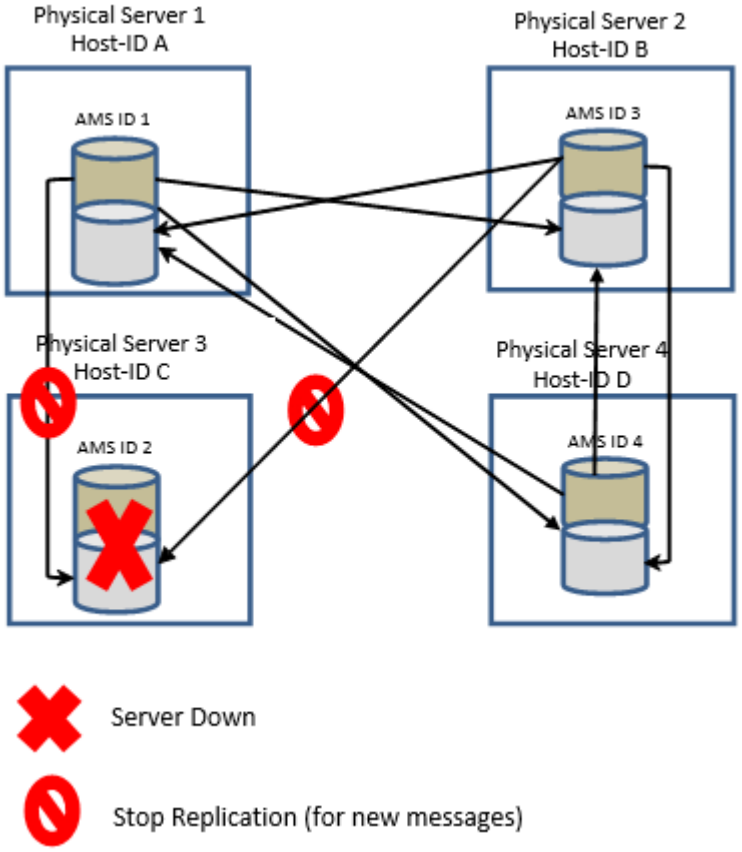


Figure 12: Standalone AMS Replication (Number of Replicas=2) when one AMS is down

- If AMS ID 2 goes down, then replication will be:
  - AMS ID 1 will replicate to AMS ID 3 and AMS ID 4
  - AMS ID 3 will replicate to AMS ID 4 and AMS ID 1
  - AMS ID 4 will replicate to AMS ID 1 and AMS ID 3

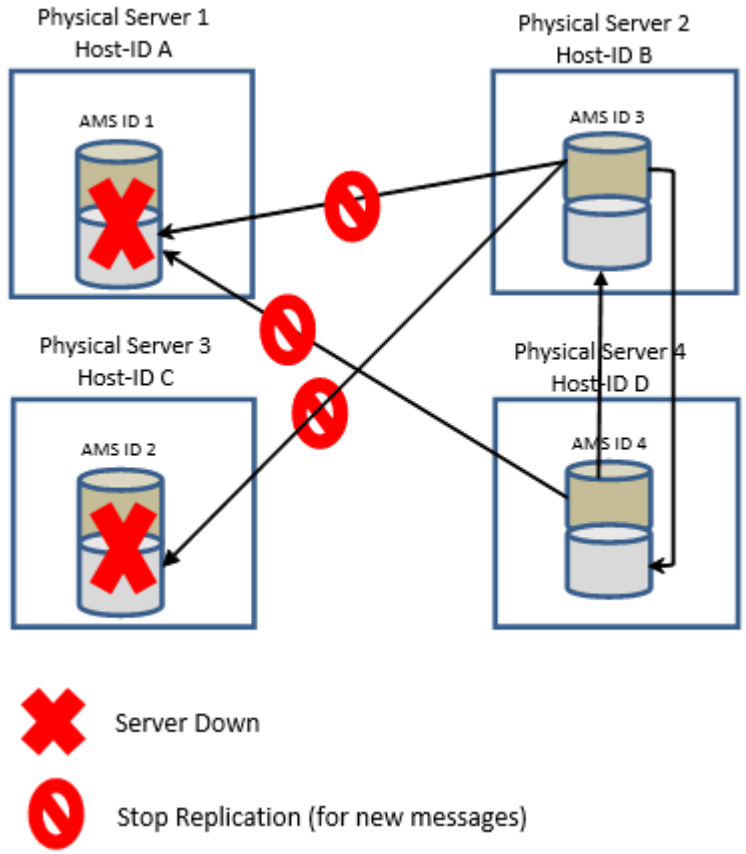


Figure 13: Standalone AMS Replication (Number of Replicas=2) when two AMS are down

- If, on the other hand, both AMS ID 1 and AMS ID 2 go down then the remaining two AMS instances (i.e. AMS ID 3 and AMS ID 4 in this case) will replicate to each other.

  Note: If two AMS instances (physical servers) out of four go down at the same time, then the remaining two AMS instances would not be able to support dual replication (even if the number of Replicas is configured as '2') because the requisite number of active AMS instances would no longer be available.

- When the AMS instances (physical servers) that had gone down come up again, the initial replication pattern will get restored:
  - AMS ID 1 will replicate to AMS ID 3 and AMS ID 2
  - AMS ID 3 will replicate to AMS ID 2 and AMS ID 4
  - AMS ID 2 will replicate to AMS ID 4 and AMS ID 1
  - AMS ID 4 will replicate to AMS ID 1 and AMS ID 3

5.2.3 Replication for Multi-instanced Deployment

In a multi-instance setup, multiple AMS instances might be running on same physical server host. The AMS replication mechanism is essentially intended to provide high availability in the case of host node failure.

Replication algorithm is as follows:

1. AMS list is sorted based on Host-ID of the AMS node, and for each Host-ID a sorted sub-list is maintained based on AMS ID (of each instance).
2. Each AMS instance replicates to the corresponding AMS instance running on next Host-ID in the sorted list and located at the same index in sorted sub-list of AMS ID.
3. If the target AMS instance is either down (temporarily unavailable) or does not exist (due to an unequal distribution of AMS instances on different nodes) and '`amsoverloadallowedforreplication`' is FALSE, again repeat step 2 if there are more physical servers.
4. If the target AMS instance is down (temporarily unavailable) and '`amsoverloadallowedforreplication`' is TRUE, then the AMS instance at the next index location in the sorted AMS ID sub-list is selected. However, this will lead to overload on the selected AMS instance.
5. On the other hand, if the target AMS instance does not exist (due to an unequal distribution of AMS instances on different nodes) and '`amsoverloadallowedforreplication`' is TRUE, then the AMS instance at the first index location in the sorted AMS ID sub-list is selected. Note that this will also lead to overload on the first AMS instance.
6. If no other physical server exists (after step 3), then replication will not happen.
7. If the configured number of Replicas is more than 1 then repeat steps 2-6 (as applicable), till either the requisite number of replicas are selected or no more physical servers are left.

Note: Do not set `amsPropPhyServerId` for this scenario; otherwise replication behavior might get impacted.

Replication is always attempted on the AMS instance running on the next Host-ID and located at the same index in sorted AMS ID sub-list as the Master AMS instance. If one or more AMS instances are down then varying number of instances will be available on different physical servers and so replication of some instances might not happen. To overcome this scenario, a semi-static parameter '`amsoverloadallowedforreplication`' is supported for AMS.

If the '`amsoverloadallowedforreplication`' parameter is set to "true" (default value is "false") then the replication algorithm will select the next AMS instance in the sorted sub-list of AMS IDs, in case the AMS instance located at the same index as the master AMS instance and on the next Host-ID is down. However, in case there is no AMS instance located at the same index as the master AMS instance and on the next Host-ID, and if the '`amsoverloadallowedforreplication`' parameter is set to "true", then the replication algorithm will select the first AMS instance in the sorted sub-list of AMS IDs.

Note: In case the configured number of replicas (N) is more than 1, then the replication algorithm ensures that no two replicas of the same master AMS instance are co-located on the same physical server. Hence if the number of physical servers is less than N+1 then replication cannot be performed for the configured number of replicas.

Example:

Consider the following scenario where there are two physical servers, each running three instances of AMS:

- Physical Server 1 , Host ID A, AMS ID 1, AMS ID 3, AMS ID 5
- Physical Server 2 , Host ID B, AMS ID 2, AMS ID 4, AMS ID 6

Considering the number of Replicas as '1', the replication logic in this case will be as follows:

- To calculate replica, AMS list is sorted based on Host-ID and within one Host-ID it is sorted based on AMS ID

A: AMS ID 1, AMS ID 3, AMS ID 5

B: AMS ID 2, AMS ID 4, AMS ID 6

- Each AMS instance replicates to the corresponding AMS instance running on next Host-ID in the list and at the same index in sorted sub-list of AMS ID. Hence replication will be:
  - AMS ID 1 will replicate to AMS ID 2
  - AMS ID 3 will replicate to AMS ID 4
  - AMS ID 5 will replicate to AMS ID 6
  - AMS ID 2 will replicate to AMS ID 1
  - AMS ID 4 will replicate to AMS ID 3
  - AMS ID 6 will replicate to AMS ID 5

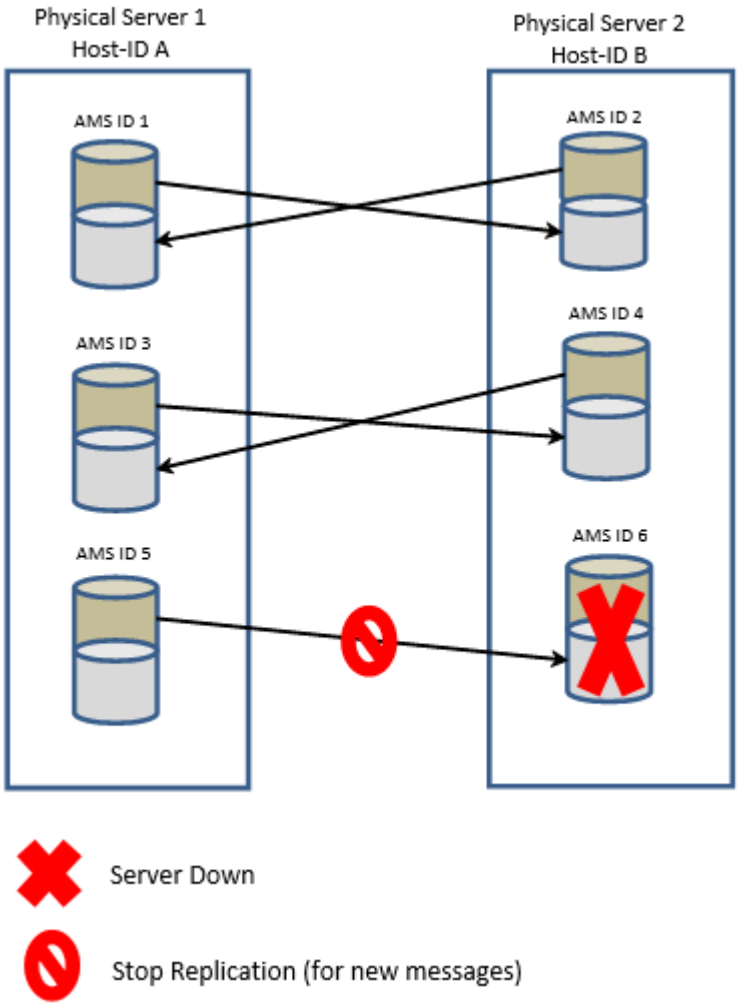


Figure 14: Multi-Instanced AMS Replication (Number of Replicas=1) when one AMS instance is down and overload is not allowed

- If AMS ID 6 is down, then replication will be:
  - AMS ID 1 will replicate to AMS ID 2
  - AMS ID 3 will replicate to AMS ID 4
  - AMS ID 5 will NOT replicate
  - AMS ID 2 will replicate to AMS ID 1
  - AMS ID 4 will replicate to AMS ID 3

Note: In the above example, it has been assumed that the value of the '`amsoverloadallowedforreplication`' parameter is FALSE.

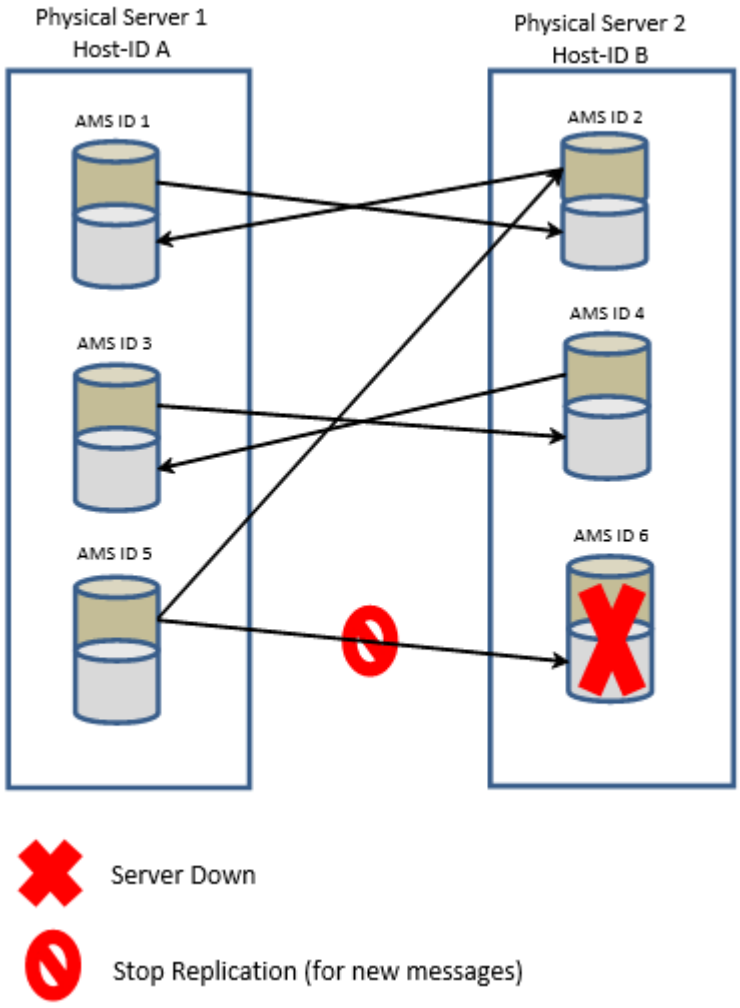


Figure 15: Multi-Instanced AMS Replication (Number of Replicas=1) when one AMS instance is down and overload is allowed

If AMS ID 6 is down and '`amsoverloadallowedforreplication`' is TRUE, then replication will be:

- AMS ID 1 will replicate to AMS ID 2
- AMS ID 3 will replicate to AMS ID 4
- AMS ID 5 will replicate to AMS ID 2
- AMS ID 2 will replicate to AMS ID 1
- AMS ID 4 will replicate to AMS ID 3

Note: AMS ID 2 is overloaded in this case as both AMS 1 and AMS 5 are replicating to it.

When AMS ID 6 comes up again, the initial replication pattern will get restored:

- AMS ID 1 will replicate to AMS ID 2
- AMS ID 3 will replicate to AMS ID 4
- AMS ID 5 will replicate to AMS ID 6
- AMS ID 2 will replicate to AMS ID 1

- AMS ID 4 will replicate to AMS ID 3
- AMS ID 6 will replicate to AMS ID 5

5.2.4 Replication for VM-based Deployment

In this deployment scenario, multiple VMs run on the same physical server and on each VM one instance of AMS is installed. It is recommended that the number of AMS instances (VMs) on each physical server should be same and their capabilities should be compatible. In addition, the Physical Server ID (semi-static parameter `amsphyserverid` in host-specific configuration file) must be set for all AMS instances running on VMs. The Physical Server ID should be same for all AMS instances on VM on same physical server. Valid values of Physical Server ID are in the range of 1-255.

Note: Similar to a multi-instanced deployment, in the case of VM-based deployment also if the configured number of replicas (N) is more than 1, then the replication algorithm ensures that no two replicas of the same master AMS instance are hosted on VMs that are co-located on the same physical server. Hence if the number of physical servers is less than N+1 then replication cannot be performed for the configured number of replicas.

Replication algorithm is as follows:

1. AMS list is sorted based on Physical Server ID and Host-ID (of each VM). For each Physical Server ID, there is a sorted sub-list of Host-ID.
2. Each AMS instance checks the following by its Physical Server ID:
   a. if it is running on a VM;
   b. how many "peer" AMS instances are available on its physical server;
   c. its own relative position (offset) among its peers on the same physical server.
3. Each AMS instance will try to replicate to the corresponding AMS instance running on the next physical server and having the same relative position among its peers.
4. If `amsoverloadallowedforreplication` is false and the target AMS instance (or its VM) is either down (temporarily unavailable) or does not exist (due to an unequal distribution of AMS instances/VMs on different nodes), then this physical server is skipped and replication is targeted at the next physical server (if it exists).
5. If `amsoverloadallowedforreplication` is true and the target AMS instance (or its VM) is down (temporarily unavailable) then replication is targeted at the AMS instance located at the next offset position relative to the old target AMS, on the same physical server.
6. On the other hand, if `amsoverloadallowedforreplication` is true and the target AMS instance (or its VM) does not exist (due to an unequal distribution of AMS instances/VMs on different nodes), then replication is targeted at an AMS instance whose relative position is calculated by adjusting the offset of the original target AMS with respect to the total number of instances/VMs on the same physical server.
7. If no other physical server exists (after step 4), then replication will not happen.
8. If the configured Number of Replicas is more than 1 then repeat steps 3-7 (as applicable), till either the requisite number of replicas are selected or no more physical servers are left.

Note: In step 5 as well as step 6, the selected AMS instance would be overloaded.

Example:

Consider the following scenario where there are two physical servers, each running 3 VMs. Each VM has one instance of AMS:

- Physical Server 1 (`amsphyserverid` = 1)
  - VM 1 : Host ID A, AMS ID 1
  - VM2 : Host ID D, AMS ID 4
  - VM3 : Host ID F, AMS ID 7
- Physical Server 2 (`amsphyserverid` = 2)
  - VM 1 : Host ID B, AMS ID 2
  - VM2 : Host ID E, AMS ID 5
  - VM3 : Host ID G , AMS ID 8

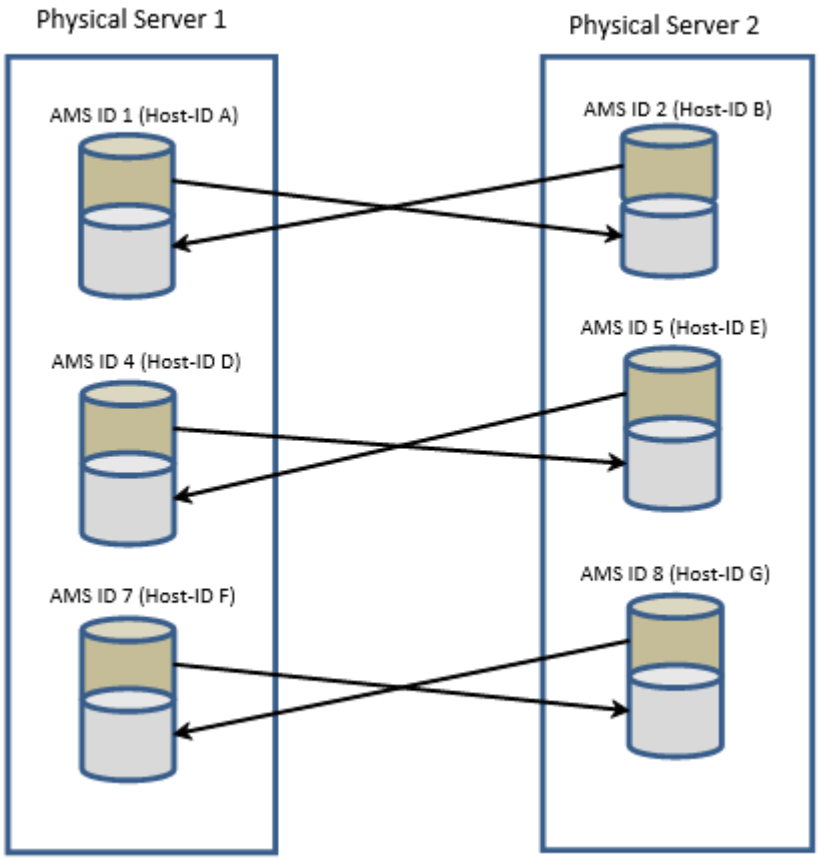


Figure 16: AMS Replication for VM-based Deployment (Number of Replicas=1)

Considering the number of Replicas as '1', the replication logic in this case will be as follows:

- To calculate replica, AMS list is sorted based on Physical Server ID and Host-ID:
  - 1 A: AMS ID 1
  - 1 D: AMS ID 4
  - 1 F: AMS ID 7
  - 2 B: AMS ID 2
  - 2 E: AMS ID 5
  - 2 G: AMS ID 8

Within each physical server sorted Host ID list is:

1. 1 : A D F
2. 2 : B E G

Each AMS replicates to AMS running on next physical server in the list and having same index in sorted Host-ID list.

