
Confidential Under Non-disclosure Agreement

EXHIBIT C

**NPAC SMS
INTEROPERABLE INTERFACE SPECIFICATION**

NANC Version 3.4.6b

Prepared for:
The North American Numbering Council (NANC)

February 14, 2014

Release 3.4: © 1997 - 2014 NeuStar, Inc.

The Work is subject to the terms of the GNU General Public License (the "GPL"), a copy of which may be found at <ftp://prep.ai.mit.edu/pub/gnu/GPL>. Any use of this Work is subject to the terms of the GPL. The "Work" covered by the GPL by operation of this notice and license is this document and any and all modifications to or derivatives of this document. Where the words "Program," "software," "source code," "code," or "files" are used in the GPL, users understand and agree that the "Work" as defined here is substituted for purposes of this notice and license.

CONFIDENTIAL UNDER NON-DISCLOSURE AGREEMENT

This page intentionally left blank.

Table Of Contents

1	Introduction	1
1.1	Document Overview	1
1.2	How To Use This Document	1
1.3	Document Numbering Strategy	1
1.4	Document Version History	2
1.4.1	Release 1.0	2
1.4.2	Release 2.0	2
1.4.3	Release 3.0	2
1.4.4	Release 3.1	3
1.4.5	Release 3.2	3
1.4.6	Release 3.3	3
1.4.7	Release 3.3.4	3
1.4.8	Release 3.4	3
1.5	References	4
1.5.1	Standards	4
1.5.2	Related Publications	5
1.6	Abbreviations/Definitions	6
2	Interface Overview	9
2.1	Overview	9
2.2	OSI Protocol Support	9
2.3	SOA to NPAC SMS Interface	10
2.3.1	Subscription Administration	11
2.3.2	Audit Requests	11
2.3.3	Notifications	11
2.3.4	Service Provider Data Administration	12
2.3.5	Network Data Download	13
2.3.6	Number Pool Block Administration	13
2.3.7	SPID Migration	13
2.4	NPAC SMS to Local SMS Interface	13
2.4.1	Subscription Version, Number Pool Block and Network Data Download	14
2.4.2	Service Provider Data Administration	14
2.4.3	Notifications	15
2.4.4	SPID Migration	15
2.5	NPAC and SOA/LSMS Interface Performance	15
3	Hierarchy Diagrams	17
3.1	Overview	17
3.1.1	Managed Object Model Inheritance Hierarchy	17
3.1.2	Log Record Managed Object Hierarchy	20
3.1.3	NPAC SMS to Local SMS Naming Hierarchy for the NPAC SMS	21
3.1.4	NPAC SMS to Local SMS Naming Hierarchy for the Local SMS	22
3.1.5	SOA to NPAC SMS Naming Hierarchy for the NPAC SMS	23
3.1.6	NPAC SMS to SOA Naming Hierarchy for the SOA	24

4	Interface Functionality to CMIP Definition Mapping	25
4.1	Overview	25
4.1.1	Primary NPAC Mechanized Interface Operations	25
4.1.2	Managed Object Interface Functionality.....	29
4.1.3	Action Interface Functionality	34
4.1.4	Notification Interface Functionality.....	35
4.2	Scoping and Filtering Support.....	39
4.2.1	Scoping	39
4.2.2	Filtering	39
4.2.3	Action Scoping and Filtering Support	41
4.3	lnpLocal-SMS-Name and lnpNPAC-SMS-Name Values	41
4.4	OID Usage Information	42
4.4.1	OIDs Used for Bind Requests.....	42
4.4.2	Other OIDs of Interest	42
4.5	Naming Attributes.....	42
4.6	Subscription Version M_DELETE Messages.....	42
4.7	Number Pool Block M_DELETE Messages	42
4.8	Subscription Version Queries	43
4.9	NPAC Rules for Handling of Optional Data Fields:	44
5	Secure Association Establishment.....	47
5.1	Overview	47
5.2	Security	47
5.2.1	Authentication and Access Control Information.....	47
5.2.1.1	System Id	49
5.2.1.2	System Type	49
5.2.1.3	User Id	49
5.2.1.4	List Id.....	49
5.2.1.5	Key Id	50
5.2.1.6	CMIP Departure Time	51
5.2.1.7	Sequence Number.....	51
5.2.1.8	Association Functions.....	51
5.2.1.9	Recovery Mode.....	53
5.2.1.10	Signature	53
5.2.2	Association Establishment.....	53
5.2.3	Data Origination Authentication.....	55
5.2.4	Audit Trail	57
5.3	Association Management and Recovery	57
5.3.1	Establishing Associations	57
5.3.1.1	NpacAssociationUserInfo.....	57
5.3.1.2	Unbind Requests and Responses	58
5.3.1.3	Aborts	58
5.3.1.4	NPAC SMS Failover Behavior.....	59
5.3.1.5	Service Provider SOA and Local SMS Procedures	59
5.3.2	Releasing or Aborting Associations.....	60
5.3.3	Error Handling.....	60

Table of Contents

5.3.3.1	NPAC SMS Error Handling	60
5.3.3.2	Processing Failure Error	61
5.3.3.3	NPAC SMS Detailed Error Codes	62
5.3.4	Recovery	62
5.3.4.1	Local SMS Recovery	67
5.3.4.2	SOA Recovery	67
5.3.4.3	Linked Action Replies during Recovery	67
5.4	Congestion Handling.....	69
5.4.1	NPAC SMS Congestion.....	69
5.4.2	NPAC Handling of Local SMS and SOA Congestion.....	69
5.4.3	Out-Bound Flow Control.....	70
5.5	Abort Processing Behavior.....	70
5.6	Single Association for SOA/LSMS	71
5.7	Separate SOA Channel for Notifications	72
6	<i>GDMO Definitions</i>	73
7	<i>General ASN.1 Definitions</i>	75
8	<i>LNP XML Schema</i>	76
9	<i>Subscription Version Status.....</i>	77
10	<i>Number Pool Block Status.....</i>	83

1 Introduction



1.1 Document Overview

The NPAC SMS Interoperable Interface Specification contains the information model for the Number Portability Administration Center and Service Management System (NPAC SMS) mechanized CMIP interfaces. Both Service Order Activation (SOA) and Local Service Management System (LSMS or Local SMS) interfaces to the NPAC SMS are described in this document.

1.2 How To Use This Document

The NPAC SMS Interoperable Interface Specification contains the following sections:

Section 1 *Introduction* -- This section describes the conventions and organization of this document. It also lists related documentation.

Section 2 *Interface Overview* -- This section contains an overview of CMIP protocol requirements and a brief description of the functionality provided in each interface.