Hence, replication will be:

- AMS ID 1 will replicate to AMS ID 2
- AMS ID 4 will replicate to AMS ID 5
- AMS ID 7 will replicate to AMS ID 8
- AMS ID 2 will replicate to AMS ID 1
- AMS ID 5 will replicate to AMS ID 4
- AMS ID 8 will replicate to AMS ID 7

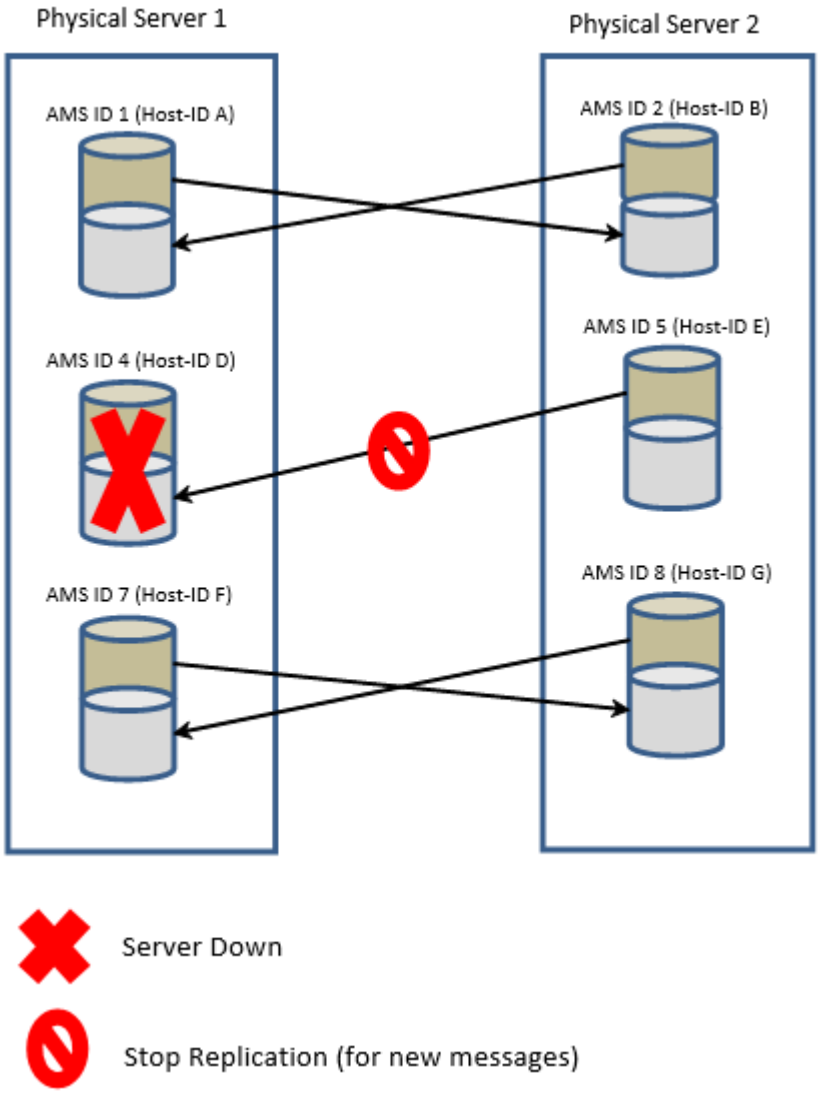


Figure 17: AMS Replication for VM-based Deployment (Number of Replicas=1) when one AMS instance/VM is down and overload is not allowed

If AMS ID 4 goes down or its VM goes down and the semi-static parameter '`amsoverloadallowedforreplication`' is set to false, then replication will be:

- AMS ID 1 will replicate to AMS ID 2
- AMS ID 7 will replicate to AMS ID 8
- AMS ID 2 will replicate to AMS ID 1
- AMS ID 5 will NOT replicate
- AMS ID 8 will replicate to AMS ID 7

Note: AMS ID 5 will not be able to replicate in this case.

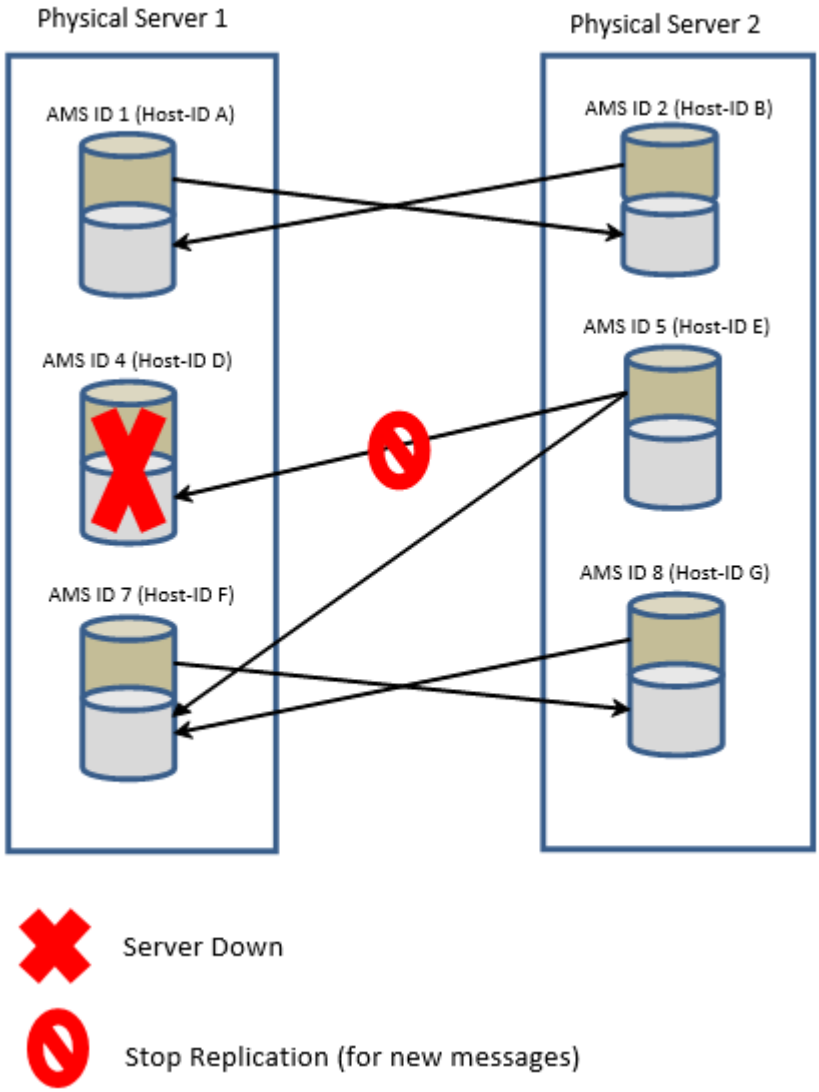


Figure 18: AMS Replication for VM-based Deployment (Number of Replicas=1) when one instance/VM is down and overloaded

If AMS ID 4 is down and '`amsoverloadallowedforreplication`' is TRUE, then replication will be:

- AMS ID 1 will replicate to AMS ID 2
- AMS ID 7 will replicate to AMS ID 8
- AMS ID 2 will replicate to AMS ID 1
- AMS ID 5 will replicate to AMS ID 7
- AMS ID 8 will replicate to AMS ID 7

Note: AMS ID 7 is overloaded in this case as both AMS 5 and AMS 8 are replicating to it.

When AMS ID 4 comes up again, the initial replication pattern will get restored:

- AMS ID 1 will replicate to AMS ID 2

- AMS ID 4 will replicate to AMS ID 5
- AMS ID 7 will replicate to AMS ID 8
- AMS ID 2 will replicate to AMS ID 1
- AMS ID 5 will replicate to AMS ID 4
- AMS ID 8 will replicate to AMS ID 7

5.3 Transaction Capability and Database Integrity

The database used in the AMS supports full transactional database capabilities, consisting of:

- Committed checkpoints after transaction completion to ensure database integrity
- Log records that provide journal files to reapply the changes on the database after the last checkpoint

Database integrity is assured by using a single table design in the AMS. This method has the advantage that the efforts to ensure database integrity are limited to the items as described under transactional capabilities above.

The integrity of the replica data is further ensured by using SCTP checksum calculations and error detection.

5.4 Contingency

The AMS provides graceful behaviour to accommodate the following contingency situations:

- When the AMS master node becomes unavailable
- When the AMS replica node becomes unavailable
- When the RAID-1 mirrored disks become unreadable
- When the disks are running full
- When the message database becomes unreadable
- When the AMS is flooded under a very high peak of incoming messages

5.5 Availability

Seamless scalability, in combination with multiple replicas, ensures the AMS has extremely high availability. Availability can easily be boosted by adding AMS nodes to the SMS network and defining a replica for the message queues to assure delivery if an AMS node fails.

This example illustrates this process. In this example, the AMS is configured such that under normal operating conditions, it will store a replica of each incoming message. The diagram below shows that the incoming MO message is handled by AMS-1 (the master for queue 2). The master (AMS-1) is responsible for providing the replicas (on node AMS-2) with a copy of the message.

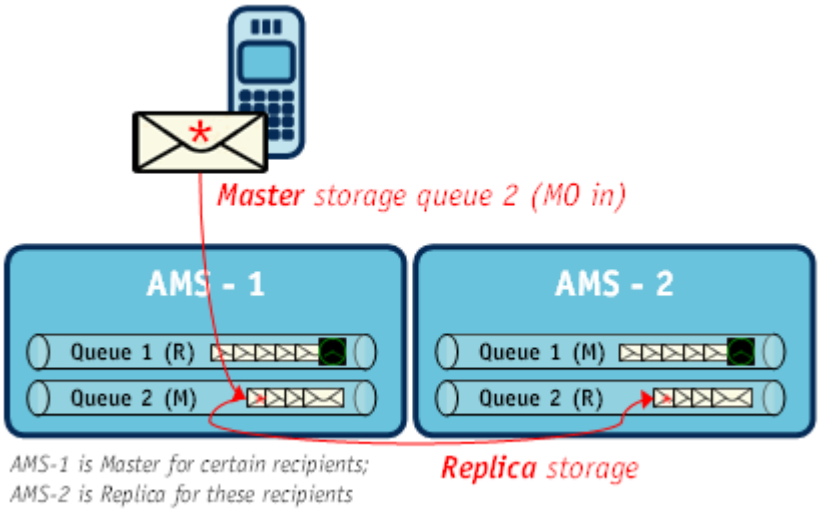


Figure 19: Incoming MO message

When node AMS-1 becomes unavailable, the other ZephyrTel Mobile Messaging nodes automatically acknowledge it. AMS-2 becomes the active replica for messages previously sent to AMS-1 and AMS-2 becomes the new master for new messages for the recipients previously stored in AMS-1.

AMS-2 will begin delivery of messages from the replicated queue and will accept new messages, just as the original master queue on AMS-1 did before it failed. Before the AMS takes over the active replica role, it waits for a configurable “hold-off time” (default 60 seconds).

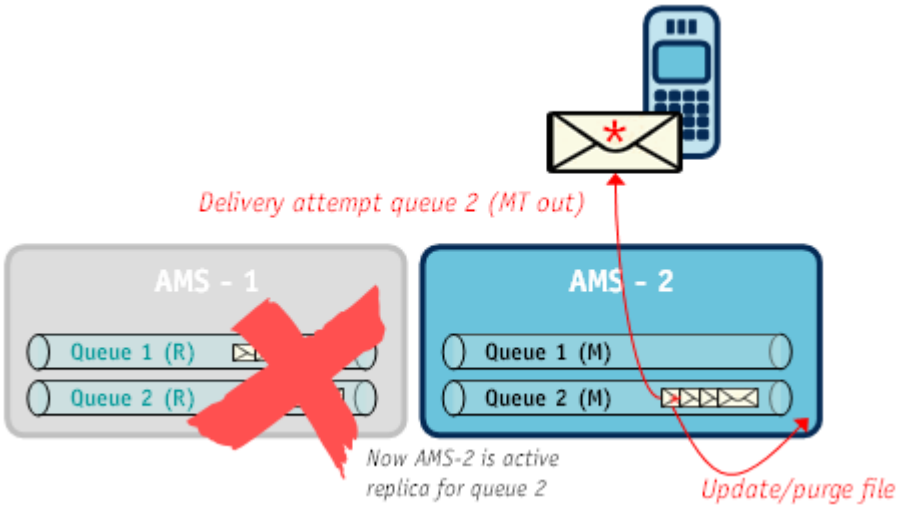


Figure 20: AMS taking active replica role

In the normal operation the AMS can store messages redundantly: one in the receiving AMS instance (the master) and a second one within another instance (the replica). In case the master becomes inactive, the replica will become active and start delivering messages.

The purge list is a list of the messages delivered by the replica while there is no master available, or vice versa a list of messages delivered by the master when there is no replica available. This purge list is stored in the memory. Upon successful delivery of a message that has been replicated before, the AMS will create an entry in the purge list. This purge list is exchanged with the other AMS node in

the AMS pair after it becomes available. With the purge list, the other AMS node is notified about the already delivered messages, and will delete (purge) the message from its own delivery queue. This happens during the AMS startup.

In case of double failures, thus in case where the master AMS process stops before the replica AMS is started or fully synchronized, or vice versa when the active replica AMS process stops before the master AMS is fully synchronized, duplicate delivery will happen due to the fact that the delivery history of all/some of the messages in the purge list is not completely synchronized with the new AMS instance.

5.6 Scalability

The AMS can be easily scaled up by adding additional nodes (up to 255 in total). For reliability purposes, the AMS scales in pairs (two nodes).

To simplify scalability, the AMS supports an automatic sign-on and configuration method for newly added AMS nodes. This "plug-and-play" mechanism guarantees that new AMS nodes are automatically included in message distribution from the RTRs.

When more message queue storage space is required, simply add more AMS pairs to the ZephyrTel Mobile Messaging SMS network. The diagram below illustrates a sample configuration with six AMS nodes (a total of three AMS pairs).

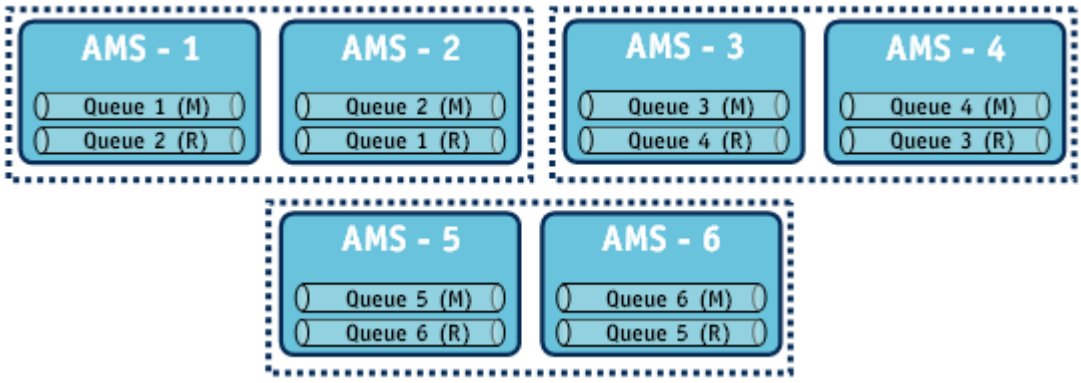


Figure 21: Six AMS nodes (three AMS pairs)

For example, node AMS-3 is the master (M) for messages of a certain range of recipients (referred to as queue 3) and replica (R) for messages stored in AMS-4 (referred to as queue 2).

This functionality ensures that the storage capacity of a single AMS node is never a bottleneck for the SMS network's buffering capacity.

5.7 Manual Control of Failover

For maintenance purposes and disaster recovery situations, it is possible to manually control the failover mechanism and instruct an active Master or active Replica to transfer delivery to another AMS.

To transfer delivery, set the `activeReplicaIdentifier` of the currently active AMS (for example, AMS-1) to the AMS that should take control (for example, AMS-2). Control does not transfer until AMS-2 accepts.

When AMS-2 takes control of delivery, an active indication is returned to AMS-1 and added to the `failoverControl` table.

Chapter 6

Routing Paths

Topics:

6.1 Introduction

This chapter discusses the routing paths that are relevant for the AMS in typical SMS network environment.

An SMS routing path is the route that an SMS message follows through the network, where each stop is expressed in the message type at that moment. For example, the routing path MO-MT-MO defines that the incoming mobile-originated (MO) message is first delivered to a mobile station (as an MT) and, if this delivery fails, is routed further as a mobile-originated (MO) message (e.g. to an SMSC). So, in fact, the actual routing path is either MO-MT or MO-MT-MO.

In summary:

- Incoming mobile-originated (MO) messages always enter the SMS network via a RTR (or RTR/FWL)
- Incoming application-originated (AO) messages always enter the SMS network via the HUB
- Mobile-terminated (MT) messages are delivered by the RTR
- Application-terminated (AT) messages leave the SMS network via the HUB.

The sections below describe the MO and AO routing paths that are relevant for the AMS in detail.

6.2 Mobile-Originated (MO) Routing Paths

Mobile-originated (MO) routing paths cover the routes of all incoming MO messages. Incoming MO messages always arrive in the SMS network via the RTRs.

The following sections only describe the routing paths related to the AMS. For a complete overview of all routing paths available in the SMS network, refer to the RTR Operator Manual.

6.2.1 MO-Store-MT: Store for Delivery to MS

In some cases, a first delivery attempt is not desired, and the incoming MO message should be sent directly to the AMS for storage and later delivery.

Examples of these cases are:

- A delivery attempt was performed by another device immediately before sending the message to the ZephyrTel Mobile Messaging system
- It is known that the destination is probably not available, so a later delivery is more appropriate
- A deferred (scheduled) delivery is required at a later date/time

The diagram below shows the flow for the MO-persistent routing path.

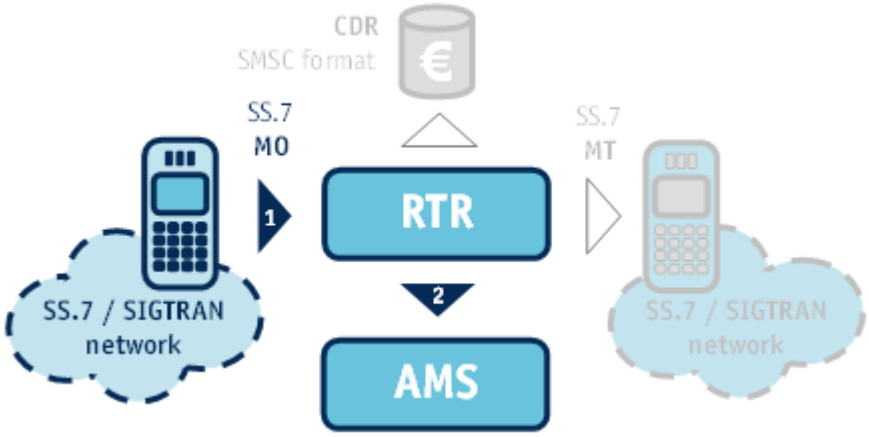


Figure 22: MO-Store for delivery to MS

The mobile station (MS) sends an MO message to the RTR [flow 1]. This MO message is immediately forwarded to the appropriate AMS [flow 2] for later (scheduled) delivery.

Note: The basics of this routing path are similar to the straightforward MO traffic handling of a traditional SMSC.

6.2.2 MO-MT-Store: Route to MS Fallback to Storage

As depicted in the diagram below, the mobile station sends an MO message to the RTR [flow 1], which performs a first delivery attempt (FDA) [flow 2].

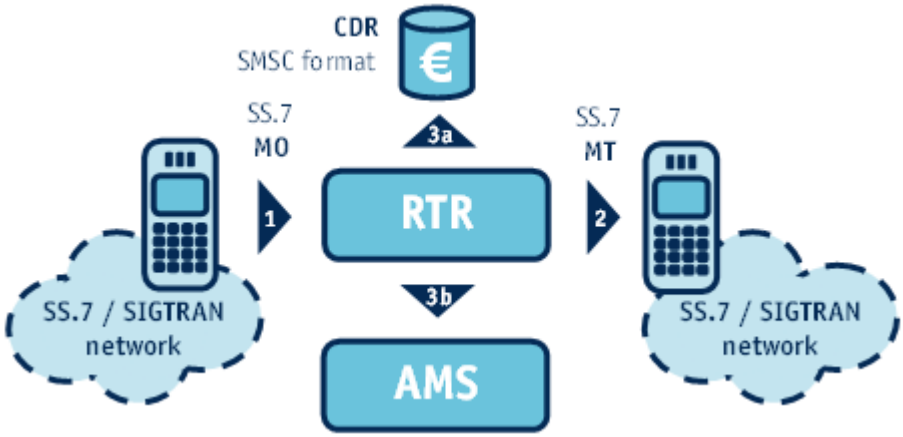


Figure 23: MO-Route to MS fallback to storage

If the FDA is successful, the RTR generates a billing record (CDR) [flow 3a]; otherwise, the message is forwarded to the AMS [flow 3b] for later delivery.

When the message reaches AMS:

- It is placed in the appropriate message queue
- The corresponding delivery scheme determines when and how often a delivery attempt is made

Optionally, the RTR can send a status report to the message originator (not depicted in the diagram).

6.2.3 MO-Store-AT: Store for Delivery to Application

In some cases, a first delivery attempt is not desired, and the incoming MO message should be sent directly to the AMS for storage and later delivery.

Examples of these cases are:

- A delivery attempt was performed by another device immediately before sending the message to the ZephyrTel Mobile Messaging system
- It is known that the destination is probably not available, so a later delivery is more appropriate
- A deferred (scheduled) delivery is required at a later date/time

The diagram shows the flow for the MO-persistent routing path.

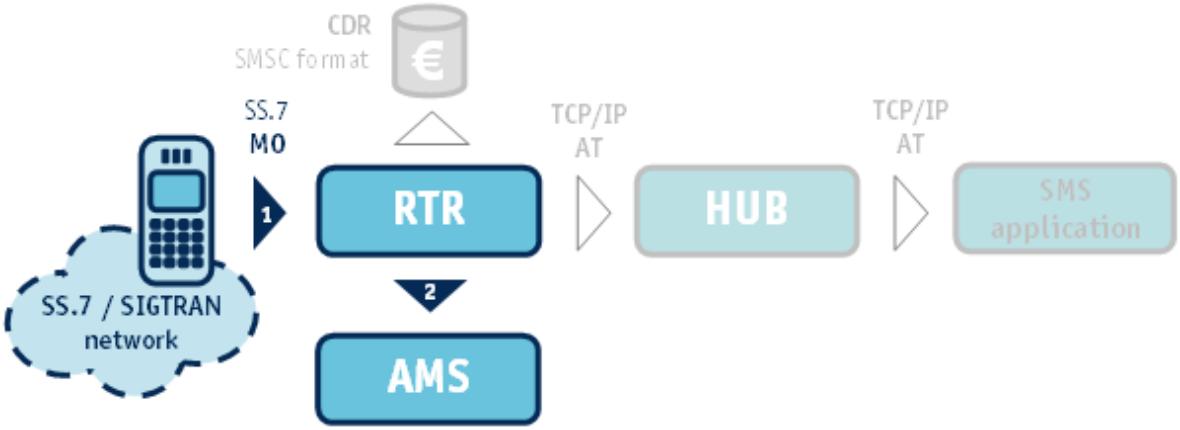


Figure 24: MO-Store for delivery to Application

The mobile station (MS) sends an MO message to the RTR [flow 1]. This MO message is immediately forwarded to the appropriate AMS [flow 2] for later (scheduled) delivery to the SMS application.

Note: The basics of this routing path are similar to the straightforward MO traffic handling of a traditional SMSC.

6.2.4 MO-AT-Store: Route to Application Fallback to Storage

As depicted in the diagram, the mobile station sends an MO message to the RTR [flow 1], which performs a first delivery attempt (FDA) toward the SMS application [flow 2a and 2b].

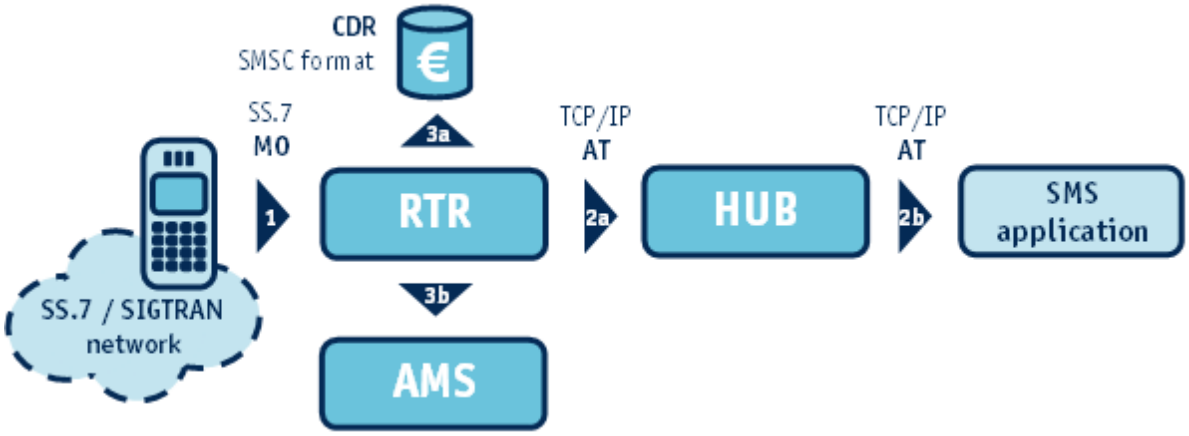


Figure 25: MO-Route to Application fallback to storage

If the FDA is successful, the RTR generates a billing record (CDR) [flow 3a]; otherwise, the message is forwarded to the AMS [flow 3b] for later delivery.

When the message reaches AMS:

- It is placed in the appropriate message queue
- The corresponding delivery scheme determines when and how often a delivery attempt is made

Optionally, the RTR can send a status report to the message originator (not depicted in the diagram).

6.3 Application-Originated (AO) Routing Paths

Application-originated (AO) routing paths cover the routes of all incoming SMS messages originating from SMS applications. Incoming AO messages always arrive in the SMS network via the HUBs.

The following sections only describe the routing paths related to the AMS. For a complete overview of all routing paths available in the SMS network, please refer to the RTR documentation.

6.3.1 AO-Store-MT: Store for Delivery to MS

In some cases, a first delivery attempt is not desired, and the incoming AO message should be sent directly to the AMS for storage and later delivery.

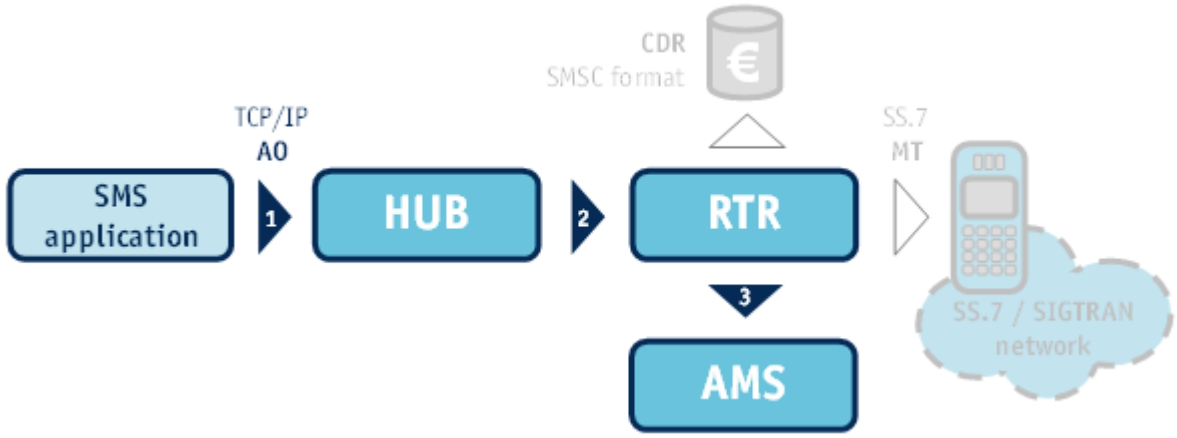


Figure 26: AO-Store for delivery to MS

The SMS application sends an AO message to the HUB [flow 1]. The AO message is immediately forwarded to the RTR [flow 2], which forwards the message to the appropriate AMS [flow 3] for later (scheduled) delivery.

Note: The basics of this routing path are similar to the simple AO traffic handling of a traditional SMSC.

6.3.2 AO-MT-Store: Route to MS Fallback to Storage

As depicted in the diagram below, the SMS application sends a message to the HUB [flow 1]. The AO message is immediately forwarded to the RTR [flow 2], RTR forwards the message to the AMS without making an FDA to the MS (message is not stored in the database at this point) [flow 3], AMS sends an Ack. and then the message back to the RTR for performing a first delivery attempt (FDA) [flow 4].

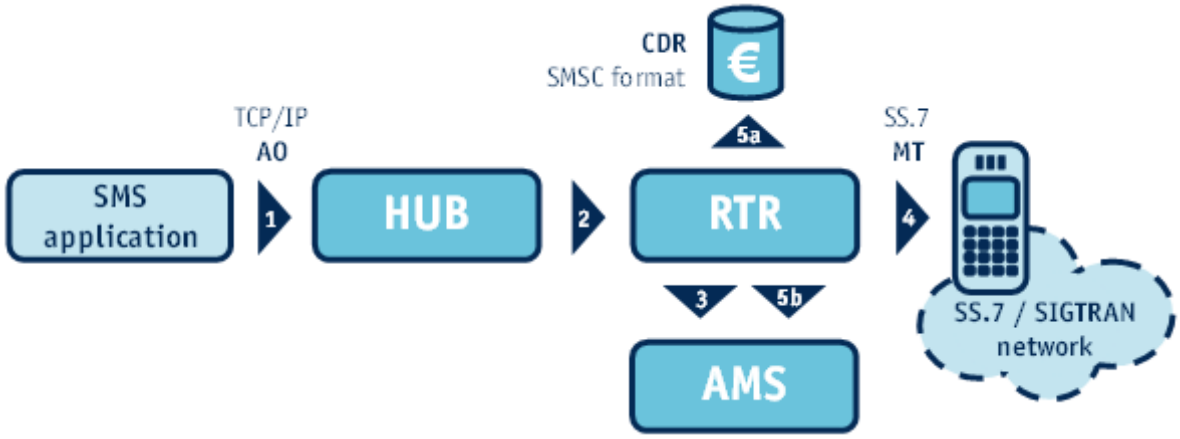


Figure 27: AO-Route to MS fallback to storage

If the FDA is successful, the RTR generates a billing record (CDR) [flow 5a]; otherwise, the routing result is returned to the AMS [flow 5b].

Note that the message had earlier been placed in the appropriate message queue (e.g. a queue defined specifically for the originator application or the standard queue for the destination mobile station)

when the AMS had first received it from the RTR (before the FDA). Hence when the AMS receives the routing result (after the failed FDA), it just stores the message in its database for later retries.

The corresponding delivery scheme determines when and how often a delivery attempt is made.

Optionally, a notification can be sent to the originating application via the HUB (not depicted in the diagram).

6.3.3 AO-Store-AT: Store for Delivery to Application

In some cases, a first delivery attempt is not desired, and the incoming AO message should be sent directly to the AMS for storage and later delivery to an SMS application.

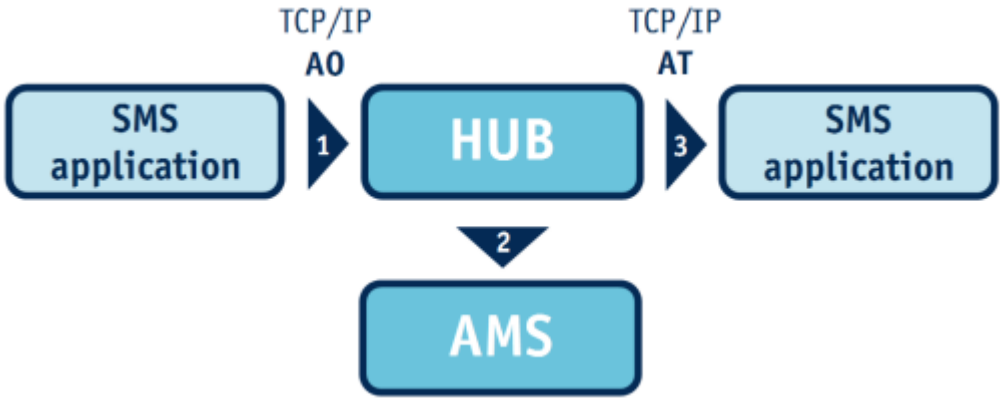


Figure 28: AO-Store for delivery to application

The SMS application sends an AO message to the HUB. The AO message is immediately forwarded to the RTR, which forwards the message to the appropriate AMS for later (scheduled) delivery.

6.3.4 AO-AO-Store: Route to SMSC, Fallback to Storage

Mobile Messaging can function as an Application Gateway (AGW), multiplexing AO messages from multiple applications and forwarding them to a service centre (as AO messages), using the sessions of a single virtual application.

In AO-AO-Store routing, the HUB attempts to deliver the message to the service centre. If the service centre rejects the message, the Mobile Messaging system falls back to storing the message in the AMS for later delivery attempts.

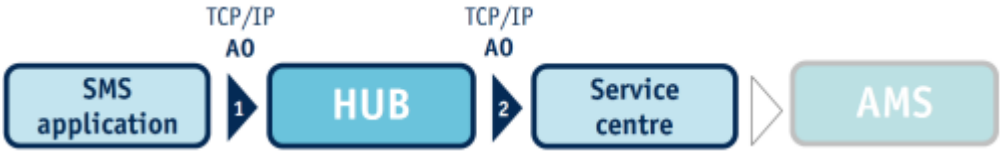


Figure 29: AO-AO-Store

6.3.5 AO-Store-AO: Store for Forwarding as AO

Mobile Messaging can function as an Application Gateway (AGW), multiplexing AO messages from multiple applications and forwarding them to a service centre (as AO messages), using the sessions of a single virtual application.

In AO-Store-AO routing, the RTR immediately stores the message in the AMS for later delivery attempts to the service centre.

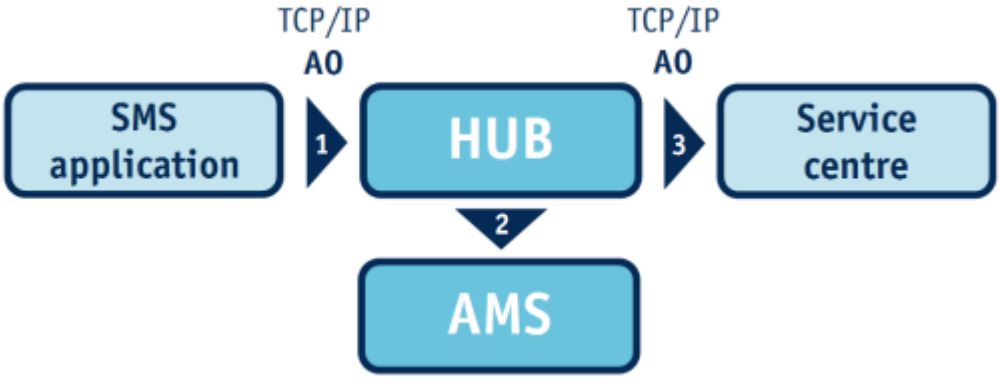


Figure 30: AO-Store-AO

6.3.6 AO-AT-Store: Route to Application, Fallback to Storage

Mobile Messaging can function as an Application Gateway (AGW), multiplexing AO messages from multiple applications and forwarding them to a service centre (as AO messages), using the sessions of a single virtual application.

In AO-AT-Store routing, the HUB attempts to deliver the message to the SMS application. If the application rejects the message, it is stored in the AMS for later delivery attempts.

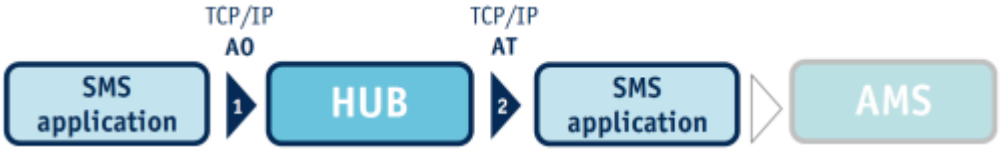


Figure 31: AO-AT-Store

6.4 Application-Terminated (AT) Routing Paths

Application-terminated (AT) routing paths cover the routes of all incoming SMS messages originating from SMS applications. Incoming AT messages always arrive in the SMS network via the HUBs.

The following sections only describe the routing paths related to the AMS. For a complete overview of all routing paths available in the SMS network, refer to the RTR documentation.

6.4.1 AT-AT-Store: Route to Application, Fallback to Storage

Mobile Messaging can function as an Application Gateway (AGW), multiplexing AT messages from multiple applications and forwarding them to a service centre (as AT messages), using the sessions of a single virtual application.

In AT-AT-Store routing, the HUB attempts to deliver the message to the application. If the application rejects the message, the Mobile Messaging system falls back to storing the message in the AMS for later delivery attempts.

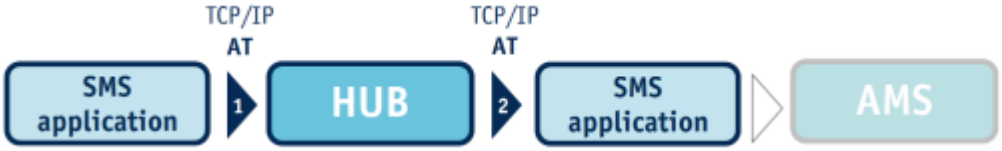


Figure 32: AT-AT-Store

6.4.2 AT-Store-AT: Store for Delivery to Application

Mobile Messaging can function as an Application Gateway (AGW), multiplexing AO messages from multiple applications and forwarding them to a service centre (as AO messages), using the sessions of a single virtual application.

In AT-Store-AT routing, the RTR immediately stores the message in the AMS for later delivery attempts to the service centre.

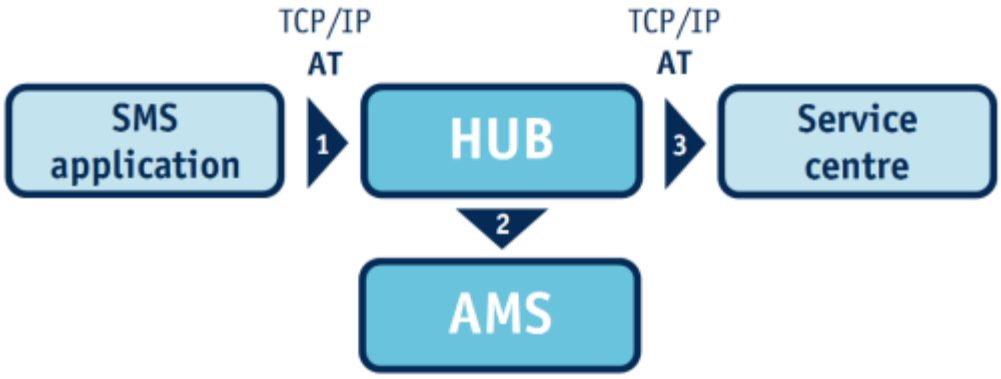


Figure 33: AT-Store-AT

6.5 Message Delivery

When a message reaches AMS:

- It is placed in the appropriate message queue
- The delivery scheme that corresponds to the message queue determines when and how often a delivery attempt is made

The same message may change delivery schedules several times, depending on the delivery errors that occurred or the availability of the destination. Message delivery stops when:

- A permanent error occurs, or
- The validity period of the message expires

The AMS is responsible for the delivery to the following destination types:

- Mobile Stations—Delivery takes place via the RTR toward the PLMN
- Applications—Delivery takes place via the RTR and HUB toward the applications

6.5.1 Delivery to Mobile Station (MT)

As shown in the diagram below, the AMS initiates a delivery by submitting the SMS message to the RTR [flow 1]. The RTR then performs the actual MT delivery attempt to the mobile station [flow 2].

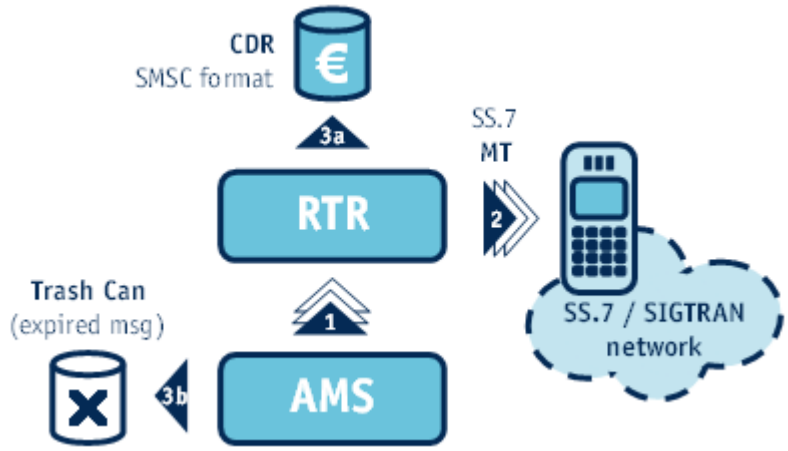


Figure 34: MT delivery attempt

If the delivery attempt is successful, the RTR generates a billing record [flow 3a]; otherwise, flows 1 and 2 are repeated according to the delivery schedule.

If no successful delivery attempt could be made during the message's validity period, or if a permanent error occurred during a delivery attempt, the message is deleted (purged) from all AMS nodes [flow 3b].

Optionally (not depicted in the diagram), the RTR can send a:

- Status report to the message originator, or
- Notification via the HUB to the originating application

Note: The MO-MT routing path must be enabled in the RTR license file for MO-Store-MT routing to work.

6.5.2 Delivery to Application (AT)

The diagram below depicts how the AMS initiates an AT delivery by submitting the SMS message to the RTR [flow 1].

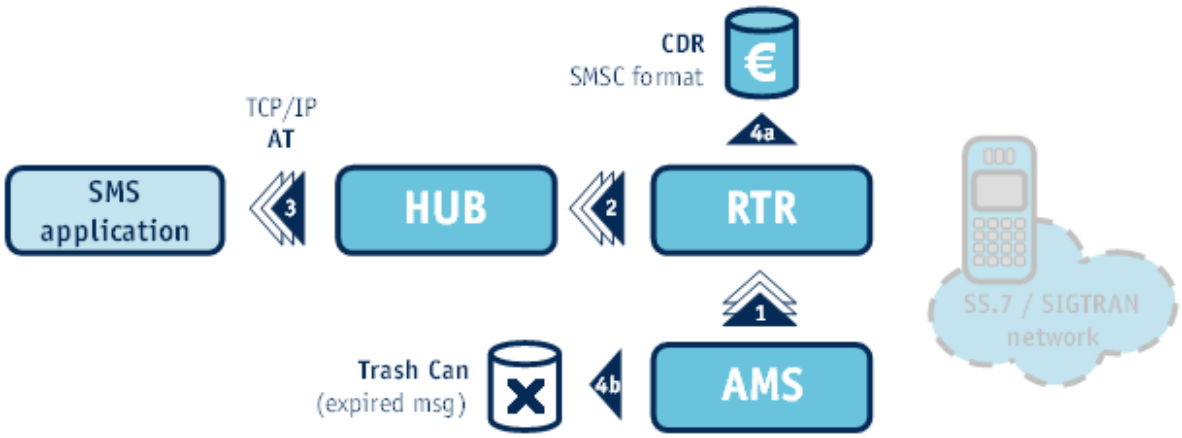


Figure 35: AT delivery attempt

The RTR then performs the actual AT delivery attempt by forwarding the message to the HUB [flow 2], which delivers it to the application [flow 3].

If the delivery attempt is successful, the RTR generates a billing record [flow 4a]; otherwise flows 1, 2, and 3 are repeated according to the delivery schedule.

If no successful delivery attempt could be made during the message's validity period, or if a permanent error occurred during a delivery attempt, the message is deleted (purged) from all AMS nodes [flow 4b].

Optionally, the RTR can send a status report to the message originator (not depicted in the diagram).

Note: The MO-AT routing path must be enabled in the RTR license file for MO-Store-AT routing to work.

Chapter 7

Message Queues

Topics:

7.1 Introduction

The AMS provides a queue-based storage system for referring to:

- Stored messages
- Stored Icache records

A message queue is a list of message references; the actual messages are stored in a persistent database. Each message reference contains the message ID and relevant status information required for delivery scheduling (such as next delivery time, number of delivery attempts, and most recent error).

7.2 Queue Types

Each message queue is associated with a certain type of destination or specific purpose. In the AMS the queue types are:

- Application queue—For AT messages to be delivered to an application
- Mobile station queue—For MT messages to be delivered to a mobile station (MS)
- Icache queue—For the state and certain parameters of a message while it is being processed by an external SMSC

Note: When an AMS message queue is configured on the MGR, it applies to all active AMS nodes within the same domain. Hence each AMS node accesses a logical partition of a single message queue.

The sample diagram below illustrates this function for Queue 1, Queue 2 and Queue 3, each of which resides in AMS-1, AMS-2, and AMS-3 nodes.

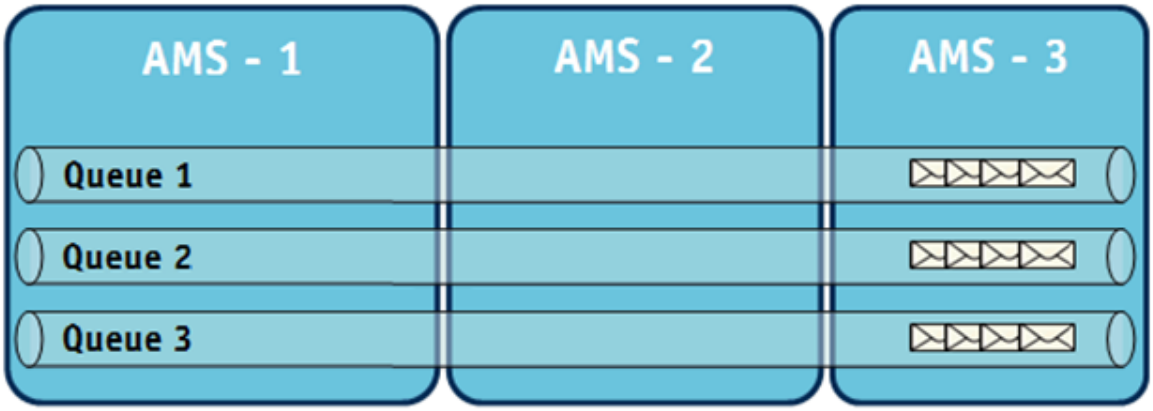


Figure 36: Message queue spanning multiple nodes

In this example, no queue is redundant (no replicas are configured).

7.3 Automatic Queue Assignment

The RTR indicates to the AMS which type of message queue should be used for a message, after which the AMS looks up a matching queue for that type and destination. If no specific message queue exists for the specific destination, the AMS automatically creates a recipient buffer in the appropriate default queue. The default queue parameters are configurable by queue type in a template.

This functionality allows an SMS network operator to predefine queues for specific originators or recipients, while others can be created dynamically using the available defaults.

Message queues are managed using the MGR.

7.4 Message Buffers

Within a message queue, messages can be stored for multiple recipients. For each new recipient, the AMS creates a recipient buffer entity automatically when a message arrives in that particular queue. Each queue has a parameter that limits the number of messages stored within a particular recipient buffer, and which thus limits the number of messages destined for a particular recipient.

Message buffers are removed when no messages are outstanding for a particular recipient.

In the case of an MT message, the recipient number that identifies the recipient buffer is based on the international format (as converted by the RTR) and used by the MGR as input in the routing rules.