Section 3 *Hierarchy Diagrams* -- This section contains the class hierarchy diagrams for all managed objects defined in the CMIP interoperable interface.

Section 4 *Interface Functionality to CMIP Definition Mapping* -- This section contains the mapping of the CMIP interface functionality to the managed objects, attributes, actions, and notifications.

Section 5 *Secure Association Establishment*-- This section contains information on secure association establishment.

Section 6 *GDMO Definitions* -- This section contains the GDMO interface definitions supporting the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface over CMIP.

Section 7 *General ASN.1 Definitions* -- This section contains the ASN.1 definitions that support the GDMO definitions in Section 7.

Section 8 *Subscription Version Status* -- This section contains a Subscription Version Status diagram, which illustrates the transition from one subscription version state to another.

1.3 Document Numbering Strategy

Starting with Release 2.0 the documentation number of the IIS document will be Version X.Y.Z as follows:

X – will only be incremented when a new major release of the NPAC SMS system is authorized. It will contain only the Change Orders that have been authorized for inclusion in this new major release.

Y – will only be incremented when a new sub-release of an existing release X is authorized. It will contain only the Change Orders that have been authorized for inclusion in this new

sub-release.

Z – will be incremented when documentation only clarifications and/or backward compatibility issues or other deficiency corrections are made in the IIS and/or FRS. This number will be reset to 0 when Y is incremented.

For example, the first release of the Release 2 IIS will be numbered 2.0.0. If documentation only clarifications are introduced in the next release of the IIS document it will be numbered 2.0.1. If requirements are added to Release 2.0 that require NPAC SMS software changes then the next release of the IIS document will be numbered 2.1.0.

Starting with Release 3.2, the documentation number of the FRS document will include a "lowercase letter" following the Z designation. This "lowercase letter" will essentially serve as a version indicator for the release of the documentation, such that the X.Y.Za will be a unique identifier. It will be used for both drafts and final versions. For example, the first release using this new convention will be 3.2.0a, followed by 3.2.0b, and so on. . The "lower case letter" shall be reset to 'a' when Z is incremented.

This number scheme is intended to make the mapping between NPAC SMS and the FRS and IIS documentation consistent.

1.4 Document Version History

1.4.1 Release 1.0

NANC Version 1.0, released on 04/07/97, contains changes from the ICC Subcommittee IIS Version 1.1.5.

NANC Version 1.1, released on 05/08/97, contains changes from the NANC IIS Version 1.0.

NANC Version 1.2, released on 05/25/97, contains changes from the NANC IIS Version 1.1.

NANC Version 1.3, released on 07/09/97, contains changes from the NANC IIS Version 1.2.

NANC Version 1.4, released on 08/08/97, contains changes from the NANC IIS Version 1.3.

NANC Version 1.5, released on 09/09/97, contains changes from the NANC IIS Version 1.4.

NANC Version 1.6, released on 11/12/97, contains changes from the NANC IIS Version 1.5.

NANC Version 1.7, released on 12/12/97, contains changes from the NANC IIS Version 1.6.

NANC Version 1.8, released on 2/11/98, contains changes from the NANC IIS Version 1.7.

NANC Version 1.9, released on 5/13/98, contains changes from the NANC IIS Version 1.8.

NANC Version 1.10, released on 7/8/98, contains changes from the NANC IIS Version 1.9.

1.4.2 Release 2.0

NANC Version 2.0.0, released on 12/14/98, contains changes from the NANC IIS Version 1.10.

NANC Version 2.0.1, released on 2/25/99, contains changes from the NANC IIS Version 2.0.0.

NANC Version 2.0.2, released on 9/1/99, contains changes from the NANC IIS Version 2.0.1.

1.4.3 Release 3.0

NANC Version 3.0.0, released on 1/28/00 and 2/14/00 (revised version), contains changes from the NANC IIS Version 2.0.2.

NANC Version 3.0.1, released on 6/6/00, contains changes from the NANC IIS Version 3.0.0.
NANC Version 3.0.2, released on 9/11/00, contains changes from the NANC IIS Version 3.0.0.

1.4.4 Release 3.1

NANC Version 3.1.0, released on 8/24/01, contains changes from the NANC IIS Version 3.0.2.

1.4.5 Release 3.2

NANC Version 3.2.0, released on 8/27/02, contains changes from the NANC IIS Version 3.1.0

NANC Version 3.2.1a, released on 7/28/03, contains changes from the NANC IIS Version 3.2.0

NANC Version 3.2.2a, released on 6/30/04, contains changes from the NANC IIS Version 3.2.1a.

1.4.6 Release 3.3

NANC Version 3.3.0a, released on 4/25/05, contains changes from the NANC IIS Version 3.2.2a.

NANC Version 3.3.0b, released on 5/27/05, contains changes from the NANC IIS Version 3.3.0a.

NANC Version 3.3.0c, released on 6/22/05, contains changes from the NANC IIS Version 3.3.0b.

NANC Version 3.3.0d, released on 7/29/05, contains changes from the NANC IIS Version 3.3.0c.

NANC Version 3.3.1a, released on 10/14/05 contains documentation changes from the NANC IIS Version 3.3.0d.

NANC Version 3.3.2a, released on 3/9/2006 contains changes from the NANC IIS Version 3.3.1a.

NANC Version 3.3.3a, released on 2/28/2006 contains changes from the NANC IIS Version 3.3.2a.

1.4.7 Release 3.3.4

NANC Version 3.3.4a, released on 12/08/2009 contains changes from the NANC IIS Version 3.3.3a.

NANC Version 3.3.4b, released on 1/22/2010 contains changes from the NANC IIS Version 3.3.4a.

1.4.8 Release 3.4

NANC Version 3.4.0a, released on 04/02/2010 contains changes from the NANC IIS Version 3.3.4b.

NANC Version 3.4.0b, released on 05/31/2011 contains changes from the NANC IIS Version 3.4.0a.