In the case of an AT message, the recipient number is based on the short number used to identify the application.

Note: If AT extrusion is enabled for AT parallel delivery, the buffer is created both per original recipient number and per AT short number. Refer to section *AMS Feature Summary*, AT parallel delivery for details of AT extrusion feature.

7.5 Queue Entity

Message queues must be defined and activated in the MGR before they can be used. The queue entity is a definition of one queue and its relevant parameters:

- Name
- Description
- Type
- Priority
- Queue size
- Delivery scheme
- Sri-SM Priority

When a new incoming message cannot be stored due to the fact that the queue is full, the message is rejected.

For application-originating (AO) messages, the RTR will also pass the service class parameter of the corresponding application to the AMS.

Up to 1000 queue entities can be defined in the AMS.

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

7.6 Queue Priority

The order of messages in a queue is such that the first message to be delivered is at the head of the queue. Therefore, messages are first sorted on priority, and then on submit time (except for messages with a deferred delivery time).

When checking for messages to be delivered, all message queues are scanned in order of their queue priorities from 99 (high) to 0 (low). Messages in a high-priority queue will always be delivered before messages in a low-priority queue.

The following example illustrates that queues 4 to 8 have the same priority (50); this implies that all messages in these five queues will be treated with the same priority. However, they may have different delivery schemes.

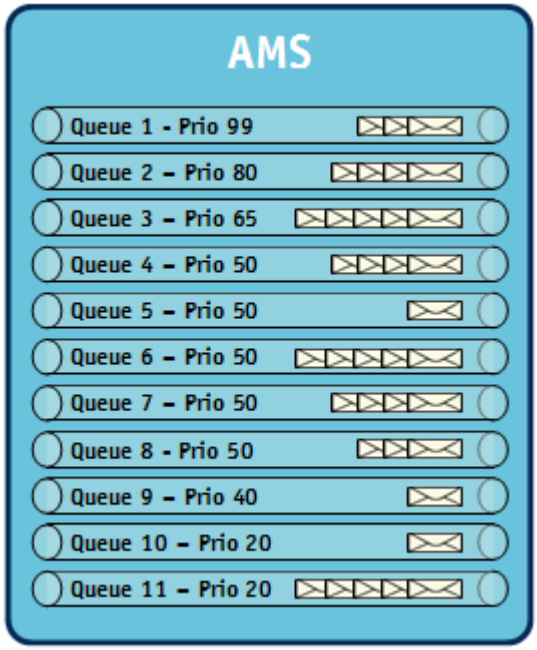


Figure 37: Queue priority example

For example, it is possible to define a separate “high-priority” queue for emergency messages that always require precedence over standard messages, for example, severe weather alerts or important financial stock updates.

7.7 Queue SRI-SM Priority

SRI-SM Priority configuration for an AMS queue controls the setting of the priority field in the SRI-SM Request (to be sent by the RTR) for the messages stored in the current queue, 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', this parameter will have no impact on the value of the sm-RP-PRI in the SRI-SM Requests for stored messages.

Default value of this parameter is 'High'.

Note: This parameter is applicable only when a message is retried from the AMS queue after having already been stored in the database.

In order to configure the 'SRI-SM Priority' setting in the Manager interface (**Storage ➤ Queues**), refer to the MGR Operator Manual.

7.8 Queue Operations

The MGR and the command-line interface enable analysis and management of message queues. The following queue operations are supported:

- Create
- Activate
- De-activate
- Delete

These queue management activities are considered system administration tasks and, as such, should be executed with care.

7.9 Message ID Generation Algorithm

This section describes the message ID generation algorithm for the benefit of the AMS operator. This algorithm is subject to change without notice.

In the AMS, the upper 8 bits are filled with the AMS ID in reverse bit order. This means that the AMS with ID 1 will have the highest (31 bit set). The AMS will generate a message ID based on an internal counter. This counter value is checked with the message IDs currently present in the AMS and incremented if a match is found.

The AMS will generate a message ID as a 32-bit unsigned integer to satisfy SMPP 3.3 requirements. In a system containing multiple AMS nodes, a range of IDs is assigned to each AMS node to ensure the uniqueness of message IDs across the nodes. When a message ID is generated, the AMS checks if a message with the same ID is already stored on the same node. If that is the case, a new message ID is generated and the ID is checked again.

Note: The AMS message ID generation algorithm is different from the algorithm that the Router (RTR) uses for the AO-MT routing path.

Applications should treat the message ID as an opaque object and should not (and do not need to) parse the message ID, as the generation algorithm could change in later AMS versions. Additionally, the same message ID could be reassigned to another message after successful delivery of the previous message. Depending on the message volume and the number of AMS nodes (or after an AMS restart), a message ID could be reassigned within hours.

Chapter 8

Delivery Schemes

Topics:

8.1 Introduction

Delivery schemes determine when the AMS will make delivery attempts for messages in the queues and when messages expire.

The following delivery schemes types are available:

- Standard delivery scheme—Used for normal traffic under standard conditions
- Error-dependent delivery schemes—Used when a specific network error occurred

This chapter provides an overview of the delivery scheme concept and the types of delivery implementations.

8.2 Delivery Scheme Concept

A delivery scheme is a set of relative time intervals that define the time that will elapse between two consecutive delivery attempts. Each delivery scheme contains up to 100 configurable relative intervals. The MGR provides a Web-based interface to manage AMS delivery schemes.

The diagram below shows an example of a simple delivery scheme with 11 delivery attempts, with interval t_i between delivery attempts i and i+1.

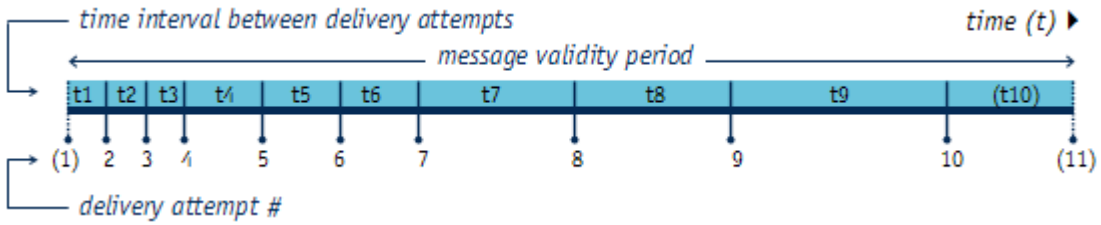


Figure 38: Sample delivery scheme

The first delivery (#1) is optional (such as when the first delivery attempt has already been performed by the RTR FDA).

Similarly, the last delivery attempt (#11) is also optional. When the message expires, the AMS can kill the message after attempt #10 or perform a final delivery attempt (#11) t10 seconds later, and then kill the message.

The expiration time is at the end of the message validity period, which is the sum of all intervals (in this example, t1 + t2 + … + t10).

8.3 Delivery Scheme Entity

Delivery schemes must be defined and activated in the MGR before they can be used. The delivery scheme entity is the definition of one delivery scheme and its relevant parameters.

Under normal circumstances, the standard delivery schemes are used to schedule each delivery attempt for an MT or AT message. Define standard delivery schemes in the AMS system by using the MGR to set attempt intervals and the number of attempts.

Standard delivery schemes are attached to one or more message queues, so all messages in that queue will follow the same scheme (albeit individually).

Up to 100 delivery scheme entities can be defined in the AMS.

For information about configuring the delivery scheme entity, refer to the MGR Operator Manual.

8.4 Delivery Scheme and Message Retention

As mentioned above (see *Delivery Scheme Concept*), the AMS normally expires a message when the last delivery interval configured for the relevant delivery scheme is completed. In such a case the message is removed immediately from the recipient buffer and the database.

However, if a delivery scheme is configured to retain messages (which are following that scheme) even after all configured delivery intervals are completed, then all messages in a recipient buffer associated with the scheme will be retained and such messages will not expire until their respective validity periods get elapsed or the number of delivery attempts exceeds the configured maximum number of attempts for the scheme.

Once the messages in a recipient buffer are retained, there will be no further periodic retries from that recipient buffer and its interval counter will not be incremented any more, since it is already at the last interval; instead, the next delivery attempt will be scheduled for the time when the message having the earliest validity period would expire.

At this scheduled time, there will be a delivery attempt for the first available retained message in the buffer (which may be different from the message about to expire); irrespective of the outcome of this delivery attempt, the message(s) which was actually expiring at this time will get removed from the AMS. Now, if the delivery attempt resulted in a temporary error then the next delivery attempt will again be scheduled based on the earliest validity period among the remaining messages, and so on; hence the message retention mechanism can be described as successive extensions of the last delivery interval based on ascending order of the validity periods of the retained messages.

In case a new/replaced message becomes available (or an existing message gets modified) in the same recipient buffer or an alert is received for the same recipient while awaiting the expiry of the next earliest message validity period, then the AMS will trigger a delivery attempt for the first available retained message in the buffer. Note that the alert could be a service center alert, an application alert or a manual alert (generated through the QCLI or the Customer Service Tool of the CCI).

Whenever a delivery attempt is made from a recipient buffer containing retained messages, depending on the outcome of that delivery attempt any one of the following actions would be taken:

- The next message in the buffer would be attempted for delivery (i.e. if the first attempt is successful or results in a permanent error w.r.t. the concerned message)
- All retained messages along with the recipient buffer would be removed (i.e. if the delivery outcome happens to be a permanent error w.r.t. the particular recipient)
- The system would again go back to waiting for the expiry of the earliest message validity period among all the remaining retained messages (i.e. if the delivery outcome is a temporary error).

A recipient buffer with retained messages can revert to “normal” scheduling, i.e. resume periodic retries based on the configured delivery intervals, only under one of the following three scenarios:

- If a new message is received without having first undergone a FDA (by the RTR), and if the delivery scheme is configured for restarting the delivery schedule upon receiving such a message (in this case periodic retries will resume with an immediate retry of the first available message in the buffer).
- If the delivery scheme changes to an error-dependent scheme based on an appropriate network error ("temporary error") encountered while making a delivery attempt for a retained message (in this case periodic retries will resume starting from the first configured delivery interval).
- If the configuration parameter necessary for enabling retention of messages is disabled by the user on the current delivery scheme, and subsequently a new/replaced message becomes available, or an existing message gets modified, or an alert is received -- for the same recipient in question.

8.5 Default Delivery Schemes

The AMS includes two default delivery schemes, which are contained in the `deliverySchemeTable`. The default schemes can be modified, but they cannot be deleted.

The first default delivery scheme (index 1 in the table) has the following interval durations:

1. 3 intervals of 5 minutes each
2. 8 intervals of 30 minutes each
3. 20 intervals of 180 minutes each

The second default delivery scheme (index 2 in the table) has the following interval durations:

1. 3 intervals of 15 minutes each
2. 8 intervals of 60 minutes each
3. 20 intervals of 180 minutes each

8.6 Error-Dependent Delivery Scheme

The error-dependent delivery scheme is a set of delivery schemes that apply in the event of network errors. Before error-dependent delivery schemes can be used, they must be defined in the MGR. Error-dependent delivery schemes are available for temporary errors and can be divided as follows:

- Application-terminated (AT) errors
- Mobile-terminated (MT) errors
- Send_Routing_Info_for_SM (SRI) errors

For more information about configuring error-dependent delivery schemes, refer to the MGR Operator Manual.

8.7 Switching Between Delivery Schemes

This chapter shortly describes how the Delivery Schemes and Error-Dependent DeliverySchemes result into the selection of the next retry.

An Error-dependant delivery scheme is determined based on the result of the last delivery. In case there is no error, then the standard assigned delivery scheme of the queue is selected.

Next to this, a reference is maintained which points to a delivery interval belonging to this scheme. On every delivery attempt which results in the same scheme, this reference is changed, so it points to the next configured interval. However, if a different scheme is determined, then the reference points to the first interval of the new delivery scheme.

An AlertSC or a recipient queue which is triggered by a new message without a prior FDA (if "Restart on new message" is enabled) will trigger a delivery attempt. After the delivery result is received, the delivery scheme is determined as described above.

Summarizing:

- A reference to the delivery scheme interval is maintained.
- After every attempt, the scheme is determined and assigned.
- Only if the scheme changed, then the reference is set to the first interval of the new scheme.
- Alerts are triggered as normal; only their result can alter the scheme.

Chapter 9

Statistics

Topics:

9.1 Introduction

The AMS provides many statistical counters for all types of results and activities related to the storage and delivery of SMS messages.

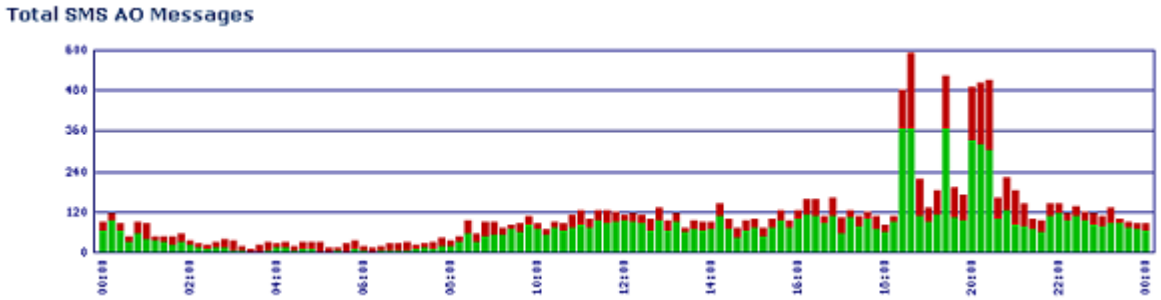


Figure 39: Sample statistical counters

For a complete overview of the AMS counters, refer to *AMS Counters Reference*.

9.2 Statistical Reporting

9.2.1 Key Performance Indicators

The following key performance indicators are available in statistical reports:

- Storage (usage percentage)
- Delivery statistics per queue
- Network topology statistics (per switch and per day/hour)
- Average success rates per delivery attempt
- Average retry factor
- Successfully delivered messages
- Buffered messages
- Deleted messages
- Peak traffic per time period
- Average traffic per time period
- Overall traffic
- Short codes (application traffic)
- Traffic per country

9.2.2 Delivery Success Rates

In a first delivery configuration, RTR statistics must be combined with AMS statistics to provide a complete overview. Depending on the first delivery success rate, which in most networks is as high as 80 percent, only 20 percent of all incoming MO and AO messages must be stored for second and further delivery.

Therefore, in practice, only one-fifth of the total network capacity arrives in the AMS (those messages that failed during the initial delivery attempt).

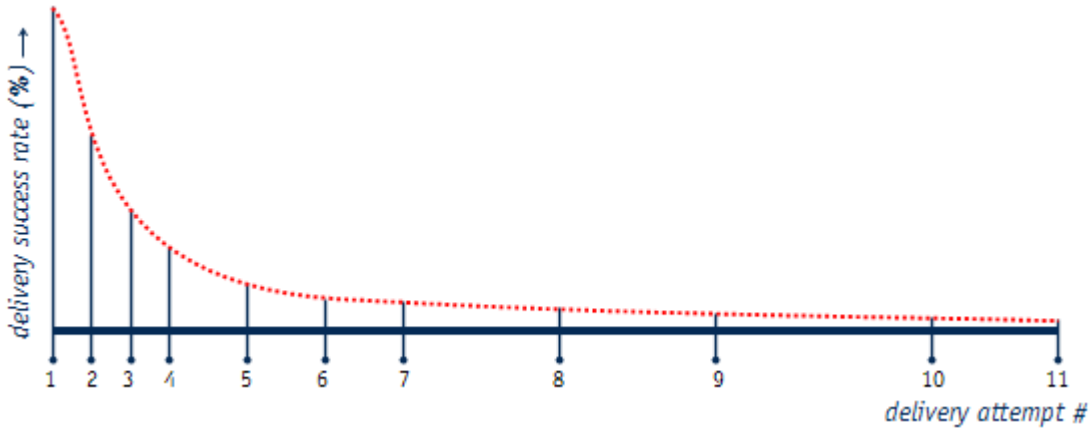


Figure 40: Delivery success rates

Based on delivery success rates and the incoming message rate (M), the average number of MT messages can be calculated:

$MT = M (1 + (1-p_{s1}) + (1-p_{s2}) + (1-p_{s3}) + (1-p_{s4}) + (1-p_{s5}) + (1-p_{s6}) + (1-p_{s7}) + (1-p_{s8}) + (1-p_{s9}) + (1-p_{s10}) + (1-p_{s11}))$

This average number of MT messages is an important factor in calculating the number of MT messages.

Also, based on this information, the average number of messages stored in the AMS can be calculated:

$MS = M ((1-p_{s1})t_1 + (1-p_{s1}) (1-p_{s2})t_2 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3})t_3 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4})t_4 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5})t_5 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5}) (1-p_{s6})t_6 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5}) (1-p_{s6}) (1-p_{s7})t_7 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5}) (1-p_{s6}) (1-p_{s7}) (1-p_{s8})t_8 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5}) (1-p_{s6}) (1-p_{s7}) (1-p_{s8}) (1-p_{s9})t_9 + (1-p_{s1}) (1-p_{s2}) (1-p_{s3}) (1-p_{s4}) (1-p_{s5}) (1-p_{s6}) (1-p_{s7}) (1-p_{s8}) (1-p_{s9}) (1-p_{s10})t_{10})$

Where MS is the total number of messages.

Note: In practice, additional margins must be calculated for network outages and applications that connect irregularly.

9.3 AMS Histogram Counters

The AMS contains histogram counters that provide statistical information. This section explains how to analyse these counters:

- `histogramCntQueues`
- `histogramCntDelivered`
- `histogramQueuesCntDelivered`
- `histogramCntStorageDuration`
- `histogramQueueCntStorageDuration`

9.3.1 histogramCntQueues

The `histogramCntQueues` counter provides the number of queues that have pending messages. The `histogramConfigSize` object provides the number of messages that are pending.

For example, the following command:

```
tp_walk --tp_ams histogramCntQueues
```

Provides the following output:

```
histogramCntQueues.1 = Gauge32: 32103
histogramCntQueues.2 = Gauge32: 5793
histogramCntQueues.3 = Gauge32: 1919
```

And the following command:

```
tp_walk --tp_ams histogramConfigSize
```

Provides the following output:

```
histogramConfigSize.1 = Gauge32: 1
histogramConfigSize.2 = Gauge32: 2
histogramConfigSize.3 = Gauge32: 3
```

Therefore, the AMS has:

- 32103 queues with one message pending
- 5793 queues with two messages pending
- 1919 queues with three message pending

9.3.2 histogramCntDelivered

The `histogramCntDelivered` counter provides the number of messages delivered for a particular number of delivery attempts, for the entire AMS (for example, the number of messages that were delivered in two attempts). The `histogramConfigNumberOfAttempts` object provides the number of attempts.

For example, the following command:

```
tp_walk --tp_ams histogramCntDelivered
```

Provides the following output:

```
histogramCntDelivered.1 = Counter32: 17452850
histogramCntDelivered.2 = Counter32: 1241051
histogramCntDelivered.3 = Counter32: 418071
```

And the following command:

```
tp_walk --tp_ams histogramConfigNumberOfAttempts
```

Provides the following output:

```
histogramConfigNumberOfAttempts.1 = Gauge32: 1
histogramConfigNumberOfAttempts.2 = Gauge32: 2
histogramConfigNumberOfAttempts.3 = Gauge32: 3
```

Therefore:

- 17,452,850 messages were delivered in one attempt
- 12,41,051 messages were delivered in two attempts
- 418,071 messages were delivered in three attempts

9.3.3 histogramQueueCntDelivered

The `histogramQueueCntDelivered` counter provides the number of messages delivered for a particular number of delivery attempts, per AMS queue (for example, the number of messages that were delivered in two attempts). The `histogramConfigNumberOfAttempts` object provides the number of attempts.

For example, the following command:

```
tp_walk --tp_ams histogramQueueCntDelivered
```

Provides the following output:

```
histogramQueueCntDelivered.1.1 = Counter32: 850
histogramQueueCntDelivered.1.2 = Counter32: 147
histogramQueueCntDelivered.1.3 = Counter32: 52
histogramQueueCntDelivered.2.1 = Counter32: 1241
histogramQueueCntDelivered.2.2 = Counter32: 51
histogramQueueCntDelivered.2.3 = Counter32: 12
histogramQueueCntDelivered.3.1 = Counter32: 41
histogramQueueCntDelivered.3.2 = Counter32: 87
histogramQueueCntDelivered.3.3 = Counter32: 18
```

And the following command:

```
tp_walk --tp_ams histogramConfigNumberOfAttempts
```

Provides the following output:

```
histogramConfigNumberOfAttempts.1 = Gauge32: 1
histogramConfigNumberOfAttempts.2 = Gauge32: 2
histogramConfigNumberOfAttempts.3 = Gauge32: 3
```

And the following command:

```
tp_walk --tp_ams queueName
```

Provides the following output:

```
queueName.1 = STRING: default MobileStation
queueName.2 = STRING: default Application
queueName.3 = STRING: VIP MobileStation
```

Therefore, for queue 1, which is called "default MobileStation":

- 850 messages were delivered in one attempt
- 147 messages were delivered in two attempts
- 52 messages were delivered in three attempts

For queue 2, which is called "default Application":

- 1241 messages were delivered in one attempt
- 51 messages were delivered in two attempts
- 12 messages were delivered in three attempts

For queue 3, which is called "VIP MobileStation":

- 41 messages were delivered in one attempt
- 87 messages were delivered in two attempts
- 18 messages were delivered in three attempts

9.3.4 histogramCntStorageDuration

The `histogramCntStorageDuration` counter provides the number of messages stored for a period of time, for the entire AMS. The queues in this counter refer to the recipient buffers, not to the queues that are defined in the Manager. The `histogramConfigDuration` object provides the time period (in seconds).

For example, the following command:

```
tp_walk --tp_ams histogramCntStorageDuration
```

Provides the following output:

```
histogramCntStorageDuration.1 = Counter32: 139583
histogramCntStorageDuration.2 = Counter32: 91353
histogramCntStorageDuration.3 = Counter32: 242137
```

And the following command:

```
tp_walk --tp_ams histogramConfigDuration
```

Provides the following output:

```
histogramConfigDuration.1 = Gauge32: 1
histogramConfigDuration.2 = Gauge32: 2
histogramConfigDuration.3 = Gauge32: 3
```

Therefore, the AMS has:

- 139583 messages stored for 0-1 seconds
- 91353 messages stored for 1-2 seconds
- 242137 messages stored for 2-3 seconds

9.3.5 histogramQueueCntStorageDuration

The `histogramQueueCntStorageDuration` counter provides the number of messages stored for a period of time, per AMS queue. The queues in this counter refer to the recipient buffers, not to the queues that are defined in the Manager. The `histogramConfigDuration` object provides the time period (in seconds).

For example, the following command:

```
tp_walk --tp_ams histogramQueueCntStorageDuration
```

Provides the following output:

```
histogramQueueCntStorageDuration.1.1 = Counter32: 13119
histogramQueueCntStorageDuration.1.2 = Counter32: 91070
histogramQueueCntStorageDuration.1.3 = Counter32: 241919

histogramQueueCntStorageDuration.2.1 = Counter32: 126444
histogramQueueCntStorageDuration.2.2 = Counter32: 270
histogramQueueCntStorageDuration.2.3 = Counter32: 160

histogramQueueCntStorageDuration.3.1 = Counter32: 0
histogramQueueCntStorageDuration.3.2 = Counter32: 2
histogramQueueCntStorageDuration.3.3 = Counter32: 14
```

And the following command:

```
tp_walk --tp_ams histogramConfigDuration
```

Provides the following output:

```
histogramConfigDuration.1 = Gauge32: 1
histogramConfigDuration.2 = Gauge32: 2
histogramConfigDuration.3 = Gauge32: 3
```

And the following command:

```
tp_walk --tp_ams queueName
```

Provides the following output:

```
queueName.1 = STRING: default MobileStation
queueName.2 = STRING: default Application
queueName.3 = STRING: VIP MobileStation
```

Therefore, for queue 1, which is called "default MobileStation", there are:

- 13,119 messages stored for 0-1 seconds
- 91,070 messages stored for 1-2 seconds
- 241,919 messages stored for 2-3 seconds

For queue 2, which is called "default Application", there are:

- 126,444 messages stored for 0-1 seconds
- 270 messages stored for 1-2 seconds
- 160 messages stored for 2-3 seconds

For queue 3, which is called "VIP MobileStation", there are:

- 0 messages stored for 0-1 seconds
- 2 messages stored for 1-2 seconds
- 14 messages stored for 2-3 seconds

9.3.6 histogramConfigDuration

The `histogramConfigDuration` object provides the time interval (in seconds). The following table shows the 100 interval values for `histogramConfigDuration`.

Note that the changes in interval occur at : 1, 10, 60, 300, 900, 3600, 21600, 43200, and 86400.

histogramConfigDuration	Values (in seconds)	Interval (in seconds)
histogramConfigDuration.1	1	1
histogramConfigDuration.2	2	1
histogramConfigDuration.3	3	1
histogramConfigDuration.4	4	1
histogramConfigDuration.5	5	1
histogramConfigDuration.6	6	1
histogramConfigDuration.7	7	1
histogramConfigDuration.8	8	1
histogramConfigDuration.9	9	1
histogramConfigDuration.10	10	1

histogramConfigDuration	Values (in seconds)	Interval (in seconds)
histogramConfigDuration.11	15	5
histogramConfigDuration.12	20	5
histogramConfigDuration.13	25	5
histogramConfigDuration.14	30	5
histogramConfigDuration.15	35	5
histogramConfigDuration.16	40	5
histogramConfigDuration.17	45	5
histogramConfigDuration.18	50	5
histogramConfigDuration.19	55	5
histogramConfigDuration.20	60	5
histogramConfigDuration.21	75	15
histogramConfigDuration.22	90	15
histogramConfigDuration.23	105	15
histogramConfigDuration.24	120	15
histogramConfigDuration.25	135	15
histogramConfigDuration.26	150	15
histogramConfigDuration.27	165	15
histogramConfigDuration.28	180	15
histogramConfigDuration.29	195	15
histogramConfigDuration.30	210	15
histogramConfigDuration.31	225	15
histogramConfigDuration.32	240	15
histogramConfigDuration.33	255	15
histogramConfigDuration.34	270	15
histogramConfigDuration.35	285	15
histogramConfigDuration.36	300	15
histogramConfigDuration.37	360	60
histogramConfigDuration.38	420	60
histogramConfigDuration.39	480	60
histogramConfigDuration.40	540	60
histogramConfigDuration.41	600	60

histogramConfigDuration	Values (in seconds)	Interval (in seconds)
histogramConfigDuration.42	660	60
histogramConfigDuration.43	720	60
histogramConfigDuration.44	780	60
histogramConfigDuration.45	840	60
histogramConfigDuration.46	900	60
histogramConfigDuration.47	1080	180
histogramConfigDuration.48	1260	180
histogramConfigDuration.49	1440	180
histogramConfigDuration.50	1620	180
histogramConfigDuration.51	1800	180
histogramConfigDuration.52	1980	180
histogramConfigDuration.53	2160	180
histogramConfigDuration.54	2340	180
histogramConfigDuration.55	2520	180
histogramConfigDuration.56	2700	180
histogramConfigDuration.57	2880	180
histogramConfigDuration.58	3060	180
histogramConfigDuration.59	3240	180
histogramConfigDuration.60	3420	180
histogramConfigDuration.61	3600	180
histogramConfigDuration.62	4500	900
histogramConfigDuration.63	5400	900
histogramConfigDuration.64	6300	900
histogramConfigDuration.65	7200	900
histogramConfigDuration.66	8100	900
histogramConfigDuration.67	9000	900
histogramConfigDuration.68	9900	900
histogramConfigDuration.69	10800	900
histogramConfigDuration.70	11700	900
histogramConfigDuration.71	12600	900
histogramConfigDuration.72	13500	900

histogramConfigDuration	Values (in seconds)	Interval (in seconds)
histogramConfigDuration.73	14400	900
histogramConfigDuration.74	15300	900
histogramConfigDuration.75	16200	900
histogramConfigDuration.76	17100	900
histogramConfigDuration.77	18000	900
histogramConfigDuration.78	18900	900
histogramConfigDuration.79	19800	900
histogramConfigDuration.80	20700	900
histogramConfigDuration.81	21600	900
histogramConfigDuration.82	25200	3600
histogramConfigDuration.83	28800	3600
histogramConfigDuration.84	32400	3600
histogramConfigDuration.85	36000	3600
histogramConfigDuration.86	39600	3600
histogramConfigDuration.87	43200	7200
histogramConfigDuration.88	50400	7200
histogramConfigDuration.89	57600	7200
histogramConfigDuration.90	64800	7200
histogramConfigDuration.91	72000	7200
histogramConfigDuration.92	79200	7200
histogramConfigDuration.93	86400	7200
histogramConfigDuration.94	172800	86400
histogramConfigDuration.95	259200	86400
histogramConfigDuration.96	345600	86400
histogramConfigDuration.97	432000	86400
histogramConfigDuration.98	518400	86400
histogramConfigDuration.99	604800	86400
histogramConfigDuration.100	691200	86400

Chapter 10

Configuration

Topics:

10.1 Introduction

The AMS has a distributed architecture and central configuration management. The AMS configuration has two parts:

- Semi-static configuration that defines fundamental AMS parameters such as TCP/IP addressing, database locations, and replication parameters.
- Dynamic configuration that defines SMS routing parameters such as SMSCs, applications, and routing rules.

Both configuration types are XML-based and are described in this chapter.

Note: It is possible to start the AMS even if no configuration files are present and the AMS device has not been added to the MGR. However, because the AMS will not receive any configuration from the MGR, it will not operate properly or be aware of other devices in the network.

10.2 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 AMS. The semi-static configuration files are configured directly in XML.

The semi-static AMS configuration files contain:

- Discovery addresses
- Database storage type
- Number of replica databases
- Database directories
- Database properties

The semi-static configuration consists of two files:

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

10.2.1 tpconfig Entity

This section describes the `tpconfig` attributes.

10.2.1.1 amsdbcachesize

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines the amount of memory used as database cache in megabytes (range 1-16384, default 350). Any cache size smaller than 500 MB is automatically increased by 25% to account for buffer pool overhead; cache sizes larger than 500 MB are used as specified. Can only be set when the device is inactive.

10.2.1.2 amsdbcheckpointsize

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines the maximum number of transactions (range 1-10000000, default 150000) after which a database checkpoint is made.

10.2.1.3 amsdbcheckpointtime

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines the maximum time in seconds (range 1-86400, default 3600) after which a database checkpoint is made.

10.2.1.4 amsdbdatabasedirectorymaster

Mandatory/Optional

Mandatory

Location

Common or host-specific configuration file

Description

Defines the directory used for storing the master message database. Can not be the same path as specified for `amsdbdatabasedirectoryreplica` or `amsdbtransactiondirectoryreplica`. May only be changed when the deviceAdminState is disabled. Default `/var/TextPass/MessageStore/master`.

Note: If multiple instance of AMS is running on same server then `amsdbdatabasedirectorymaster` must be specified in host-specific configuration file and two ZMM users must not use the same `amsdbdatabasedirectorymaster`.

10.2.1.5 amsdbdatabasedirectoryreplica

Mandatory/Optional

Mandatory

Location

Common or host-specific configuration file

Description

Defines the directory used for storing the replica message databases. Can not be the same path as specified for `amsdbdatabasedirectorymaster` or `amsdbtransactiondirectorymaster`. May only be changed when the deviceAdminState is disabled. Default `/var/TextPass/MessageStore/replica`.

Note: If multiple instance of `AMS` is running on same server then `amsdbdatabasedirectoryreplica` must be specified in host-specific configuration file and two ZMM users must not use the same `amsdbdatabasedirectoryreplica`.

10.2.1.6 amsdbmaxsequentialwrites

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines the maximum number of sequential write operations (default 5) when flushing dirty pages from the cache. When set to 0, no limitation is enforced.

10.2.1.7 amsdbremovetransactionlogs

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines whether completed transaction log files are automatically removed (default true).

10.2.1.8 amsdbstorageengine

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines how data is stored in the Berkeley DB. This can be `btree` or `hash`. Previous AMS version only used the `btree` storage mechanism. The `hash` storage mechanism was introduced to achieve a higher performance.

Existing setups must first convert the database before changing this setting. See OM on how to convert the existing database from `btree` to `hash`.

WARNING: Changing the AMS database storage engine could lead into the situation that the database cannot be read. Changing it without a conversion, could cause the database not being able to be opened. The error in system log is:

```
DAL_open BDB error: Invalid argument
```

Therefore, all the databases .fcl, master and replicas must be changed to the corresponding types if conversion is needed.

Values

- `btree` (default)
- `hash`

10.2.1.9 amsdbtransactiondirectorymaster

Mandatory/Optional

Mandatory

Location

Common or host-specific configuration file

Description

Defines the directory used for storing the TransactionLog records of the master message database. Can be the same path as specified for `amsdbdatabasetransactiondirectorymaster`. May only be changed when the deviceAdminState is disabled. Default is `/var/TextPass/MessageStore/master`.

Note: If multiple instances AMS are running on same server then `amsdbtransactiondirectorymaster` must be specified in host-specific configuration file and also it must be ensure that directory for each ZMM users are unique.

10.2.1.10 amsdbtransactiondirectoryreplica

Mandatory/Optional

Mandatory

Location

Common or host-specific configuration file

Description

Defines the directory used for storing the TransactionLog records of the message replica databases. Can be the same path as specified for `amsdbtransactiondirectoryreplica`. May only be changed when the deviceAdminState is disabled. Default `/var/TextPass/MessageStore/replica`.

Note: If multiple instance of AMS is running on same server then `amsdbtransactiondirectoryreplica` must be specified in host-specific configuration file and two ZMM users must not use the same `amsdbtransactiondirectoryreplica`.

10.2.1.11 amsdbwritepause

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Defines the number of milliseconds (default 20) to pause before further write operations.

10.2.1.12 amsdefaultvalidity

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates the default number of hours (range 1-2232, default 72) that a message is stored before it expires. This value is applied to messages that do not specify a validity period itself. The actual validity period will be shorter if a shorter period is specified in the message itself, or the applicable delivery scheme has a shorter maximum validity period.

10.2.1.13 amsfailoverholdofftime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds (default 60) that the AMS will wait before taking the active replica role for a failed peer AMS node (failover).

10.2.1.14 amsmasterstoragetype

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the storage type for the master message store (volatile or nonvolatile, default nonvolatile). Choosing a non-volatile storage type will ensure that messages are securely stored on disk before an ACK is returned to the RTR. Volatile storage is faster, but cannot be restored after a system outage. This type of storage should only be used if loss of messages after outage or shutdown is acceptable. May only be changed when the deviceAdminState is disabled.

10.2.1.15 amsmaxdeliveryrate

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the maximum number of delivery attempts per second (range 1-5000; default 150) that the AMS is allowed to perform.

10.2.1.16 amsmaxdeliveryresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds (range 0-3600, default 100) an AMS node waits for a response from a SS7 or application gateway after sending a MXP message.

If a SS7 gateway attempts a MT delivery, it will only respond to the AMS after that attempt is completed.

10.2.1.17 amsmaximumdeferreddelivery

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates the maximum number of hours (default 168) a message may be stored before its first delivery attempt. If the difference between the deferred delivery time specified in a store request and the current time exceeds this value, the message is refused.

10.2.1.18 amsmaximumvalidity

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates the maximum number of hours (range 1-2232, default 168) that a message is stored before it expires. The actual validity period will be shorter if a shorter period is specified in:

- The message itself, or
- The applicable delivery scheme

10.2.1.19 amsmaxnumberofmessages

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the maximum number of messages that a single AMS node can hold, including the replicas. The hard limit is 40 million messages, which will override any higher value specified in the license. The default is the value specified by the license.

10.2.1.20 amsmaxreplicateresponsetime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds (range 0-3600, default 5) the AMS waits for a response on a replication request to another AMS node.

10.2.1.21 amsmaxreplicationrate

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the maximum number of replication requests per second (range 1-3000, default 2000) that the AMS is allowed to send to each replica node.

10.2.1.22 amsmaxshutdowndelaytime

Mandatory/Optional

Optional

Location

Common configuration file

Description

The maximum number of seconds (default 20) that the AMS will wait for pending deliveries and replication attempts, after a shutdown command is received.

10.2.1.23 amsmaxterminationindications

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of TerminationIndications (default 10) that may be sent consecutively by the AMS to a RTR.

10.2.1.24 amsmessagestorethreshold1

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.25 amsmessagestorethreshold2

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.26 amsmessagestorethreshold3

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.27 amsmessagestorethreshold4

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.28 amsmessagestorethreshold5

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.29 amsnetworkalertdelaytime

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

The time the AMS will wait for delivery after a network alert has been received for an MT message. A value of 0 (default) will cause a direct delivery attempt.

10.2.1.30 amsnumberofreplicas

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the number of nodes each message is replicated to (range 0-4, default 0, no replication). Value is taken from the license by default and can never be higher than the one set in the license. Replicating messages will decrease AMS nodes' performance.

10.2.1.31 amsoverloadallowedforreplication

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines whether AMS replication is allowed to overload other AMS instances in case less number of instances is running on other servers. For example, three AMS instances are running on server 1, and

one AMS instance is running on the other server. This could send three times traffic to one AMS running on other node. This parameter can enable or disable that behavior. Note that this parameter is applicable only for multi-instanced and VM-based deployments.

Default is `false`.

10.2.1.32 amspendinginmemoryconversions

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the maximum number of pending in-memory conversions (used only for messages requiring a conversion). A in-memory conversion record is valid for a full message (so all segments share the same information). If set too low, the conversion functionality stays in place but it may be less efficient. In-memory conversion records are cleaned up once the message is fully delivered, meaning the limits are based on the traffic and average delays.

Please refer to the counters `amsCntPendingInMemoryConversions` and `amsCntInMemoryConversionLimitReached` in *AMS Counters Reference* for more insight in usage.

Default Value

30000

10.2.1.33 amsphyserverid

Mandatory/Optional

Optional

Location

Host-specific configuration file

Description

A unique identifier assigned to each physical server, when a VM-based deployment is used, i.e. a single AMS instance is running on a VM and multiple VMs running on each physical server. In such a scenario, this parameter must be configured in the Host-specific configuration files of individual AMS instances and must be set to a value in the range of 1-255, while ensuring that all AMS instances located on the same physical server are assigned the same value for this parameter.

Note: This parameter must be 0 (default value) for all AMS instances when they are not deployed on VMs, i.e. when either a single instance or multiple instances (not VMs) are running on each physical server.

10.2.1.34 amspurgedeletedelaytime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds (default 360) before the AMS actually deletes its purge list after synchronisation.

10.2.1.35 amsqueuefordialoutatstatusreports

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies the AMS queue in which to store AT status reports that may only be delivered on dial-out sessions (range 1-1000, default 2). If many AT status reports need to be delivered using dial-out, it is strongly recommended that a specific AMS queue be configured for this type of report, as this enables AMS to deliver these status reports faster.

10.2.1.36 amsqueueforexternalconditionfailuremessages

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies the AMS queue in which to store messages that terminate in mobiles and that are sent as a result of an external condition that did not satisfy (range 1-1000, default 1).

10.2.1.37 amsqueueformtstatusreports

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies the AMS queue in which to store MT status reports (range 1-1000, default 1).

10.2.1.38 amsqueueforregularatstatusreports

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies the AMS queue in which to store AT status reports that may be delivered on sessions established by the application (range 1-1000, default 2).

10.2.1.39 amsqueuestorethreshold1

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.40 amsqueuestorethreshold2

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.41 amsqueuestorethreshold3

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.42 amsqueuestorethreshold4

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.43 amsqueuestorethreshold5

Mandatory/Optional

Optional

Location

Common configuration file

Description

A threshold at which a trap will be generated if the total number of messages stored in the AMS crosses this (shrinking) or reaches it (growing). When 0 (default), no traps will be generated.

10.2.1.44 amsreplacedconcatenateddelaytime

Mandatory/Optional

Optional

Location

Common configuration file

Description

The maximum number of seconds (default 10) that the AMS will wait for delivery after a concatenated message has been replaced.

10.2.1.45 amsrecipientbufferblackoutperiod

Mandatory/Optional

Optional

Location

Common configuration file

Description

Blackout in seconds if the recipient buffer fails to send out a message to any Router because of the system/network error. The blackout period will reduce the chance when the situation is further deteriorated because of the immediate retry attempts.

Default is 0, meaning immediate retry is triggered without blackout for this recipient buffer. The value range is [0, 60].

Note: The blackout period is triggered after the network/system issue already happened. It reduces the chance of deterioration of the existing issue, but it cannot avoid the issue like congestion from happening. Such system/network issues can be triggered by many reasons. Performance tuning is needed to avoid such system/network issues.

10.2.1.46 amsparalleldeliveryminimuminterval

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates the minimum interval in seconds of doing parallel delivery (range 0-1440, default 0). Message can only be attempt again after the interval. 0 means the parameter is not used so that there is no limit to the next attempt time.

Note: The parameter shall be configured to a value that is smaller than the minimum configured interval in the delivery scheme.

10.2.1.47 amsreplicastoragetype

Mandatory/Optional

Optional

Location

Common configuration file

Description

Defines the storage type for the replica message stores (volatile or nonvolatile, default nonvolatile). Choosing a non-volatile storage type will ensure that messages are securely stored on disk before an ACK is returned to the replicating AMS. Volatile storage is faster, but cannot be restored after a system outage. This type of storage should only be used if loss of messages after outage or shutdown is acceptable. May only be changed when the deviceAdminState is disabled.

10.2.1.48 amsrouterdelaytime

Mandatory/Optional

Optional

Location

Common configuration file

Description

The time in seconds (default 60 seconds) between detection of the join of the first RTR and the moment the AMS starts delivering messages. The delay can be longer if an AMS node joins while waiting. Default is 60 seconds.

10.2.1.49 amsschedulerholdofftime

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of seconds (default 600) that the scheduler runs at a reduced speed after start up. After this time the maximum number of delivery attempts is increased each minute until the normal limit is reached.

10.2.1.50 amsschedulerincompletethreshold

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates the threshold for raising the schedulingIncomplete trap. When the delivery scheduler is not able to serve all pending queues for more than this number of seconds (range 0-3600, default 30), the trap is sent.

10.2.1.51 amsschedulerreducedrate

Mandatory/Optional

Optional

Location

Common configuration file

Description

Number of delivery attempts (default 40) that the scheduler triggers during the "hold-off" period after start up (see `amsschedulerholdofftime`). If the reduced rate is higher than `amsmaxdeliveryrate/5`, then the latter is used.

10.2.1.52 amsstatusupdatemaster

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Indicates if and how the master message status (delivery status, number of attempts, error code) should be updated after a failed delivery attempt. Updating a message's status is at least as performance-intensive as storing a new message. Status updates should be minimized for better system performance. Not having the latest message status on switch-over to a replica will make the delivery schemes restart. Values:

- never: No updates are made in the message store itself, only the internal memory structures of the master are maintained (best performance).

- always (default): After each failed delivery attempt the status is updated in the message store.
- smart: Updates are only made when the system load allows it. This value does not affect a master AMS.

10.2.1.53 amsstatusupdatereplica

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates how the messages status in the replica store is updated after a failed delivery attempt. This update-mode defines how the message status (i.e. number of attempts, last error) is updated in the master or replica store, after a failed delivery attempt. Performing updates in the store(s) should be minimized for better performance of the system. Not having the latest message status on switch-over to a replica will make the delivery schemes restart.

Possible values are:

- never (default): No updates are made in the message store itself, only the internal memory structures of the master are maintained.
- always: After each failed delivery attempt the status is updated in the message store.
- smart: Only when the system load allows it, updates are made in the message store.

10.2.1.54 amsstoragemode

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies the AMS mode:

- messagestore: The AMS should only act as a message store
- transactionstore: The AMS should only act as an Icache
- messageandtransactionstore: The AMS should act as a message store and as an Icache

10.2.1.55 amsupdateresponses

Mandatory/Optional

Optional

Location

Common configuration file

Description

Indicates if the AMS must send a response on the UpdateSmRequest from another AMS (default never).

10.2.1.56 dbproplasterrorstring

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

May be used to identify the last database error that occurred.

10.2.1.57 enabledialoutnotificationasamsmessagetype

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

Specifies whether the RTR generates AMS store requests with message type dialoutNotification. To support AO messages that require a dial-out delivery notification (DDN), set this attribute to true (default false).

10.2.1.58 pairedunicodecharlist

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description
Specifies paired unicode characters that should not be split between consecutive segments of a concatenated message.

Valid Values
A sequence of hexadecimal values separated by semicolons (';'). The values should be encoded in UTF-16 format. The string must end with ';' if it is not empty.

Default

002320E3;003120E3;003220E3;003320E3;003420E3;003520E3;003620E3;003720E3;003820E3;003920E3;003020E3;

10.2.1.59 processpriority

Mandatory/Optional

Optional

Location

Host-specific configuration file

Description

Specifies the priority of the process started.

Valid Values

- critical
- high
- normal
- low

Default

normal

10.2.1.60 runqclidprocess

Mandatory/Optional

Mandatory (for running Q-CLI)

Location

Host-specific configuration file

Description

Indicates whether the Q-CLI process should be started on this node. Should be "true" for running Q-CLI

Valid Values

- true
- false

Default

true

10.2.1.61 runtextamsprocess

Mandatory/Optional

Mandatory (for running the AMS)

Location

Host-specific configuration file

Description

Indicates whether the AMS process should be started on this node. Should be "true" for running the AMS

Valid Values

- true
- false

Default

false

10.2.1.62 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

10.2.1.63 runtextpassprocess

Mandatory/Optional

Mandatory (for running the RTR)

Location

Host-specific configuration file

Description

Specifies if the RTR process should be started. Should be "true" for running the RTR.