NANC Version 3.4.2a, released on 02/08/2013 contains the following changes from the NANC IIS Version 3.4.0b:

Introduction

- **Change Order** NANC 448 – NPAC Sunset of non-EDR

NANC Version 3.4.6a, released on 11/30/2013 contains the following changes from the NANC IIS Version 3.4.2a:

- **Change Order** NANC 372 – SOA/LSMS Interface Protocol Alternatives (i.e., NPAC XML Interface)

NANC Version 3.4.6b, released on 02/14/2014 contains the following changes from the NANC IIS Version 3.4.6a:

- **Change Order** NANC 450 – Doc-Only Change Order: FRS/IIS Updates

1.5 References

1.5.1 Standards

ANSI T1.224-1992, *Operations, Administration, Maintenance, and Provisioning (OAM&P) - Protocols for Interfaces between Operations Systems in Different Jurisdictions.*

ANSI T1.243-1995, *Telecommunications, Operations, Administration, Maintenance and Provisioning (OAM&P) - Baseline Security Requirements for the Telecommunications Management Network (TMN).*

ANSI T1.246, *Operations, Administration, Maintenance and Provisioning (OAM&P) - Information Model and Services for Interfaces between Operations Systems across Jurisdictional Boundaries to Support Configuration Management - Customer Account Record Exchange (CARE).*

Bellcore TA- 1253, *Generic Requirements for Operations Interfaces Using OSI Tools: Network Element Security Administration.*

Committee T1 Technical Report No, 40, *Security Requirements for Electronic Bonding Between Two TMNs.*

ISO/IEC 11183-1:1992, *Information Technology - International Standardized Profiles AOM In OSI Management - Management Communications - Part 1 Specification of ACSE, Presentation and Session Protocols for the use by ROSE and CMISE.*

ISO/IEC 11183-2:1992, *Information Technology - International Standardized Profiles AOM In OSI Management - Management Communications - Part 2: CMISE/ROSE for AOM12 - Enhanced Management Communications.*

ISO/IEC 11183-3:1992, *Information Technology - International Standardized Profiles AOM In OSI Management - Management Communications - Part 3: CMISE/ROSE for AOM12 - Basic Management Communications.*

ITU X.509, *Information Technology - Open Systems Interconnection - The Directory Authentication Framework.*

ITU X.690/ISO IS 8825-1 Annex D, *ASNI/BER Encoding of Digital Signatures and Encrypted Cyphertext.*

ITU X.741, *OSI Systems Management, Objects and Attributes for Access Control*

ITU X.803, Upper Layers Security Model.

NMF Forum 016, Issue 1.0, 1992, *OMNIPoint 1 Specifications and Technical Reports, Application Services Security of Management.*

OIW Stable Implementation Agreement, Part 12, 1995.

- Rec. M.3100:1992 & 1995 draft, *Generic Network Information Model*.
- Rec. X.701 | ISO/IEC 10040:1992, *Information Technology - Open System Interconnection - Common Management Overview*.
- Rec. X.710 | ISO/IEC 9595:1990, *Information Technology - Open System Interconnection - Common Management Information Service Definitions*.
- Rec. X.711 | ISO/IEC 9596-1:1991, *Information Technology - Open System Interconnection - Common Management Information Protocol - Part 1: Specification*.
- Rec. X.720 | ISO/IEC 10165-1:1991, *Information Technology - Open System Interconnection - Structure of Management Information - Part 1 Management Information Model*.
- Rec. X.721 | ISO/IEC 10165-2:1992, *Information Technology - Open System Interconnection - Structure of Management Information: Guidelines for the Definition of Managed Objects*.
- Rec. X.722 | ISO/IEC 10165-4:1992, *Information Technology - Open System Interconnection - Structure of Management Information: Guidelines for the Definition of Managed Objects*.
- Rec. X.730 | ISO/10164-1:1992, *Information Technology - Open System Interconnection - System Management - Part 1: Object Management Function*.
- Rec. X.734 | ISO/10164-5:1992, *Information Technology - Open System Interconnection - System Management - Part 5: Event Report Management Function*.
- Rec. X.735 | ISO/10164-6:1992, *Information Technology - Open System Interconnection - System Management - Part 6: Log Control Function*.
- Rec. X.209: 1988, *Specification for Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.
- Rec. X.690: 1994, *ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER), and Distinguished Encoding Rules (DER)*.
- Rec. X.208: 1988, *Specification of Abstract Syntax Notation One (ASN.1)*.
- Rec. X.680 | ISO/IEC 8824-1: 1994, *Information Technology - Abstract Syntax Notation One (ASN.1) - Specification of Basic Notation*.
- Rec. X.680 Amd.1 | ISO/IEC 8824-1 Amd.1, *Information Technology - Abstract Syntax Notation One (ASN.1) - Specification of Basic Notation 1 Amendment 1: Rules of Extensibility*.
- ITU-T Recommendations are available from the US Department of Commerce, National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161. ISO standard are available from the American National Standards Institute, 11 West 42nd Street, New York, NY 10036.

1.5.2 Related Publications

- Illinois Commerce Commission Number Portability Administration Center and Service Management System Request for Proposal (ICC NPAC/SMS RFP)*, February 6, 1996.
- Lockheed Martin Team Response to the Illinois Commerce Commission Number Portability Administration Center and Management System Request for Proposal*, March 18, 1996.
- Scoggins, Sophia and Tang, Adrian 1992. *Open networking with OSI*. Englewood Cliffs, NJ, Prentice-Hall.

Introduction

Stallings, William 1993. *SNMP, SNMPv2, and CMIP, The Practical Guide to Network-Management Standards*, Reading Massachusetts, Addison-Wesley.

North American Number Council (NANC) Functional Requirements Specification, Number Portability Administration Center (NPAC), Service Management System (SMS), Version 3.4.6b February 14, 2014.

NPAC SMS Interoperable Interface Specification (IIS), – Appendix A and B, Errors and Message Flow Diagrams, Version 3.4.6b February 14, 2014.

NPAC SMS XML Interface Specification (XIS), Version 1.5.1, February 14, 2014.

CTIA Report on Wireless Portability Version 2, July 7, 1998

1.6 Abbreviations/Definitions

A-PDU	Application Protocol Data Unit
ASN.1	Abstract Syntax Notation 1
BER	Basic Encoding Rules
CARE	Customer Account Record Exchange
Central Time (standard/daylight)	This is the time in the central time zone, which includes daylight savings time. It changes twice a year based on standard time and daylight savings time. The NPAC SMS runs on hardware that uses this time.
CER	Canonical Encoding Rules
CLASS	Custom Local Area Signaling Services
CME	Conformance Management Entity
CMIP	Common Management Information Protocol
CMISE	Common Management Information Service Element
CNAM	Caller Id with Name
GDMO	Generalized Definitions of Managed Objects
DER	Distinguished Encoding Rules
DES	Data Encryption Standard
FR	Frame Relay
IEC	International Electrotechnical Commission
ISO	International Organization of Standardization
ISVM	Inter-Switch Voice Mail
Local Time	The time zone of the local user. Most time representations in the NPAC OP GUI are represented in the user's local time zone based on the PC's clock setting. The time zone label is included in time display in the GUI. EST for Eastern Time Zone CST for Central Time Zone MST for Mountain Time Zone PST for Pacific Time Zone
LIDB	Line Information Database
LNP	Local Number Portability
LRN	Location Routing Number
LSMS	Local Service Management System
LSPP	Local Service Provider Portability
MAC	Media Access Control
MD5	Message Digest (Version 5)
MIB	Management Information Base
NE	Network Element
NMF	Network Management Forum
NPAC SMS	Number Portability Administration Center and Service Management System
NPA	Numbering Plan Area
NXX	Exchange