Valid Values

- true
- false

10.2.1.64 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

10.2.1.65 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

10.2.1.66 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

10.2.1.67 amsatextrusion

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description

This parameter is used to enable or disable AT extrusion for AT parallel delivery.

Valid Values

- true
- false

Default

false

10.2.1.68 amsatextrusionreuselastinterval

Mandatory/Optional

Optional

Location

Common/Host configuration file

Description

This parameter is used to enable or disable AT extrusion reuse last configured interval for AT parallel delivery.

Valid Values

- true
- false

Default

false

10.2.2 trapreceiver Entity

This section describes the `trapreceiver` attributes.

10.2.2.1 ipaddress

Mandatory/Optional

Mandatory

Location

Host-specific or Common configuration file

Description

IP address (IPv4 or IPv6) or Hostname of the trap receiver.

10.2.2.2 udpport

Mandatory/Optional

Optional

Location

Host-specific or Common configuration file

Description

UDP port on the trap receiver to which traps are sent.

10.2.3 amsparalleldelivery Entity

This section contains the settings for parallel delivery applications. If a message service type matches configured service type, then the parallel delivery scheme will be using with the given settings. General remarks:

1. Enabling the entry changes (e.g. *initialstart* from 0 to 10) are allowed. They will generally only take effect when a message are stored.
2. Disabling the entry changes will take effect upon a next attempt caused by initial start.
3. Only the first matched service type in a recipient queue will be stored and its parameters are used. Until that queue becomes empty again.
4. The feature is applicable to both AT delivery in case of AO-ST-AT and AT notification in case of AO-ST-MT, AO-MT and AO-ST-AT. This feature is only applicable to SMPP protocol.

10.2.3.1 description

Mandatory/Optional

Optional

Location

Host-specific or common configuration file

Description

Optional description of the parallel delivery application (for reference only).

10.2.3.2 smppservicetype

Mandatory/Optional

Mandatory

Location

Host-specific or common configuration file

Description

The SMPP service type for a given SMPP application. The parallel delivery is enabled when the this service type is matched for a receiving application. It cannot be changed if the entry is active. An empty `smppservicetype` will disable the entry.

Default Value

Empty string

10.2.3.3 initialstart

Mandatory/Optional

Mandatory

Location

Host-specific or common configuration file

Description

The max amount of deliveries which can be started initially. This is not limit the subsequent deliveries up to the max window size. An initial start of 0 will fall-back to normal behavior.

Valid Values

0 - 255

Default Value

50

10.2.3.4 rate

Mandatory/Optional

Mandatory

Location

Host-specific or common configuration file

Description

The rate to start new initial deliveries. Thus, new messages are delivered with this rate until the *initialstart* limit is pending. This is not the exact timing and some rates may be rounded.

Valid Values

1 - 50

Default Value

10

10.2.4 whitelist Entity

The `whitelist` entity of the XML configuration file can have a number of subordinates for each trap facility. Supported facilities in the AMS are `gen` and `ams`. Each can have the attribute `trap`, which is the name of a trap or the wildcard (`*`).

10.2.5 blacklist Entity

The `blacklist` entity of the XML configuration file can have a number of subordinates for each trap facility. Supported facilities in the AMS are `gen` and `ams`. Each can have the attribute `trap`, which is the name of a trap or the wildcard (`*`).

10.2.6 postbootscript Entity

This section describes the `postbootscript` attributes.

10.2.6.1 command

Mandatory/Optional

Mandatory

Location

Common configuration file

Description

A UNIX command.

10.2.7 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".

ZephyrTel 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 ZephyrTel 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.

10.2.8 Activating Configuration Files

To activate the configuration files, execute the following command at the command prompt of each AMS:

```
tp_config [<specific file> [<common file>]]
```

If you do not specify the configuration files in the command, `tp_config` uses both of them. Note that:

- Executing `tp_config` restarts the RTR, which is service-affecting. Therefore, it is recommended that the configuration files be activated during low-traffic hours.
- The `tp_start --tp_ams` command-line tool, which starts the AMS, automatically executes `tp_config`.

10.3 Dynamic Configuration

The dynamic configuration is called dynamic because, in general, the parameters change frequently. The dynamic configuration defines AMS parameters that determine the SMS message flow. These parameters are related to the AMS delivery behavior. The dynamic configuration is configured in the MGR, which is a Web interface.

The dynamic AMS configuration files contain:

- AMS devices
- Message queues
- Delivery schemes
- Error-dependent delivery schemes

Refer to the MGR Operator Manual for more information about the dynamic configuration.

Chapter 11

OAM Interface (SNMP)

Topics:

11.1 Introduction

The AMS uses the Simple Network Management Protocol (SNMP) to configure and monitor interfaces, system statuses, and settings. SNMP is a widely used industry standard for managing and configuring network components.

All statistical and configuration information (even internal values) that can be configured and/or viewed with SNMP are described in the Management Information Base (MIB) files (`*.my`).

11.2 MIB Files

The following MIB files apply to the AMS:

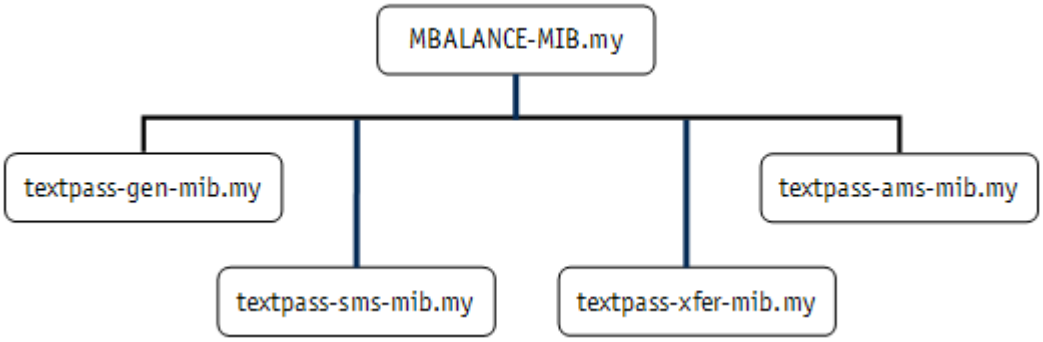


Figure 41: AMS MIBs

Each MIB 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 in the AMS system. The MIB files are located in `/usr/local/share/snmp/mibs/`.

11.3 SNMP Manager

For configuration and monitoring purposes, an SMNP Manager or Management Station issues SNMPv1 requests to the AMS. SNMP Managers should send such requests to UDP port 11661 of the AMS. The device does not enforce an SNMP Manager to originate requests from any specific UDP port (any UDP port can be used for this purpose).

The device requires an SNMP Manager to use a community string equal to:

- `public` for read operations
- `private` for set operations

Requests that do not satisfy these requirements are silently discarded.

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

11.4 Trap Service

Up to eight SNMP Managers can subscribe to the AMS trap service. When a trap condition occurs, the AMS 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
- 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 AMS always originates SNMP traps from UDP port 11662 and terminates them in the UDP ports that are specified in the Alarm Station Table. The community string that the AMS specifies in SNMP traps is always equal to `public`.

The AMS uses an SMNP trap daemon to log generated SNMP traps locally in `/var/log/messages`. The daemon uses 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.

11.5 SNMP Trap Reference

Refer to the ZephyrTel Mobile Messaging SNMP Trap Reference Guide for a complete overview of the Mobile Messaging SNMP traps. The SNMP Trap Reference Guide is available on the Mobile Messaging documentation CD.

11.6 System Management

The following system management-related features are provided with the AMS system:

- Central management:
  - Centralized management of all ZephyrTel Mobile Messaging nodes
  - Replication of configuration
  - Web-based Manager
  - Command-line interface
- Real time system monitoring and management:

- User definition of queues and delivery schemes
- Queue inspection and screening
- Purging of messages
- Special manual queue (triggered)
- On-line queue parameter control

- Billing:
  - CDR format from all major SMSC brands
  - Prepaid interface via IN or other (TCP/IP-based) protocols

11.7 Command-Line Interface

Basic message and Icache queue querying, viewing, and management commands are made available via the Query Command Line Interface (Q-CLI), which is executed as the command-line tool `tp_qcli`.

For information about Q-CLI, refer to *Query Command Line Interface*.

11.8 XML Interface

The XML interface enables you to manage the messages stored in the AMS (e.g. from the Customer Care department). Interaction is done by posting an XML message to the MGR (Web server). Each request can contain the following data:

- Authentication information to identify the posting entity
- A command such as `show`, `alert`, or `delete`
- The time period in which the message or record was first stored in the AMS
- One or more identifying attributes (all messages or records that match all attributes as defined in the specified command)
- An ID supplied by the client and returned by the server (optional)
- A path specifier, which the AMS uses to identify the request

The MGR's Web server supports HTTP/1.1 (RFC2068). The secure Web server supports SSL (v2/v3) and TLS (v1). This release supports XML version 1.0.

Chapter 12

Security

Topics:

12.1 Introduction

This chapter provides an overview of security aspects of ZephyrTel Mobile Messaging systems.

12.2 Controlling System Access

Access to ZephyrTel Mobile Messaging systems is controlled using the available Red Hat Enterprise Linux security mechanisms.

Refer to the Red Hat Enterprise Linux documentation at *http://www.redhat.com/docs/* for more information.

12.3 User Group Privileges

The MGR allows users to be assigned to groups. User group privileges are controlled using the **Settings ➤ User Admin ➤ Groups** screen.

Refer to the MGR Operator Manual for more information.

12.4 Detecting and Reporting Security Violations

Access to log files or audit files and other system resources on ZephyrTel Mobile Messaging systems is controlled using the available Red Hat Enterprise Linux security mechanisms.

Refer to the Red Hat Enterprise Linux documentation at *http://www.redhat.com/docs/* for more information.

Chapter 13

Software License

Topics:

13.1 Introduction

Some ZephyrTel Mobile Messaging software components are licensed features; therefore, the appropriate software licenses must be purchased before the corresponding functionality can be used.

This chapter discusses commercially licensed features, how to check licenses, and how to install new ZephyrTel Mobile Messaging licenses.

13.2 Licensed Items

The following AMS software components are licensed:

Item	Settings	M/O
Total number of messages stored (in millions); the maximum allowed total number of elementary messages stored, including status reports and notifications. This total includes the replicated messages.	1-40M	M
Ability	• 0: Message store • 1: Intermediate Cache (Icache) • 2: Message store and Icache	M
Total number of redundant nodes (replicas)	0-4	O
Scrambled storage	Enabled/Disabled	O
Error Dependent Delivery Schemes	Enabled/Disabled	O
Queue query interface	Enabled/Disabled	O
Query type queue	Enabled/Disabled	O
Originator type query	Enabled/Disabled	O
Hide User Data	Enabled/Disabled	O

Note: At least one AMS-related routing path on the RTR must be active (licensed); otherwise, no messages can be routed to the AMS.

13.2.1 Multi-Instance License

Multiple instances feature allows you to run multiple AMS (up to 10 instances) on the same node. Multi-instance license should be enabled for ZMM user to run one additional instance of AMS. To run one additional instance of AMS from newly created ZMM user (using script `tp_manage_user`), instance license for newly create user id (operating system user identifier) should be enabled in license file.

13.3 Checking Your License

To view the current license values, execute the following command at the command prompt:

```
tp_system [system]
```

Where `[system]` is the IP address or host name of the AMS.

The following is a sample of the output of the `tp_system` tool.

```
Identification:
    TextPass/AMS R01.11.04.00
    Linux 3.10.0-862.14.4.el7.x86_64  #1 SMP Fri Sep 21 09:07:21 UTC 2018
    Linux build

Uptime:
    0 days 02h:19m:00s

License key:
    XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX


License information:
    License version 9
    License number TST-93814
    Hardware ID 287567c3
    Ext Serial No SGH509XRXJ
    Instance Serial No 200
    License exceeds in 2122 hours
    License exceeds at Sun Mar 10 00:00:00 2019
    License issue number 17
    Hide User Data in Events disabled
    VmWare Support disabled
    Core Count 0
    Encrypt User Data disabled
    Scrambled Storage enabled
    1 Redundancy Nodes
    1000000 Total Buffer Size
    Queue Active Period              enabled
    Queue Dependant Delivery Schemes enabled
    Queue Query Interface            enabled
    Query Type Originator            enabled
    Query Type Queue                 enabled
    Query Type Hide User Data        disabled
    Queue Unsplit Surrogate Pairs    disabled
    AMS Ability                      messageAndTransactionStore

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
```

13.4 Activating a New License

A new license key is required to activate new services or adapt connectivity or performance settings. Please contact your ZephyrTel Mobile Messaging account manager to obtain a new license key.

To activate a new license key:

1. Place a valid AMS 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_ams --read_licensekey [system]
   ```

   Where `[system]` is the IP address or host name of the AMS.

   It is not required to restart the AMS when activating a new license; however, you should always restart the Manager 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_ams --show_licensekey [system]
   ```

   Where `[system]` is the IP address or host name of the AMS.

   If the AMS is started without a valid license, AMS initialisation will fail after a series of errors. After this, a valid license can be activated.

Note: Also ensure that a valid license is placed and activated on the MGR node as well.

13.5 License Warnings

The following SNMP traps warn when the license limits are approaching:

- `licenseWillExpire`—The temporary 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 days in advance
- `licenseWillExpire`—7 days in advance
- `licenseWillExpire`—3 days in advance
- `licenseWillExpire`—1 day 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 14

System Management

Topics:

14.1 Introduction

This chapter describes the command-line tools available to assist in determining the status of the system or locating the cause of any problems.

The AMS system is designed to run unattended and almost maintenance-free for normal operation.

14.2 Stopping the System

If the AMS process needs to be stopped, execute the following command at the command prompt of the AMS node:

```
tp_stop --tp_ams
```

This command will gracefully stop the AMS process and the watchdog process.

14.3 Starting the System

To start the AMS process, execute the following command at the command prompt of the AMS node:

```
tp_start --tp_ams
```

This command will start the AMS and the watchdog process.

When the AMS node restarts after an unplanned outage (such as a power failure), the AMS process is restarted automatically and resumes service.

As an alternative to using the `tp_start` tool, the AMS can be rebooted, after which it will automatically restart.

It is possible to start the AMS even if no configuration files are present and the device has not been added to the Manager. However, because the AMS will not receive any configuration from the Manager, it will not operate properly or be aware of other devices in the network.

It is possible to start the AMS even if no RTR is present in the network (or if the AMS does not detect the RTR(s) that are present), though the AMS will not receive or be able to send messages if this is the case.

14.3.1 Starting Two AMS Nodes

When AMS is working as two or more nodes, the second node should not be started until the first has been started and established (its ID is set usually to 0). When the second node is started, it will be established as node 1. Otherwise, both nodes may start as 0.

If both nodes are node 0, then the second node started will be disabled and a special command will be required to correct this.

Note: Manually setting the AMS ID should only be done after consulting ZephyrTel support.

To set the ID to 1, use:

```
tp_ams --stderr --amsid=1 --makemyday=yes
```

Then, start the AMS using:

```
tp_start --tp_ams
```

14.3.2 Operational States of AMS

When the AMS process is started, it first comes up in the “starting” state. The AMS process will change its state to “synching” from “starting” when the following conditions are satisfied:

1. The master database must be “open”, i.e. correctly initialized to handle the AMS transactions.
2. The semi-static parameter `amsmasterstoragetype` is set to "nonvolatile" (default) and the master database environment (i.e. a collection of configuration settings that BerkeleyDB uses) is properly set up.
3. The semi-static parameter `amsreplicastoragetype` is set to "nonvolatile" (default) and the replica database environment is properly set up.

The AMS will change its state to “operating” from “synching” when it is allocated a valid AMS ID. The current operational state of the AMS process can be checked using the following command:

```
tp_status --tp_ams
```

14.4 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`, execute:

```
# tail -f /var/log/messages
```

14.5 System Verification

14.5.1 Basic System Verification

For basic verification of the AMS status, execute the following command at the command prompt:

```
tp_system [system]
```

Where `[system]` is a resolvable host name or the IP address of the AMS. The response contains the time that the AMS process has been running. If there is no response, the specified system cannot be reached or the AMS is not running correctly.

For more information about command-line tools, refer to the Tools Operator Manual.

14.5.2 Advanced System Verification

For more detailed information about the AMS system, use the `tp_walk --tp_ams` command-line tool (as user textpass) to retrieve information about specific AMS counters. Useful attribute groups for AMS verification are:

- `amsCounters`
- `amsLoadCounters`
- `amsReplicateCounters`
- `amsProperties`
- `failoverControlTable`
- `queueTable`
- `deliverySchemeTable`
- `deliverySchemeIntervalTable`
- `networkDiscovery`
- `histogramCntTable`
- `histogramQueueTable`

14.5.3 Software Processes

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

- Two instances of the AMS process, the watchdog and the AMS process (`tp_ams`).
- One instance of the FXFER client (`tp_fclient`) ensuring continuous configuration updates
- One instance of the local trap daemon (`snmptrapd`) for logging SNMP traps locally

To verify that all processes on the AMS are running, execute the following command on the command line:

```
ps -ef | grep tp_ams
ps -ef | grep tp_fclient
ps -ef | grep snmptrapd
```

14.6 Command-Line Tools for Troubleshooting

Tool	Description	More Information
`tp_ams_db`	Rolls back the database after upgrading the AMS.	*tp_ams_db*
`tp_status`	Provides the operational state and uptime of ZephyrTel Mobile Messaging components.	Tools Operator Manual
`tp_system`	Allows you to: • View software and hardware information • Activate licenses • Boot the system • Enable and disable subscriptions to the trap service	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

14.6.1 tp_ams_db

To perform a rollback after upgrading the AMS, use the `tp_ams_db` command-line tool. This database conversion tool:

- Requests the configuration of the source database path, destination database path, and destination database version.
- Verifies that no overwriting of the original database takes place.
- Ensures that data is not permanently lost.

`tp_ams_db` is scriptable and exits with value 0 upon successful execution and with value 1 when an error is encountered.

Note: `tp_ams_db` should only be used when the AMS is inactive. In a rollback scenario, after the `tp_ams_db` tool is run, the AMS must be configured to use the converted database.

14.7 Commands for Troubleshooting

The following operating system tools are available for troubleshooting:

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

Tool	Description
`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/*.

Appendix A

AMS Counters Reference

Topics:

A.1 AMS Counters Reference

Counter Name	Indicates...
amsCntDeletedMaster	Total number of local AMS messages that were removed due to a permanent delivery error or through the OAM interface
amsCntDeletedReplica	Total number of replica messages on the local AMS that were removed due to a permanent delivery error or through the OAM interface
amsCntDeliveredMaster	Total number of successful deliveries for local AMS messages
amsCntDeliveredReplica	Total number of successful deliveries for the messages that the AMS is delivering as active replica
amsCntEventTriggeredGeneralSucessMaster	Total number of local AMS messages that were successfully delivered on delivery attempts triggered by the OMM interface
amsCntEventTriggeredGeneralSucessReplica	Total number of messages that the AMS is delivering as active replica that were successfully delivered on delivery attempts triggered by the OMM interface
amsCntEventTriggeredNetworkFailedMaster	Total number of failed attempts to deliver master messages on delivery attempts triggered by the reception of an alert
amsCntEventTriggeredNetworkFailedReplica	Total number of failed attempts to deliver messages that the AMS is delivering as active replica on delivery attempts triggered by the reception of an alert
amsCntEventTriggeredNetworkSuccessMaster	Total number of local AMS messages that were successfully delivered on delivery attempts triggered by the reception of an alert
amsCntEventTriggeredNetworkSuccessReplica	Total number of messages that the AMS is delivering as a replica that were successfully delivered on delivery attempts triggered by the reception of an alert
amsCntExpiredMaster	Total number of local AMS messages that were removed because they expired
amsCntExpiredReplica	Total number of messages for which the AMS is serving as active replica that were removed because they expired

Counter Name	Indicates...
amsCntFailedMaster	Total number of failed deliveries for local AMS messages
amsCntFailedReplica	Total number of failed deliveries for the messages that the AMS is delivering as active replica
amsCntInMemoryConversionLimitReached	Total amount of times a conversion could not be kept in memory because too many of them where pending.
amsCntInprogressReplication	Number of replicate requests currently in progress
amsCntPeakStoredMaster	Maximum number of stored messages for the local AMS device has been reached
amsCntPeakStoredReplica	Total number of replica messages stored in this AMS device for the other AMS devices
amsCntPendingDeliveries	Total number of pending delivery attempts for this AMS device
amsCntPendingInMemoryConversions	Total number of pending in-memory conversions in this AMS device.
amsCntRejectedDatabaseError	Indicates the total number of storage requests that failed because of one of the following errors: • Writing the message to the database failed • It was not possible to create the associated replica database
amsCntRejectedInvalidDeferred	Total number of storage requests that failed because the message deferred delivery time indicated a time interval before the first delivery attempt that is longer than defined in amsPropMaximumDeferredDelivery
amsCntRejectedInvalidMessage	Total number of storage requests that failed because they contained corrupted contents
amsCntRejectedInvalidValidity	Total number of storage requests that failed because the message parameters caused the message to expire before its first delivery attempt
amsCntRejectedMaxQueueSizeExceeded	Total number of storage requests that failed because the requested message queue was full
amsCntRejectedMaxRecipientMessagesExceeded	Total number of storage requests that failed because the maximum allowed number of messages for the recipient queue was reached
amsCntRejectedMaxTotalMessagesExceeded	Total number of storage requests that failed because the maximum allowed number of stored messages was reached

Counter Name	Indicates...
amsCntRejectedNotOperational	Total number of storage requests that failed because the AMS was not operational
amsCntRejectedNumberOfAttempts	Total number of storage requests that failed because of the number of previous delivery attempts (upon storage, the AMS detects that the current number of attempts exceeds the maximum number of attempts in the delivery scheme associated with the requested queue)
amsCntRejectedQueueNotActive	Total number of storage requests that failed because the requested message queue was not active
amsCntRejectedQueueNotExist	Total number of storage requests that failed because the requested message queue did not exist
amsCntRejectedRateExceeded	Total number of storage requests that failed because the ingress storage rate was exceeded. **Note:** The ingress rate is internally calculated by the AMS, it is not controlled through any configuration parameter.
amsCntReplacedMaster	Total number of replaced messages for the local AMS
amsCntReplacedReplica	Total number of replaced replicated messages; also incremented when replacement by duplicate replicate requests occurs. **Note:** Currently not used as the reason for replacing/updating a message in the replica store is not always known.
amsCntScheduledQueues	Total number of scheduled queues in this AMS
amsCntSearchList	Number of search list elements for replication
amsCntStoredActive	Number of stored messages that are currently being scheduled for delivery, including messages for the local AMS and messages for which this AMS is serving as active replica
amsCntStoredDeferredMaster	Total number of messages with a deferred delivery time not yet reached for the local AMS device
amsCntStoredMaster	Total number of messages stored for the local AMS device
amsCntStoredReplica	Total number of replica messages stored in this AMS device for the other AMS devices
amsCntSubmittedMaster	Total number of submitted messages

Counter Name	Indicates...
amsCntTotalMasterQueuesAllocated	Total number of queues allocated for the local AMS device
amsCntTotalQueuesAllocated	Total number of queues allocated in this AMS device
amsCntTotalReplicaQueuesAllocated	Total number of replica queues allocated in this AMS device for other AMS devices
amsCntTotalStored	Total number of messages stored in this AMS device
amsLoadCntDeletePeakRequests	Peak delete requests per second
amsLoadCntDeleteRequests	Number of delete requests of the previous second
amsLoadCntDeleteTotalRequests	Total number of delete requests received
amsLoadCntRetrievePeakRequests	Peak retrieve requests per second
amsLoadCntRetrieveRequests	Number of retrieve requests of the previous second
amsLoadCntRetrieveTotalRequests	Total number of retrieve requests received
amsLoadCntStoreNonVolatilePeakRequests	Peak store nonvolatile requests per second
amsLoadCntStoreNonVolatileRequests	Number of store nonvolatile requests of the previous second
amsLoadCntStoreNonVolatileTotalRequests	Total number of store nonvolatile requests received
amsLoadCntStoreVolatilePeakRequests	Peak store volatile requests per second
amsLoadCntStoreVolatileRequests	Number of store volatile requests of the previous second
amsLoadCntStoreVolatileTotalRequests	Total number of store volatile requests received
amsLoadCntUpdatePeakRequests	Peak update requests per second
amsLoadCntUpdateRequests	Number of update requests of the previous second
amsLoadCntUpdateTotalRequests	Total number of update requests received
histogramCntDelivered	Number of messages delivered in a particular number of attempts (for example, number of messages that were delivered in two attempts)
histogramCntQueues	Number of queues containing particular numbers of pending messages (for example, number of queues containing two pending messages)
histogramCntStorageDuration	Number of messages stored for a particular range of time (for example, number of messages stored for 1 to 2 seconds)

Counter Name	Indicates...
histogramQueueCntDelivered	Number of messages delivered in a particular number of attempts, per queue
histogramQueueCntStorageDuration	Number of messages stored for a particular range of time, per queue
amsCntRejectedIcacheDisabled	Total number of Icache requests rejected due to Icache support being disabled
amsCntStoredIcacheRecords	Number of Icache records stored in the AMS
amsCntReservedIcacheRecords	Number of Icache records reserved in the AMS
amsCntStoredMasterIcacheRecords	Number of Icache records stored in the AMS, where the allocation of message IDs occurred in this AMS
amsCntStoredReplicaIcacheRecords	Number of Icache records stored in the AMS, where the allocation of message IDs occurred in another AMS
queueCntTotalIcacheRecordsStored	Number of Icache records stored in the queue
queueCntReservedIcacheRecords	Number of Icache records reserved in the queue
queueCntMasterIcacheRecordsStored	Number of Icache records stored in the queue, where the allocation of message IDs occurred in this AMS
queueCntReplicaIcacheRecordsStored	Number of Icache records stored in the queue, where the allocation of message IDs occurred in another AMS

Appendix B

Query Command Line Interface

Topics:

B.1 Introduction

This section discusses the AMS Query Command Line Interface (Q-CLI). The Q-CLI can be used to manage messages and Intermediate Cache (Icache) records that are stored in the AMS. It is part of the AMS license.

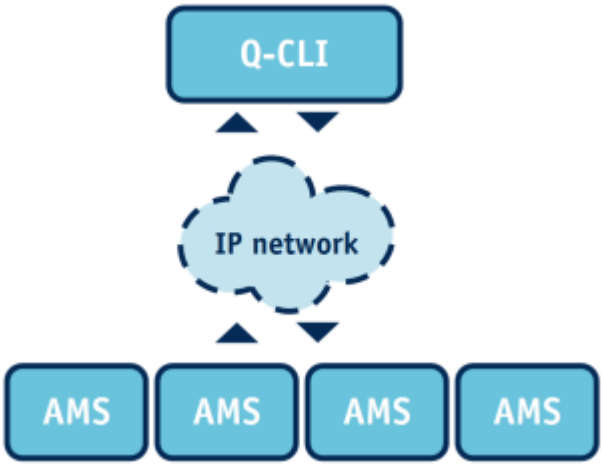


Figure 42: Q-CLI architecture

Q-CLI can be operated in the following modes:

- CLI mode

  CLI mode in the Q-CLI helps in managing the data in its local network. In the CLI mode Q-CLI takes the user command as input and executes that command on all the AMS nodes present in its own MXP network. In the CLI mode Q-CLI works as a single shot process, i.e. for every user command the tp_qcli gets invoked, processes the command and exits after the result (for both success and failure) is displayed on the console.

- Server mode

  Server mode in the Q-CLI is used by the client application which needs to manage the data remotely. The clients can be either the CCI or Q-CLI operating in a client mode. Q-CLI server accepts the TCP connection from the client. The default port on which Q-CLI server operates is 9600. On connecting with the client, Q-CLI server will process the command received from the client. Result of the processed command will be returned to the client. Command processed by the Q-CLI server is same as that in the CLI mode. When the client happens to be the CCI, the Q-CLI Server instance interfaces only with its own AMS node, not with any other AMS nodes in the local network.

- Client mode

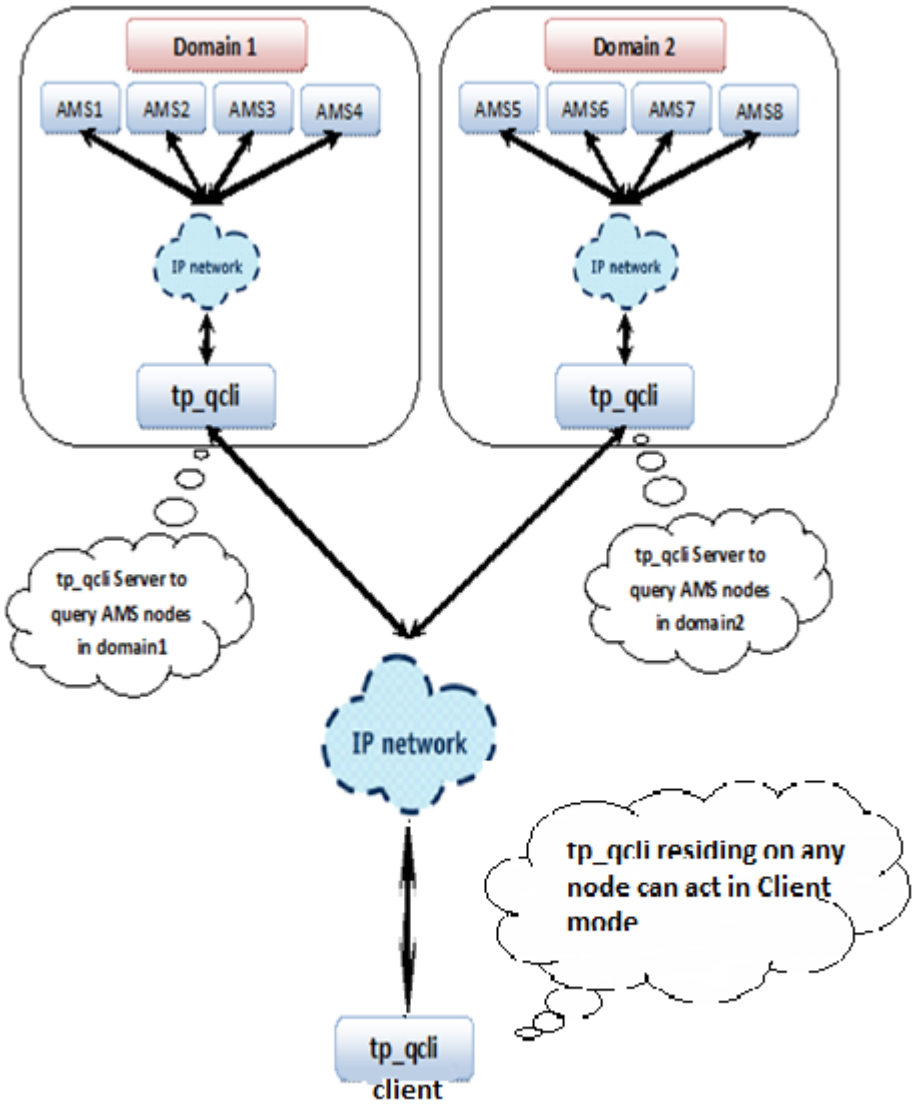


Figure 43: Q-CLI client-server architecture

Client mode in the Q-CLI is used to manage the data across different domains. Here a domain represents a single MXP network, such that all nodes belonging to the same domain share the same network address and can have direct TND/TNC communication with each other. Any mention of the term "domain" in the subsequent sections should be interpreted as referring to a MXP domain only, unless explicitly mentioned otherwise. This implies Q-CLI client connects to the Q-CLI servers residing on different domains. In Client mode, Q-CLI connects with one Q-CLI server per domain. Connection between Q-CLI client and server is a TCP based connection. Q-CLI client identifies which servers to be connected are based on the configuration file which can be provided as an input parameter. This configuration file is discussed later in this document. The behavior of the Q-CLI in the client mode will depend upon how the configuration file has been created.

Q-CLI supports the following data management functionality:

- `show`: Show information about specific messages or nodes
- `delete`: Delete a specific message
- `alert`: Alert a specific recipient

Q-CLI can run on any ZephyrTel Mobile Messaging device. It does not have to run on an AMS device. To facilitate scripting, it exits with value 0 upon successful execution and with value 1 when an error is encountered.

The `/usr/TextPass/etc/common_config.txt` file contains the configuration for all ZephyrTel Mobile Messaging devices. Q-CLI uses it to retrieve network discovery settings. These network discovery settings are used in the CLI and Server mode to identify its own local domain/network.

Q-CLI Log Messages

Q-CLI will generate two types of messages in the device's system log file (`/var/log/messages`):

- Messages from the `tp_qcli` program announcing that it can communicate with ZephyrTel Mobile Messaging network elements
- Messages from ZephyrTel Mobile Messaging processes that a link toward Q-CLI has been established or closed; these messages can be identified by the phrases:
  - `TND_NODE_JOIN (QCLI`
  - `TND_NODE_LEAVE (QCLI`
- In case of Client mode, log file will contain messages stating the connection status with the different configured domains.

B.2 Q-CLI Network Discovery Configuration

The Q-CLI gets network discovery parameters from the `common_config.txt` file. No additional parameters are needed.

The network discovery defaults are:

- Multicast address: 239.255.55.55
- Discovery address: 10.0.0.0
- Discovery mask: 255.255.255.0

You can change these default values using environment variables or command-line options.

The environment variables are:

- `MULTICAST_ADDR`
- `NETWORK_ADDRESS`
- `NETWORK_MASK`

Note: To view the current network discovery settings, use the Q-CLI `--help` command.

B.3 Q-CLI Client configuration

In Client mode, Q-CLI reads a configuration file to identify to which of the Q-CLI servers the client needs to create a connection with. This configuration file is provided as an input parameter to the Q-CLI client mode using the `--config_file` parameter (explained later in the document). If this configuration file is not provided as an input parameter, following default path of the configuration file will be used:

`~/tp_qcli_server.config`

This default configuration file is kept in the home directory of the concerned Unix account under which the client is being executed.

Note: There is no utility which aids in creation of the configuration file. Its upto to the operator to create the configuration as per the requirement and the way MXP-networks are being maintained in the network.

This configuration file contains entries in the following format:

`Domain_name, IP_address[:port]`

For example :

```
usa,10.0.1.1:9600
china,10.1.1.1
usa,10.0.1.2:9600
```

Some invalid configuration entries are mentioned below:

```
,10.0.1.1
usa,3131:9600
-usa,10.0.1.1:9600
Usa,10.1.1.1:,9600
```

Here `Domain_name` is a unique logical name given to each MXP-network for identification. Henceforth domain will be used for the MXP-network in the document.

IP address and port number is used to make a connection with the Q-CLI server.

Port number is an optional parameter. If this parameter is absent Q-CLI uses default port number 9600 to connect to the server.

This configuration file can contain multiple entries for different domains and for each domain multiple entries for different IP addresses. But the Q-CLI client will read and store information for a maximum of 5 IP address-ports per domain. Remaining information for the domain will not be used by the Q-CLI.

Q-CLI client will try to connect to only one Q-CLI server per domain. Client will try to connect to the first Q-CLI server configured in the configuration file. In case of a connection failure it will try connecting to the next configured server for that domain.

Sample Configuration File:

```
#domain,IP[:port]
Network1, 10.0.1.11
Network2,10.0.2.22:9700
Network3, 10.0.3.33:9600
```

B.4 Q-CLI Commands in the Client mode

B.4.1 Command-Line Usage

B.4.1.1 Synopsis

```
tp_qcli --client [--config_file] <configuration_file> [ tp_qcli_commands ] [
command_options ]
```

Command	Description
tp_qcli	Name of the program.
client	Specifies that tp_qcli will act in client mode.
config_file	Used in the client mode to specify the configuration file. It is an optional parameter.
[tp_qcli_commands]	A command that can be used with Q-CLI.
[command_options]	Options required by certain commands.

B.4.1.2 Command Description

Command	Description
`--show` or `-s`	Retrieves information about messages and nodes.
`--delete` or `-d`	Deletes a specific message.
`--alert` or `-a`	Alerts a specific recipient.
`--version` or `-v`	Provides the Q-CLI version number and the network discovery settings
`--help` or `-?`	Provides the Q-CLI's help information

B.4.1.3 Command Options

Option	show	delete	alert
`recipient=<address>`	Recipient address to search for	N/A	Address of recipient to alert
`r=<address>`			
`originator=<address>`	Originator address to search for	N/A	N/A
`o=<address>`			
`type=[mob|app|alpha|ip|icache]`	Type of address; defaults to international mobile number (`mob`)		
`t=[mob|app|alpha|ip|icache]`			

Option	show	delete	alert
`queue=<ID>` `q=<ID>`	Queue query (very resource consuming)	N/A	N/A
`nodes` `n`	Shows the connection status of the configured domains	N/A	N/A
`id=<AMS ID>:<message ID>` `i=<AMS ID>:<message ID>`	N/A	ID of message to delete. Along with this domain parameter is mandatory.	N/A
`multicast=<IP>` `c=<IP>`	Changes the default multicast address		
`network=<IP>` `n=<IP>`	Changes the default network address		
`mask=<IP>` `m=<IP>`	Changes the default network mask		
`port=<port>` `p=<port>`	N/A	N/A	N/A
`--verbose=[1|2|3|4]` `v=[1|2|3|4]`	Verbose mode: **1.** Short output (default) **2.** Extended output; shows recipient and originator from AMS fields [1] **3.** Extended output; shows recipient and originator from message fields [2] **4.** Extended output; shows detailed message fields along with the	N/A	N/A

[1] Messages with NPI=9 are marked by two asterisks (**). This only applies to `-v=2`.
[2] Shows the addresses as they entered the system, so not normalized or with a 'special' NPI value.

Option	show	delete	alert
	message content and message length.		
`start=<HHMMSSddmmyy>`	Optional start time	N/A	N/A
`s=<HHMMSSddmmyy>`			
`end=<HHMMSSddmmyy>`	Optional end time	N/A	N/A
`e=<HHMMSSddmmyy>`			
`passive`	Enables querying messages in the passive replica	N/A	N/A
`a`			
`domain`	Domain identifies the configured network from the configuration file in which messages will be searched for	Domain identifies the configured network from the configuration file in which message identified by `msg_id` to be deleted	Domain identifies the configured network from the configuration file in which message to be alerted
`d`			

B.4.1.4 Command Option Order

The order of the command options is important. The following options:

- `--recipient` or `-r`
- `--originator` or `-o`
- `--queue` or `-q`

Must be specified before the following options:

- `--type` or `-t`
- `--start` or `-s`
- `--end` or `-e`
- `--passive` or `-a`

B.4.1.5 Querying Passive Replicas

Use the `--passive` or `-a` option to view passive replica messages. When using this option, carefully analyze the output for the following parameters:

- `aa` indicates which AMS a message is in
- `s` indicates whether the AMS is a master (M) or replica (R)

To compare active replicas and passive replicas, execute Q-CLI with the `--passive` option and again without the `--passive` option, and compare the outputs.

Note: The retry time and the number of retries for passive replica messages do not show updated values.

B.4.2 Commands

B.4.2.1 show

Synopsis

```
tp_qcli --client [--config_file=<file_name>] --show --recipient=<address>
{--type=[mob|app|alpha|ip|icache]} \
{--verbose=[1|2|3|4]}
```

```
tp_qcli --client [--config_file=<file_name>] -s -r=<address>
{-t=[mob|app|alpha|ip|icache]} {-v=[1|2|3|4]}
```

```
tp_qcli --client [--config_file=<file_name>] --show --originator=<address>
{--type=[mob|app|alpha|ip|icache]} \
{--verbose=[1|2|3|4]}
```

```
tp_qcli --client [--config_file=<file_name>] -s -o=<address>
{-t=[mob|app|alpha|ip|icache]} {-v=[1|2|3|4]}
```

```
tp_qcli --client [--config_file=<file_name>] --show --nodes
```

```
tp_qcli --client [--config_file=<file_name>] -s -n
```

Description

The `show` command shall request all AMSs present in the connected domains for a specified recipient. Only the active messages are shown; replicated messages are not. Use the verbose level to change the output behavior. The mobile address must be in international format.

The `show` command can also be used to show the connection status of the configured domains.

Note: Originator queries are very I/O-intensive; use them with care.

Examples

The following examples show all stored messages for mobile recipient 31612345678:

```
tp_qcli --client -s -r=31612345678
```

```
tp_qcli --client --show --recipient=31612345678
```

```
tp_qcli --client -s -r=31612345678 -t=mob -v=1
```

In the above example, `tp_qcli` reads the default configuration file.

The following example shows the status of the configured domains specified in the user defined configuration file:

```
tp_qcli --client --config_file=/usr/TextPass/etc/ams_config.txt -s -n
```

The output of this command would be:

```
List of configured AMS domain
Domain_name             status
------------------------------------------------------------
usa                     Connected
```

B.4.2.2 delete

Synopsis

```
tp_qcli --client [--config_file=<file_name>] --delete --id=<message id>
--domain=<domain_name>
```

```
tp_qcli --client [--config_file=<file_name>] -d -i=<message id> -d=<domain_name>
```

Description

The `delete` command deletes a specific message stored in an AMS. Use the show command to retrieve the message ID. The `delete` command requires domain as an input parameter along with the message ID. Specified domain helps in identifying on which network this delete command is to be processed.

Examples

The following examples delete the message with ID 12345678:

```
tp_qcli --client --delete --id=12345678 --domain=usa
```

```
tp_qcli --client -d -i=12345678 -d=usa
```

The output of this command would be:

```
Message with id 12345678 deleted
```

In the above mentioned example, the message with message_id 12345678 will be deleted only from domain `usa`.

B.4.2.3 alert

Synopsis

```
tp_qcli --client [--config_file=<file_name>] --alert --recipient=<address>
{--type=[mob|app|alpha|ip]}
```

```
tp_qcli --client [--config_file=<file_name>] -a -r=<address> {-t=[mob|app|alpha|ip]}
```

Description

The `alert` command triggers a deliver attempt (unless already pending) for a specific recipient on the connected domains.

Note: The format of the alert command is the same for messages from applications with the **SMSC HLR Redirection Bypass** parameter set.

Examples

The following examples will trigger the mobile recipient with address 31612345678 to do a delivery attempt:

```
tp_qcli --client --alert --recipient=31612345678
```

```
tp_qcli --client -a -r=31612345678
```

The output of this command would be:

```
Recipient 31612345678 alerted
```

B.4.2.4 version

Synopsis

```
tp_qcli --client --version
```

```
tp_qcli --client -v
```

Description

The `version` command shows the version of the Q-CLI and the configuration of the network discovery.

Example

The following example retrieves the version and network discovery configuration:

```
tp_qcli --client --version
```

B.4.2.5 help

Synopsis

```
tp_qcli --help
```

```
tp_qcli -?
```

Description

The `help` command shows the Q-CLI's help information and the current network discovery default settings.

Examples

The following example shows the help information:

```
tp_qcli --help
```

The output of this command would be:

```
version: R01.11.04.00
 date   : 2018-11-28 09:19:48 QCLI_x86_64_LINUX dprabhu roanoke-vm1
 multicast address : 239.255.55.136
 network address   : 10.200.9.0
 network mask      : 255.255.255.0
```

B.5 Q-CLI Commands in the CLI and Server modes

Q-CLI commands have two parts:

- The command (for example, `--show`)
- The command options (for example, `--recipient`)

Every command returns a result code and a description to stderr (in case of an error). Returned information is printed to stdout. `tp_qcli` will return exit value 0 upon successful execution and exit value 1 when an error is encountered.

B.5.1 Command-Line Usage

B.5.1.1 Synopsis

```
tp_qcli [ tp_qcli_commands ] [ command_options ]
```

Command	Description
tp_qcli	Name of the program.
[tp_qcli_commands]	A command that can be used with Q-CLI.
[command_options]	Options required by certain commands.

B.5.1.2 Command Description

Command	Description
`--show` or `-s`	Retrieves information about messages and nodes.
`--delete` or `-d`	Deletes a specific message.
`--alert` or `-a`	Alerts a specific recipient.
`--cm_server` or `-c`	Acts as a Configuration Manager server.
`--version` or `-v`	Provides the Q-CLI version number and the network discovery settings
`--help` or `-?`	Provides the Q-CLI's help information

B.5.1.3 Command Options

Option	show	delete	alert	cm_server
`recipient=<address>`	Recipient address to search for	N/A	Address of recipient to alert	N/A
`r=<address>`				
`originator=<address>`	Originator address to search for	N/A	N/A	N/A
`o=<address>`				
`type=[mob|app|alpha|ip|icache]`	Type of address; defaults to international mobile number (`mob`)			
`t=[mob|app|alpha|ip|icache]`				
`queue=<ID>`	Queue query (very resource consuming)	N/A	N/A	N/A
`q=<ID>`				
`nodes`	Show the available AMS nodes	N/A	N/A	N/A
`n`				

Option	show	delete	alert	cm_server
`id=<AMS ID>:<message ID>`	N/A	ID of message to delete	N/A	N/A
`i=<AMS ID>:<message ID>`				
`multicast=<IP>`	Changes the default multicast address			
`c=<IP>`				
`network=<IP>`	Changes the default network address			
`n=<IP>`				
`mask=<IP>`	Changes the default network mask			
`m=<IP>`				
`port=<port>`	N/A	N/A	N/A	Optional port number (default 9600)
`p=<port>`				
`--verbose=[1|2|3|4]`	Verbose mode: **1.** Short output (default) **2.** Extended output; shows recipient and originator from AMS fields [3] **3.** Extended output; shows recipient and originator from message fields [4] **4.** Extended output; shows detailed message fields along with the	N/A	N/A	N/A
`v=[1|2|3|4]`				

[3] Messages with NPI=9 are marked by two asterisks (**). This only applies to `-v=2`.
[4] Shows the addresses as they entered the system, so not normalized or with a 'special' NPI value.