Introduction

OCN	Operating Company Number
OSI	Open Systems Interconnect
PPP	Point-To-Point Protocol
RFP	Request for Proposal
RSA	Encryption Scheme
SOA	Service Order Activation
SMS	Service Management System
TMN	Telecommunications Management Network
TN	Telephone Number
URI	Uniform Resource Identifier
WSMSC	Wireless Short Message Service Center

2 Interface Overview

2

2.1 Overview

This specification defines the CMIP interfaces between the NPAC SMS and the service providers' Service Order Entry System and Local SMS. The CMIP interfaces, defined using the CMIP protocol, are referred to as the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface respectively. CMISE M-CREATE, M-DELETE, M-SET, M-GET, M-EVENT-REPORT, and M-ACTION primitives are fully supported in a confirmed mode. Thus, the sequencing of operations is implied by the receipt of the confirmation or operation response, and NOT by the sequence that the operation request is received. The relationship from the SOA to the NPAC SMS and from the Local SMS to NPAC SMS is a manager to agent or an agent to manager relationship depending on the function being performed. The SOA and Local SMS interfaces are defined by Association Functions. These functions allow each association to define the services it supports. Association establishment from the SOAs and Local SMSs to the NPAC SMS, Association Function and security for each of these interfaces is discussed in Section 5, *Secure Association Establishment*.

Note: The M-CANCEL-GET primitive may not be supported in some NPAC SMS implementations due to the fact that this functionality was not determined necessary for the interface defined.

The sections that follow provide an overview of protocol requirements and a brief description of the functionality provided in each interface. Complete functional descriptions for the interfaces are provided in the process flow diagrams in Appendix B, *Message Flow Diagrams*, as well as the behavior for the managed objects.

The interface between the SOA and the NPAC SMS is called the "SOA to NPAC SMS interface". The interface between the Local SMS and the NPAC SMS is called the "NPAC SMS to Local SMS interface". No direction for operations is implied by the names of these interfaces.

All timestamps (GeneralizedTime fields) that are sent over the SOA to NPAC SMS interface and NPAC SMS to Local SMS interface, shall use Greenwich Mean Time (GMT). The universal time format (YYYYMMDDHHMMSS.0Z) is used. The default value is a non-specific format of 00000000000000.0Z.

2.2 OSI Protocol Support

The SOA to NPAC SMS and NPAC SMS to Local SMS interfaces must be implemented over the protocol stack shown in Exhibit 1.

Exhibit 1. NPAC/SMS Primary Network Protocol Stacks

Layer	Mechanized Interface	Function
7	CMIP Agent Server CMISE, ACSE, ROSE	User Application

6	ANSI T1.224	Presentation
5	ANSI T1.224	Session
4	TCP, RFC1006, TPO	Transport
3	IP	Network
2	PPP, MAC, FRAME Relay, ATM (IEEE 802.3)	Link
1	DS-1, DS-0 x n, ISDN, V.34	Physical

Multiple associations per service provider to the NPAC SMS can be supported when using different function masks. The secure association establishment is described in *Section 5*.

2.3 SOA to NPAC SMS Interface

The SOA to NPAC SMS interface, which allows communication between a service provider’s Service Provisioning Operating Systems and/or Gateway systems and the NPAC SMS, supports the retrieval and update of subscription, service provider, and network information. The following transactions occur to support local number portability functionality:

- SOA requests for subscription administration to the NPAC SMS and responses from the NPAC SMS to the SOA.
- Audit requests from the SOA to the NPAC SMS and responses from the NPAC SMS to the SOA.
- Notifications from the NPAC SMS to the SOA of subscription version data and number pool block data changes, needed for concurrence or authorization for number porting, conflict-resolution, cancellation, outage information, customer disconnect dates, or the first use of an NPA-NXX.
- Network data from the NPAC SMS to SOA.
- Service provider data administration from the SOA to the NPAC SMS.
- SOA requests for number pool block administration (creation and modification) to the NPAC SMS and responses from the NPAC SMS to the SOA.
- SPID Migration data from the NPAC SMS to SOA.

Mapping of this functionality into the CMIP Definitions is provided in *Section 4 (see Exhibit 8.)* The NPAC SMS currently uses a 32-bit signed integer for the Naming ID Value. ID value interpretation is based on the way an LNP system treats binary integer numbers. Signed interpretation will see negative numbers when the 32nd bit is used. Unsigned interpretation will always see positive numbers.

Binary Numbers	Signed Numbers	Unsigned Numbers
00000000000000000000000000000001	1	1
00000000000000000000000000000010	2	2
00000000000000000000000000000011	3	3
...
01111111111111111111111111111110	2147483646	2147483646
01111111111111111111111111111111	2147483647	2147483647
	Rollover	
10000000000000000000000000000000	-2147483648	2147483648
10000000000000000000000000000001	-2147483647	2147483649
10000000000000000000000000000010	-2147483646	2147483650
10000000000000000000000000000011	-2147483645	2147483651
...

notified of the customer’s disconnect date. SOA systems are also sent notifications to ensure they are aware of planned down time in the NPAC SMS. Notification of data value changes and object creations are sent for number pool block objects.

First usage notifications are also sent to the SOA when the first use of an NPA-NXX occurs from a subscription version or number pool block creation.

Each SOA notification is assigned a priority of **high**, **medium**, **low** or **none**. The category of **none** indicates that a Service Provider does not want to receive a particular notification. Notifications are then sent in order of priority from **high** to **low**.

SOA Service Providers can receive single or range versions of some notifications. If the service provider’s TN Range Notification Indicator is turned **OFF** in their service provider profile on the NPAC SMS, the following notifications will be sent:

- Attribute Value Change for subscriptionVersionNPAC objects
- Object Creation for subscriptionVersionNPAC objects
- subscriptionVersionCancellationAcknowledgeRequest
- subscriptionVersionDonorSP-CustomerDisconnectDate
- subscriptionVersionNewSP-CreateRequest
- subscriptionVersionNewSP-FinalCreateWindowExpiration
- subscriptionVersionOldSP-ConcurrenceRequest
- subscriptionVersionOldSPFinalConcurrenceWindowExpiration
- subscriptionVersionStatusAttributeValueChange

If the service provider’s TN Range Notification Indicator is turned **ON**, the following notifications will be sent:

- subscriptionVersionRangeAttributeValueChange for subscriptionVersionNPAC objects
- subscriptionVersionRangeCancellationAcknowledgeRequest
- subscriptionVersionRangeDonorSP-CustomerDisconnectDate
- subscriptionVersionRangeNewSP-FinalCreateWindowExpiration
- subscriptionVersionRangeNewSP-CreateRequest
- subscriptionVersionRangeObjectCreation for subscriptionVersionNPAC objects
- subscriptionVersionRangeOldSP-ConcurrenceRequest
- subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration
- subscriptionVersionRangeStatusAttributeValueChange

Notifications can be recovered by the SOA from the NPAC SMS. Notifications to be recovered are requested by time range and are recovered in the order the NPAC SMS attempted to send them. Alternatively, notifications can be recovered using SWIM (Send What I Missed) recovery.