Option	show	delete	alert	cm_server
	message content.			
`start=<HHMMSSddmmyy>`	Optional start time	N/A	N/A	N/A
`s=<HHMMSSddmmyy>`				
`end=<HHMMSSddmmyy>`	Optional end time	N/A	N/A	N/A
`e=<HHMMSSddmmyy>`				
`passive`	Enables querying messages in the passive replica	N/A	N/A	N/A
`a`				

B.5.1.4 Command Option Order

The order of the command options is important. The following options:

- `--recipient` or `-r`
- `--originator` or `-o`
- `--queue` or `-q`

Must be specified before the following options:

- `--type` or `-t`
- `--start` or `-s`
- `--end` or `-e`
- `--passive` or `-a`

B.5.1.5 Querying Passive Replicas

Use the `--passive` or `-a` option to view passive replica messages. When using this option, carefully analyze the output for the following parameters:

- `aa` indicates which AMS a message is in
- `s` indicates whether the AMS is a master (M) or replica (R)

To compare active replicas and passive replicas, execute Q-CLI with the `--passive` option and again without the `--passive` option, and compare the outputs.

Note: The retry time and the number of retries for passive replica messages do not show updated values.

B.5.2 Commands

B.5.2.1 show

Synopsis

```
tp_qcli --show --recipient=<address> {--type=[mob|app|alpha|ip|icache]}
{--verbose=[1|2|3|4]}
```

```
tp_qcli -s -r=<address> {-t=[mob|app|alpha|ip|icache]} {-v=[1|2|3|4]}
```

```
tp_qcli --show --originator=<address> {--type=[mob|app|alpha|ip|icache]}
{--verbose=[1|2|3|4]}
```

```
tp_qcli -s -o=<address> {-t=[mob|app|alpha|ip|icache]} {-v=[1|2|3|4]}
```

```
tp_qcli --show --nodes
```

```
tp_qcli -s -n
```

Description

The `show` command requests all AMSs for a specified recipient. Only the active messages are shown; replicated messages are not. Use the verbose level to change the output behavior. The mobile address must be in international format.

The `show` command can also be used to show all available AMS nodes.

Note: Originator queries are very I/O-intensive; use them with care.

Examples

The following examples show all stored messages for mobile recipient 31612345678:

```
tp_qcli -s -r=31612345678
```

```
tp_qcli --show --recipient=31612345678
```

```
tp_qcli -s -r=31612345678 -t=mob -v=1
```

The following example shows all available AMS nodes:

```
tp_qcli -s -n
```

The output of this command would be:

```
AMS    status       host_id     sub_id      port     addresses
-------------------------------------------------------------
0      available    832e13b9    00005a08    11168    10.0.0.70
```

B.5.2.2 delete

Synopsis

```
tp_qcli --delete --id=<ams id:message id>
```

```
tp_qcli -d -i=<ams id:message id>
```

Description

The `delete` command deletes a specific message stored in an AMS. Use the show command to retrieve the message ID.

Examples

The following examples delete the message belonging to AMS 1 (not necessarily handled by AMS 1) with message ID 12345678:

```
tp_qcli --delete --id=1:12345678
```

```
tp_qcli -d -i=1:12345678
```

The output of this command would be:

```
Message with id 12345678 deleted
```

The following example deletes the Icache record with message ID 0000000003:

```
tp_qcli -d -i=0000000003
```

The output of this command would be:

```
Message with id 0000000003 deleted
```

B.5.2.3 alert

Synopsis

```
tp_qcli --alert --recipient=<address> {--type=[mob|app|alpha|ip]}
```

```
tp_qcli -a -r=<address> {-t=[mob|app|alpha|ip]}
```

Description

The `alert` command triggers a deliver attempt (unless already pending) for a specific recipient.

Note: The format of the alert command is the same for messages from applications with the **SMSC HLR Redirection Bypass** parameter set.

Examples

The following examples will trigger the mobile recipient with address 31612345678 to do a delivery attempt:

```
tp_qcli --alert --recipient=31612345678
```

```
tp_qcli -a -r=31612345678
```

The output of this command would be:

```
Recipient 31612345678 alerted
```

B.5.2.4 version

Synopsis

```
tp_qcli --version
```

```
tp_qcli -v
```

Description

The `version` command shows the version of the Q-CLI and the configuration of the network discovery.

Example

The following example retrieves the version and network discovery configuration:

```
tp_qcli -version
```

B.5.2.5 help

Synopsis

```
tp_qcli --help
```

```
tp_qcli -?
```

Description

The `help` command shows the Q-CLI's help information and the current network discovery default settings.

Examples

The following example shows the help information:

```
tp_qcli --help
```

The output of this command would be:

```
version: R01.11.04.00
 date   : 2018-11-28 09:19:48 QCLI_x86_64_LINUX dprabhu roanoke-vm1
 multicast address : 239.255.55.136
 network address   : 10.200.9.0
 network mask      : 255.255.255.0
```

B.6 Q-CLI Output

B.6.1 Q-CLI Record Example

Sample command:

```
$ tp_qcli -s -r 249922413868 -v=2
```

Sample output:

```
msg_id     t aa s originator       recipient       sc.timestamp next retry   nr prot
 oc dcs pid defdel.time  val.period   ds errc submit.time                    nt
notif. addr    npid pri rpi billing_id sc.adress
authen.code    lastresort addr  lpid ss orig.imsi    orig.msc addr.
-------------------------------------------------------------------------------
000000163A M  0 M   +249921950005    +249922413868 131513010207 133020010207  1
      00  00              171513050207    0001 131513010207                 df
                          1               +249921999995
                                                  +249921999930
```

B.6.2 Q-CLI Icache Record Example

Sample command:

```
$ tp_qcli -s -q=1 -t=icache
```

Sample output:

```
msg_id     t aa s originator recipient          sc.timestamp         next retry
nr
-------------------------------------------------------------------------------
0000000003 I  0 M xe1 4016   unk 00495800020000 135603010210           -
 0
0000000004 I  0 M xe1 4016   unk 00495800020000 144718010210           -
 0
0000000005 I  0 M xe1 4016   unk 00495800020000 144912010210           -
 0
```

B.6.3 Output Parameters

Parameter	Description
msg_id	Identifier of the message. A unique number is assigned to each message.
t	Type of message: • M: Message • S: Status report
aa	ID of the AMS handling the message, indicating in which AMS node the message is stored.
s	Type of queue the message is stored in: • M: Master • R: Replica
originator	Originator address: • MSISDN in international format • Application short code
recipient	Recipient address: • MSISDN in international format • Application short code If the message originated from an application with the **SMSC HLR Redirection Bypass** parameter set, two asterisks (`**`) will precede the recipient address when Q-CLI is run with the `-v=2` option.
sc.timestamp	Service center timestamp in hh:mm:ss:dd:mm:yy format.
next retry	Time and date for which the next retry is scheduled, in hh:mm:ss:dd:mm:yy format.
nr	Number of retries already attempted for the message.

Parameter	Description
prot	The type of protocol over which an AT message is to be sent: • UCP • SMPP • CIMD
oc	Operation code of the AO message (only applies to AT messages).
dcs	Data coding scheme (0-255); refer to GSM03.38.
pid	PID value of the message (0-255); refer to GSM03.40.
defdel.time	Deferred delivery time in hh:mm:ss:dd:mm:yy format.
val.period	Validity period in hh:mm:ss:dd:mm:yy.
ds	AMS internal delivery status; the value in hexadecimal indicating the internal message status: • 0: No status • 1: In progress • 2: Validity expired • 3: Delivery failed • 4: Delivery successful • 5: No response • 6: Last no response • 7: Cancelled (CIMD specific) • 8: Deleted • 9: Deleted by cancellation • A: Scheduled • B: Accepted • C: Rejected • D: Skipped • E: Replaced **Note:** This field is not yet implemented.
errc	Error code returned from the mobile network on delivery attempt; refer to *Q-CLI Error Codes* for possible error codes.
submit.time	Submit time in hh:mm:ss:dd:mm:yy format.
nt	Value (0-7) indicating if an application requested a delivery notification; derived from a combination of notification request options: • 0: Delivery notification • 1: Non-delivery notification • 2: Buffered notification
notif.addr	Notification address, representing an application address as a short number .

Parameter	Description
npid	Notification PID (UCP only).
pri	Priority value as it was included in the AO message; note the priority values are different for UCP, SMPP, and CIMD.

rpi	Reply-path indicator of the incoming message:
	• UCP:
	  • 0: None
	  • 1: Request
	  • 2: Response
	• Non-UCP:
	  • t: True
	  • f: False

billing_id	Specific field in case of a UCP 51 message.
sc.adress	Service center address in international format.
authen.code	The AC field in case of a UCP 51 message.
lastresortaddr	Specific field in case of a UCP 51 message.
lpid	Specific field in case of a UCP 51 message.
ss	Single-shot indicator of the AO message: • t: True • f: False
orig.imsi	Originating IMSI.
orig.mscaddr	Originating MSC address in international format.
message_length	Length of the message content stored in the AMS.
domain	Domain name from which the message was retrieved.
message_contents	Message content of the stored message. It displays a maximum of 160 bytes. (To view the message content the Translation setting of the putty or the corresponding SSH client needs to be set accordingly).

Note: Certain fields will only contain a value for certain message types (for example, in case of a message received over SS7, the UCP 51, parameter values will be empty).

B.7 Q-CLI Error Codes

B.7.1 MT Errors

```
00000001 hex   MT TIMEOUT
00000002 hex   MT ABSENT SUBSCRIBER
00000003 hex   MT SYSTEM FAILURE
00000004 hex   MT DATA MISSING
00000005 hex   MT UNEXPECTED DATA VALUE
00000006 hex   MT FACILTY NOT SUPPORTED
00000007 hex   MT UNIDENTIFIED SUBSCRIBER
00000008 hex   MT ILLEGAL SUBSCRIBER
00000009 hex   MT ILLEGAL EQUIPMENT
0000000a hex   MT SUBSCRIBER BUSY FOR MT SM
0000000b hex   MT INVALID SME ADDRESS
0000000c hex   MT EQUIPMENT PROTOCOL ERROR
0000000d hex   MT EQUIPMENT NOT SM EQUIPPED
0000000e hex   MT DESTINATION PHONE OUT OF MEMORY
0000000f hex   MT OTHER MAP ERROR
00000010 hex   MT TCAP ABORTED
00000011 hex   MT SCCP ABORTED
00000012 hex   MT NO PAGING RESPONSE
00000013 hex   MT IMSI DETACHED ERROR
00000014 hex   MT ROAMING RESTRICTION ERROR
00000015 hex   MT SHORT MSG TYPE0 NOT SUPPORTED ERROR
00000016 hex   MT CAN NOT REPLACE SHORT MSG
00000017 hex   MT UNSPECIFIED TP-PID ERROR
00000018 hex   MT MESSAGE CLASS NOT SUPPORTED
00000019 hex   MT UNSPECIFIED TP-DCS ERROR
0000001a hex   MT TPDU NOT SUPPORTED ERROR
0000001b hex   MT (U)SIM SMS STORAGE FULL ERROR
0000001c hex   MT NO SMS STORAGE CAPABILITY IN (U)SIM ERROR
0000001d hex   MT ERROR IN MS
0000001e hex   MT (U)SIM APPLICATION TOOLKIT BUSY ERROR
0000001f hex   MT (U)SIM DATA DOWNLOAD ERROR
00000020 hex   MT APPLICATION SPECIFIC ERROR
00000021 hex   MT UNSPECIFIED ERROR CAUSE
00000022 hex   MT UE DEREGISTERED
00000023 hex   MT NO RESPONSE VIA IPSMGW
0000ffff hex   MT BLOCKED BY MT RULE
```

B.7.2 SRI-SM Errors

```
01000001 hex   SRISM TIMEOUT
01000002 hex   SRISM SYSTEM FAILURE
01000003 hex   SRISM DATA MISSING
01000004 hex   SRISM UNEXPECTED DATA VALUE
01000005 hex   SRISM FACILITY NOT SUPPORTED
01000006 hex   SRISM UNKNOWN SUBSCRIBER
01000007 hex   SRISM ABSENT SUBSCRIBER
01000008 hex   SRISM CALL BARRED
01000009 hex   SRISM TELESERVICE NOT PROVISIONED
0100000a hex   SRISM OTHER MAP ERROR
0100000b hex   SRISM TCAP ABORTED
0100000c hex   SRISM SCCP ABORTED
0100000d hex   SRISM MS DEREGISTERED ERROR
0100000e hex   SRISM MS PURGED ERROR
```

Note: The 'MT ABSENT SUBSCRIBER' and 'SRISM ABSENT SUBSCRIBER' errors have been deprecated. RTR will not be sending these two error codes to AMS anymore. Instead, it will send either 'MT NO PAGING ERROR', 'MT IMSI DETACHED ERROR' or 'MT ROAMING RESTRICTION ERROR' in case it is a MT-FSM error; if it is a SRI-SM error it will send either 'SRISM MS DEREGISTERED ERROR' or 'SRISM MS PURGED ERROR'.

B.7.3 AT Errors

```
02000001 hex   AT SYSTEM ERROR
02000002 hex   AT SHUTTING DOWN
02000003 hex   AT MXP FAILURE
02000004 hex   AT MXP TIMEOUT
02000005 hex   AT TX FAILURE
02000006 hex   AT TEMPORARY ERROR
02000007 hex   AT PERMANENT DEST ERROR
02000008 hex   AT PERMANENT MSG ERROR
02000009 hex   AT DESTINATION NOT AVAILABLE
0200000a hex   AT SOURCE NOT AVAILABLE
0200000b hex   AT THROUGHPUT EXCEEDED
02000011 hex   AT OPERATION NOT ALLOWED
02000012 hex   AT OPERATION NOT SUPPORTED
02000013 hex   AT RESPONSE TIMEOUT
02000014 hex   AT INVALID SYNTAX
02000015 hex   AT INVALID CHECKSUM
02000016 hex   AT ADDRESS BLOCKED
02000017 hex   AT RECIPIENT ERROR
02000018 hex   AT INVALID RECIPIENT
02000019 hex   AT INVALID MSG TYPE
0200001a hex   AT INVALID MSG
0200001b hex   AT INVALID TIME PERIOD
0200001c hex   AT INVALID ADDRESS
0200001d hex   AT MSG NOT FOUND
0200001e hex   AT DELIVERY IN PROGRESS
0200001f hex   AT TRANSPARENT
02000021 hex   AT WINDOW FULL
02000022 hex   AT MAX MESSAGES BUFFERED
02000023 hex   AT INVALID BIND STATUS
02000024 hex   AT NO ROUTING RULE
```

B.7.4 AMS Errors

```
03000001 hex   AMS DEVICE NOT ACTIVE
03000002 hex   AMS DB ERROR
03000003 hex   AMS STORE FULL
03000004 hex   AMS QUEUE FULL
03000005 hex   AMS RECIP BUFFER FULL
03000006 hex   AMS INVALID QUEUE
03000007 hex   AMS INVALID MESSAGE
03000008 hex   AMS INVALID VALIDITY TIME
03000009 hex   AMS INVALID DEFERRED DELIVERY
0300000a hex   AMS STORAGE RATE EXCEEDED
0300000b hex   AMS DELIVERY ATTEMPTS EXCEEDED
```

B.7.5 RTR Errors

```
04000000 hex   RTR BLOCKED BY THROUGHPUT CONTROL
04000001 hex   RTR STORAGE FAILURE
04000002 hex   RTR TIMEOUT
04000003 hex   RTR BLOCKED BY RULE
```

```
04000004 hex   RTR NO RULE MATCHING
04000005 hex   RTR LICENSE EXCEEDED
04000006 hex   ORIGINATOR IN BLACK LIST
04000007 hex   ORIGINATOR NOT IN WHITE LIST
04000008 hex   RTR MESSAGE TOO LONG
```

B.7.6 MO Errors

```
05000003 hex   MO TIMEOUT
05000004 hex   MO DROPPED
05000005 hex   MO DISCARDED
05000006 hex   MO TPR REJECTED
05000009 hex   MO SYSTEM FAILURE
0500000a hex   MO DATA MISSING
0500000b hex   MO UNEXPECTED DATA VALUE
0500000c hex   MO FACILTY NOT SUPPORTED
0500000d hex   MO UNKNOWN SERVICE CENTRE
0500000e hex   MO SERVICE CENTRE CONGESTION
0500000f hex   MO INVALID SME ADDRESS
05000010 hex   MO SUBSCRIBER NOT SC SUBSCRIBER
05000011 hex   MO OTHER MAP ERROR
05000012 hex   MO TCAP ABORTED
05000013 hex   MO SCCP ABORTED
```

Appendix C

Sample Configuration File

Topics:

C.1 Sample Common Configuration File

```
<tpconfig
    networkdiscoverymulticastaddress="239.255.55.51"
    networkdiscoverynetworkaddress="172.30.235.0"
    networkdiscoverynetworkmask="255.255.255.0"

    amsnumberofreplicas="1"
    amsmasterstoragetype="nonvolatile"
    amsreplicastoragetype="volatile"

    amsdbdatabasedirectorymaster="/dbamsstore/master"
    amsdbtransactiondirectorymaster="/dbamslog/master"
    amsdbdatabasedirectoryreplica="/dbamsstore/replica"
    amsdbtransactiondirectoryreplica="/dbamslog/replica"
  >

</tpconfig>
```

C.2 Sample Host-Specific Configuration File

```
<tpconfig
    ipaddress="127.0.0.1"
    runtextpassprocess="false"
    runtpfclientprocess="true"
    runtextamsprocess="true"
    runqclidprocess="true"

    processpriority="critical"
    >


    <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>
```

C.3 Sample Common Configuration File in Multi-Instance Setup

```
<tpconfig
 networkdiscoverymulticastaddress="239.255.55.51"
 networkdiscoverynetworkaddress="172.30.235.0"
 networkdiscoverynetworkmask="255.255.255.0"
 amsnumberofreplicas="1"
```

```
 amsmasterstoragetype="nonvolatile"
 amsreplicastoragetype="volatile"
    >
</tpconfig>
```

C.4 Sample Host-Specific Configuration File in Multi-Instance Setup

```
<tpconfig
 ipaddress="127.0.0.1"
 runtextpassprocess="false"
 runtpfclientprocess="true"
 runtextamsprocess="true"
 runqclidprocess="true"
 processpriority="critical"
 amsdbdatabasedirectorymaster="/dbamsstore/master"
 amsdbtransactiondirectorymaster="/dbamslog/master"
 amsdbdatabasedirectoryreplica="/dbamsstore/replica"
 amsdbtransactiondirectoryreplica="/dbamslog/replica"
 >
 <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. ZephyrTel Mobile Messaging RTR Operator Manual
2. ZephyrTel Mobile Messaging HUB Operator Manual
3. ZephyrTel Mobile Messaging Tools Operator Manual
4. ZephyrTel Mobile Messaging MGR Operator Manual

Glossary

A

ACK	Data Acknowledgement
AMS	Active Message Store Provides store-and-forward functionality for SMS messages.
AO	Application Originated Short message traffic that is originated by an application.
ASN.1	Abstract Syntax Notation One
AT	Application Terminated Short message traffic that terminates at an application.

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

C

	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.

D

daemon	A process that runs in the background (rather than under the direct control of a user) and performs a specified operation at predefined times or in response to certain events. Generally speaking, daemons are assigned names that end with the letter "d." For example, sentryd is the daemon that runs the Sentry utility.
Database	All data that can be administered by the user, including cards, destination point codes, gateway screening tables, global title translation tables, links, LNP services, LNP service providers, location routing numbers, routes, shelves, subsystem applications, and 10 digit telephone numbers.
disk	A single disk drive residing in a Controller Enclosure or a Disk Enclosure. A Disk can be assigned to a Disk Group, designated as a Spare or Global Spare, or left unused.

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

F

	failover may not be desirable, and human intervention may be required to initiate the failover manually.
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 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.
FWL	Firewall Helps protect subscribers from receiving unwanted messages and provides statistical information and message details about inbound suspect messages.
FXFER	ZephyrTel proprietary file transfer solution for the Mobile Messaging network. It uses a server process (tp_fserver) and client processes (tp_fclient).

G

GB	Gigabyte — 1,073,741,824 bytes
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

G

	than being limited to character based commands.

H

HDD	Hard Disk Drive
HLR	Home Location Register
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
IN	Intelligent Network A network design that provides an open platform for developing, providing and managing services.
I/O	Input/Output
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

I

	re-assembly through the data link layer.

L

LGV	Log Viewer Logs information about ZephyrTel Mobile Messaging operations and displays it in the Manager.

M

MAP	Mobile Application Part
MGR	A Web-based interface for managing ZephyrTel Mobile Messaging components. Prior to Suite 6, the Configuration Manager (CM) provided this functionality.
MIB	Management Information Database
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
MT	Mobile Terminated All transmissions that reach the mobile station and are accepted by it, such as calls or short messages.

M

MXP	Message eXchange Protocol ZephyrTel proprietary protocol used for communication between the Mobile Messaging HUB, RTR, and AMS components.

O

OAM	Operations, Administration, and Maintenance The application that operates the Maintenance and Administration Subsystem which controls the operation of many ZephyrTel products.

P

PBC	Prepaid Billing Controller Performs prepaid charging using the Diameter, CAMEL, or SMPP+ interface.
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 packet loss (if any) rate between hosts.
PLMN	Public Land Mobile Network

Q

Q-CLI	AMS Query Command Line Interface Enables operators to query messages that are stored in the AMS.

R

RAID	Redundant Array of Independent Disks A group of disks presented to clients as one or more large virtual disks, with accesses coordinated among multiple disks concurrently to increase performance, reliability, or both.
RAM	Random Access Memory A type of computer memory that can be accessed randomly; that is, any byte of memory can be accessed without touching the preceding bytes.
RTR	Router Routes all types of SMS traffic.

S

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.

S

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 original purpose wherever reliable datagram service is desired.
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
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

S

	called managed objects and are stored in a management information base (MIB). The SNMP protocol arranges managed objects into groups.
SRI	Send_Route_Information Message
SS7	Signaling System #7
SSL	Secure Socket Layer
STV	Statistics Viewer Collects statistical data about ZephyrTel Mobile Messaging components and displays it in the Manager.

T

TCP	Transfer Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security A cryptographic protocol that provides security for communications over networks such as the Internet. TLS encrypts the segments of network connections at the transport layer end-to-end.
Tools	A collection of command-line tools for managing and troubleshooting ZephyrTel Mobile Messaging components.

T

trap	A mechanism used in the context of SNMP (Simple Network Management Protocol) for one-way event notification.

U

UCP	Universal Computer Protocol Protocol used to connect to SMSCs.
UDH	User Data Header
UDP	User Datagram Protocol

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.