In situations where Subscription Versions are initially created in ranges, then have subsequent activity (modify, activate, disconnect, cancel) performed in singles, TN Range Notifications may change. Specifically, if subsequent activity on a TN range does not equal the initial TN range (subsequent activity is either singles or a subset of the TN range), then initial and final timers (T1, T2) will result in single TN Notifications. TN range requests after the timers would still have the potential to generate TN Range Notifications for Service Providers that support this feature.

2.3.4 Service Provider Data Administration

Service providers can use, read, and update their service provider information on the NPAC SMS using the SOA. Service providers can update some information in the service provider profile as well as add and delete their own network data. Changes to

network data that result in mass updates are prevented from the SOA to the NPAC. Mass changes must be initiated by the service provider contacting the NPAC personnel directly.

2.3.5 Network Data Download

When network data (NPA-NXX, NPA-NXX-X, Service Provider, or LRN data for service providers) is created, modified, or deleted on the NPAC SMS, the data is automatically downloaded from the NPAC SMS to the SOA. The SOA may request that data be recovered using a recovery request that is sent from the SOA to the NPAC SMS. The SOA then receives the data to be recovered in the request response. Network data to be recovered can be requested based on a time range, SWIM data, service provider or all service providers, an NPA-NXX range or all NPA-NXX data, an NPA-NXX-X range or all NPA-NXX-X data, an LRN range or all LRN data, or all network data can be requested. If all network data is specified and the “NPAC Customer SOA NPA-NXX-X Indicator” has been set to TRUE in the service provider’s profile on the NPAC SMS, then NPA-NXX-X object data will be included in the recovery response.

Service providers can also directly read data they wish to download from the NPAC SMS MIB.

2.3.6 Number Pool Block Administration

Number pool blocks are a set of 1000 TNs represented by a 7 digit NPA-NXX-X (i.e. 555-333-1 represents 555-333-1000 through 1999). Service providers can create and modify the number pool blocks for which they are the block holder. Service providers can query all number pool block objects. Only the NPAC Personnel can initiate the removal of a number pool block object.

2.3.7 SPID Migration

Service Providers that support the functionality will receive SPID Migration data over the NPAC SMS to SOA Interface. SPID Migration data is not included in recovery responses.

2.4 NPAC SMS to Local SMS Interface

The NPAC SMS to Local SMS interface is used for communications between a service provider’s Local SMS and the NPAC SMS for support of LNP network element provisioning. The following transactions occur to support Local Number Portability:

- Subscription version, number pool block and network data from the NPAC SMS to the Local SMS.
- Service provider data administration from the Local SMS to the NPAC SMS.
- Notifications from the NPAC SMS to the Local SMS of planned NPAC SMS outages and the first use of a new NPA-NXX.

Mapping of this functionality into the CMIP Definitions is provided in *Section 4 (see Exhibit 8.)* The NPAC SMS currently uses a 32-bit signed integer for the Naming ID Value. ID value interpretation is based on the way an LNP system treats binary integer numbers. Signed interpretation will see negative numbers when the 32nd bit is used. Unsigned interpretation will always see positive numbers.

<u>Binary Numbers</u>	<u>Signed Numbers</u>	<u>Unsigned Numbers</u>
00000000000000000000000000000001	1	1

Service providers can use, read, and update their service provider information on the NPAC SMS using the Local SMS to NPAC SMS interface. Service providers can update some information in the service provider profile as well as add and delete their own network data. Changes to network data that result in mass updates are prevented by the NPAC SMS to Local SMS interface. Mass changes must be initiated by the service provider contacting the NPAC personnel directly.

2.4.3 Notifications

Local SMSs are sent notifications to ensure they are aware of planned down time in the NPAC SMS. Local SMSs are also sent notifications when a new NPA-NXX is to be used for the first time in a subscription version or number pool block by a serviceProvNPA-NXX-X creation.

Notifications can be recovered by the Local SMS from the NPAC SMS. Notifications to be recovered are requested by time range. Alternatively, notifications can be recovered using SWIM recovery.

2.4.4 SPID Migration

Service Providers that support the functionality will receive SPID Migration data over the NPAC SMS to Local SMS Interface. SPID Migration data is not included in recovery responses.

2.5 NPAC and SOA/LSMS Interface Performance

In NPAC Release 3.4, performance requirements were increased for each NPAC region from 4 transactions per second per Service Provider to 7 transactions per second per Service Provider.

An engineering assumption is that Service Providers must support these new performance requirements, such that a Service Provider's local systems will support the minimum throughput rate with each of a Service Provider's specific association to NPAC regions. As Service Providers are responsible for their local systems that support their interfaces to the NPAC (SOA, LSMS, and corresponding downstream network elements), each Service Provider should work with their local system vendors to ensure that the Service Provider's interface solution will adequately support the same industry requirements with the NPAC without impact to other Service Providers in the industry.

It is recommended that each Service Provider spend time working performance requirements with their local system vendors as well as the NPAC vendor.

3 *Hierarchy Diagrams*

3

3.1 Overview

The following five exhibits show the class hierarchy diagram for all managed objects (*Exhibit 2*), Log Record Objects (*Exhibit 3*), the Local SMS (*Exhibit 4*), the NPAC SMS naming hierarchies for the Local SMS (*Exhibit 5*), the SOA (*Exhibit 6.*), and the NPAC SMS naming hierarchies for the SOA. (*Exhibit 7*). These exhibits will help the user gain a better understanding of the structure of the interface definitions provided.

3.1.1 Managed Object Model Inheritance Hierarchy

The Managed Object Model Inheritance Hierarchy shows the inheritance hierarchy used for object definitions in the NPAC SMS to Local SMS and the SOA to NPAC SMS interfaces.

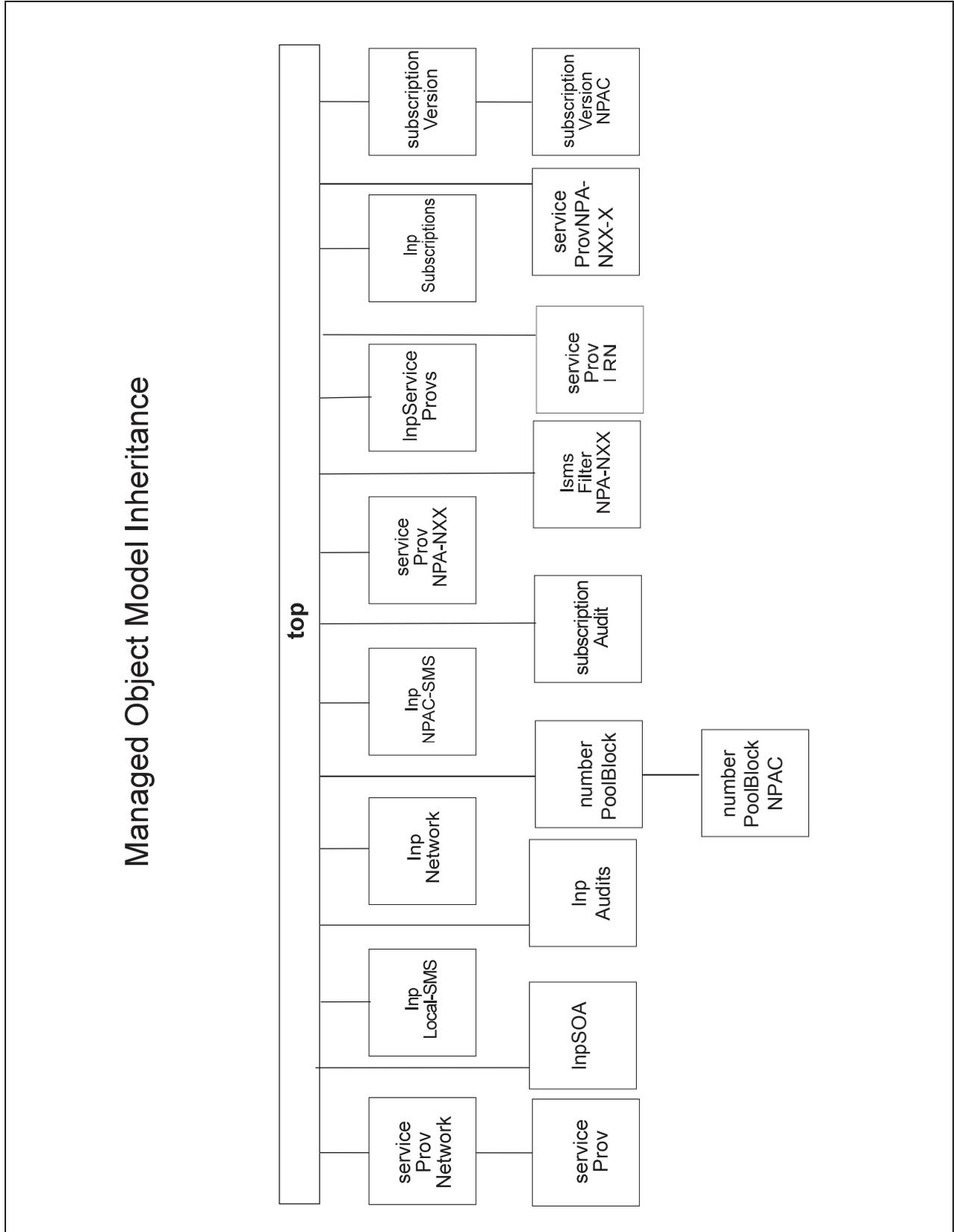
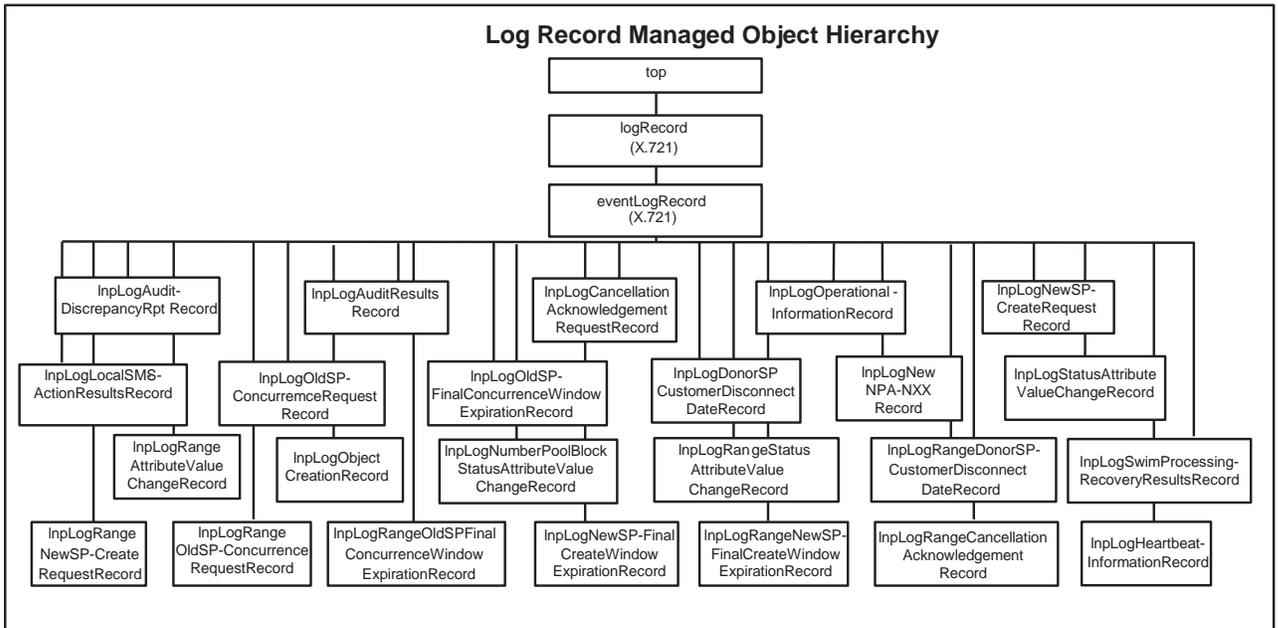


Exhibit 2. The Managed Object Model Inheritance Hierarchy

3.1.2 Log Record Managed Object Hierarchy



The Log Record Managed Object Hierarchy shows the inheritance hierarchy of the log records used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces.

Exhibit 3 . Log Record Managed Object Hierarchy

3.1.3 NPAC SMS to Local SMS Naming Hierarchy for the NPAC SMS

The NPAC SMS to Local SMS Naming Hierarchy for the NPAC SMS shows the naming hierarchy used in the NPAC SMS to instantiate objects defined in the NPAC SMS to Local SMS interface.

Shaded objects are instantiated at NPAC SMS start-up and are not created via M-CREATE or M-DELETE requests. All other objects are created at start-up from a persistent object store on the NPAC SMS or from actions taken while the NPAC SMS is running.

Each object class belongs to one or more Association Functions.
Refer to *Section 5.2.1.8, Association Functions*.

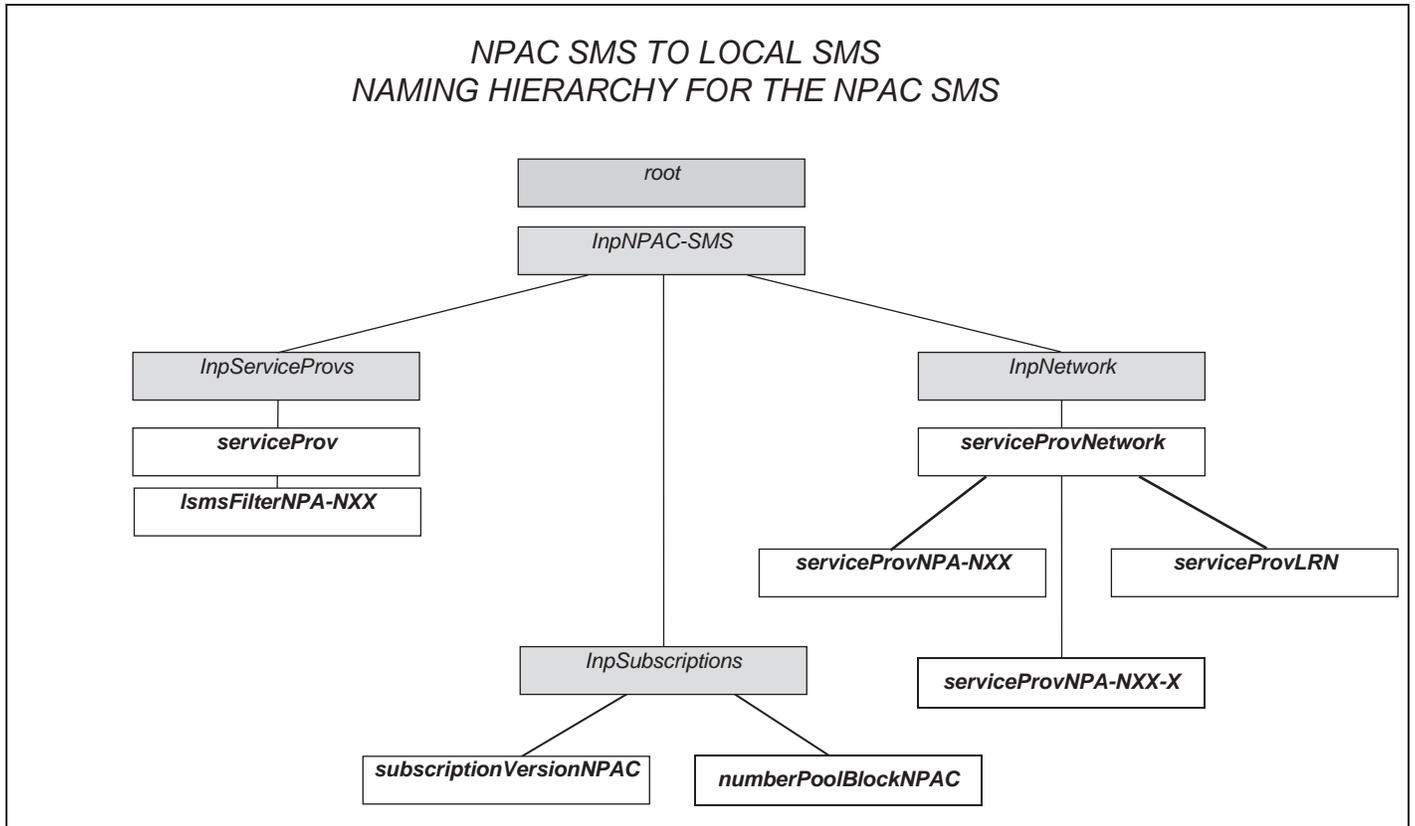


Exhibit 4. The NPAC SMS to Local SMS Naming Hierarchy for the NPAC SMS.

3.1.4 NPAC SMS to Local SMS Naming Hierarchy for the Local SMS

The NPAC SMS to Local SMS Naming Hierarchy for Local SMS shows the naming hierarchy used in the Local SMS to instantiate objects defined in the NPAC SMS to Local SMS interface.

Shaded objects are instantiated at Local SMS start-up and are not created via M-CREATE or M-DELETE requests. All other objects are created at start-up from a persistent object store on the Local SMS or from actions taken while the Local SMS is running.

Each object class belongs to one or more Association Functions.
Refer to *Section 5.2.1.8, Association Functions*.

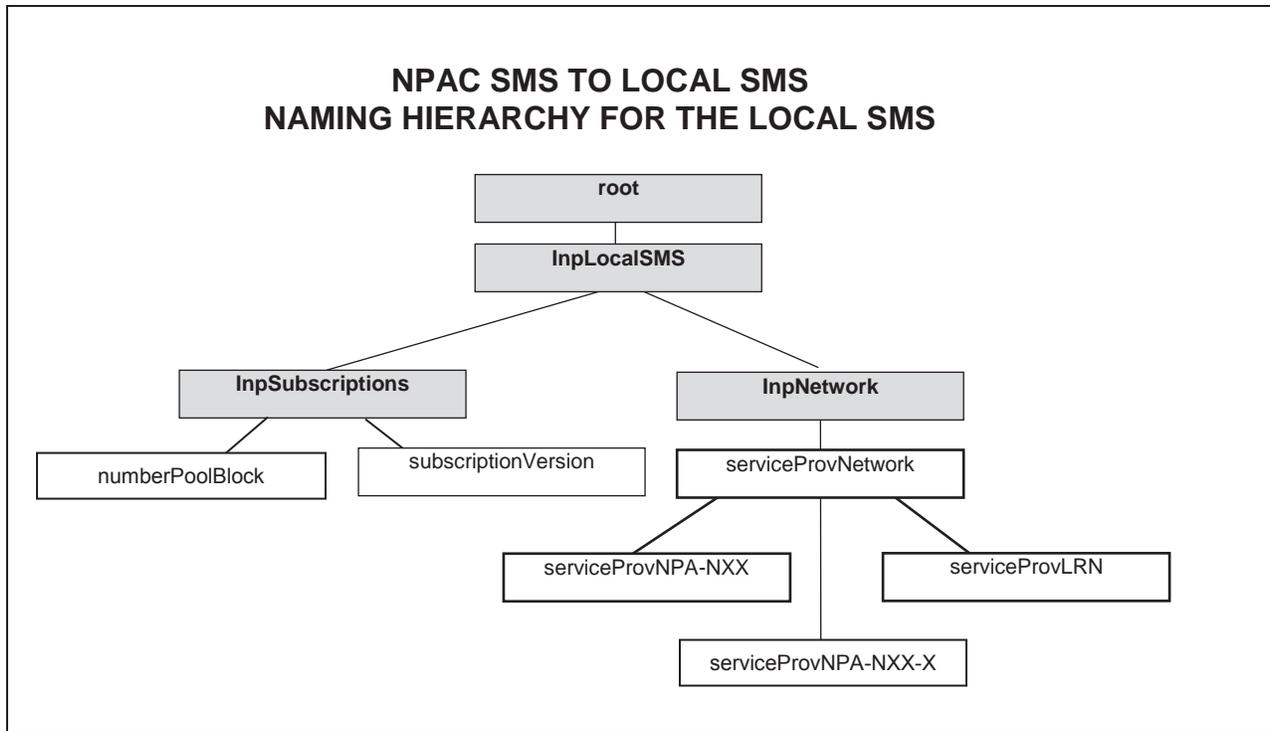


Exhibit 5. The NPAC SMS to Local SMS Naming Hierarchy for the Local SMS.

3.1.5 SOA to NPAC SMS Naming Hierarchy for the NPAC SMS

The SOA to NPAC SMS Naming Hierarchy for the NPAC SMS shows the naming hierarchy used in the NPAC SMS to instantiate objects defined in the SOA to NPAC SMS interface.

Shaded objects are instantiated at NPAC SMS start-up and are not created via M-CREATE or M-DELETE requests. All other objects are created at start-up from a persistent object store on the NPAC SMS or from actions taken while the NPAC SMS is running.

Each object class belongs to one or more Association Functions.
Refer to *Section 5.2.1.8, Association Functions*.

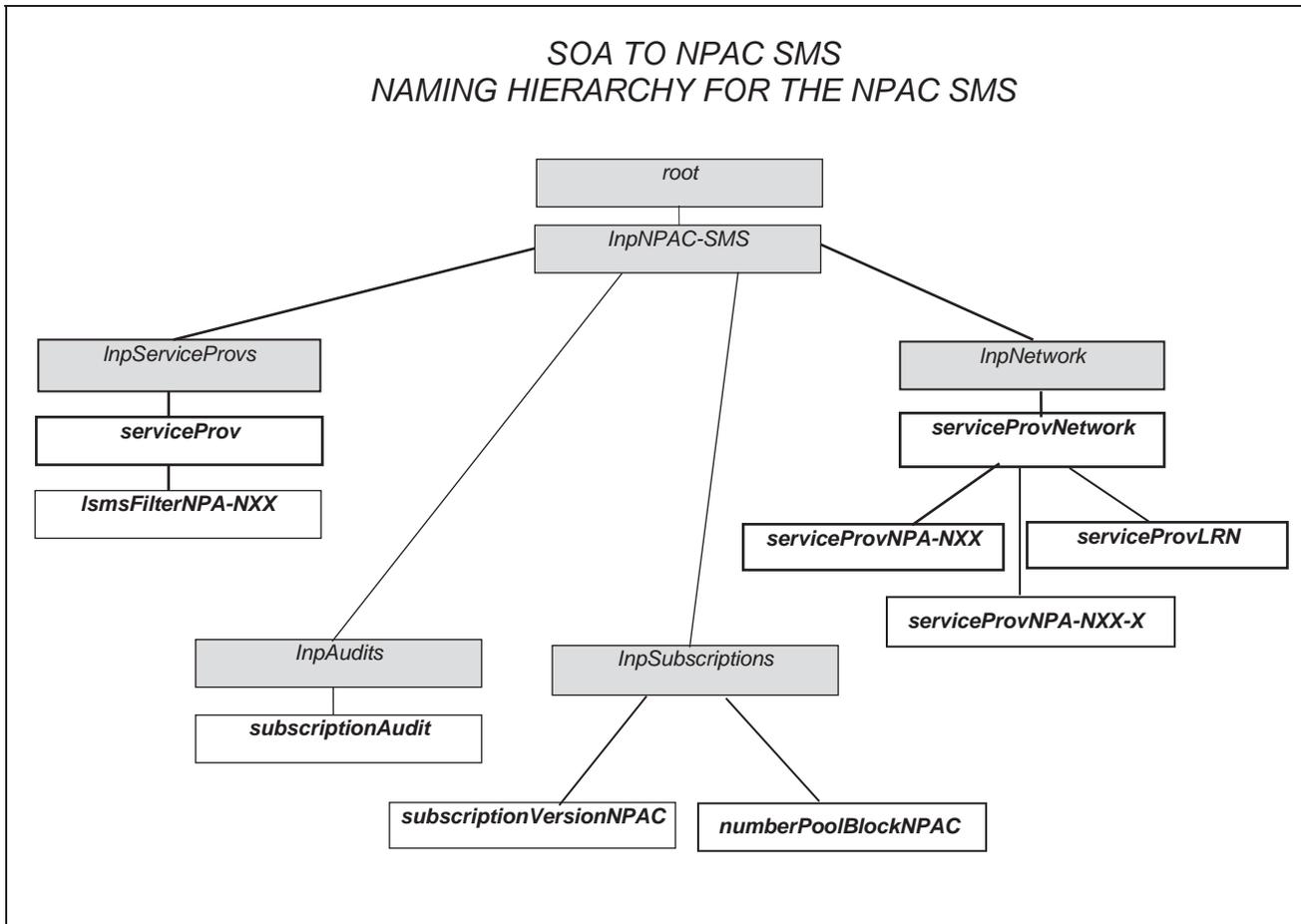


Exhibit 6. The SOA to NPAC SMS Naming Hierarchy for the NPAC SMS.

3.1.6 NPAC SMS to SOA Naming Hierarchy for the SOA

The NPAC SMS to SOA Naming Hierarchy for SOA shows the naming hierarchy used in the SOA to instantiate objects defined in the SOA to NPAC SMS interface.

Shaded objects are instantiated at SOA start-up and are not created via M-CREATE or M-DELETE requests. All other objects are created at start-up from a persistent object store on the SOA or from actions taken while the SOA is running.

Each object class belongs to one or more Association Functions. Refer to Section 5.2.1.8, **Association Functions**.

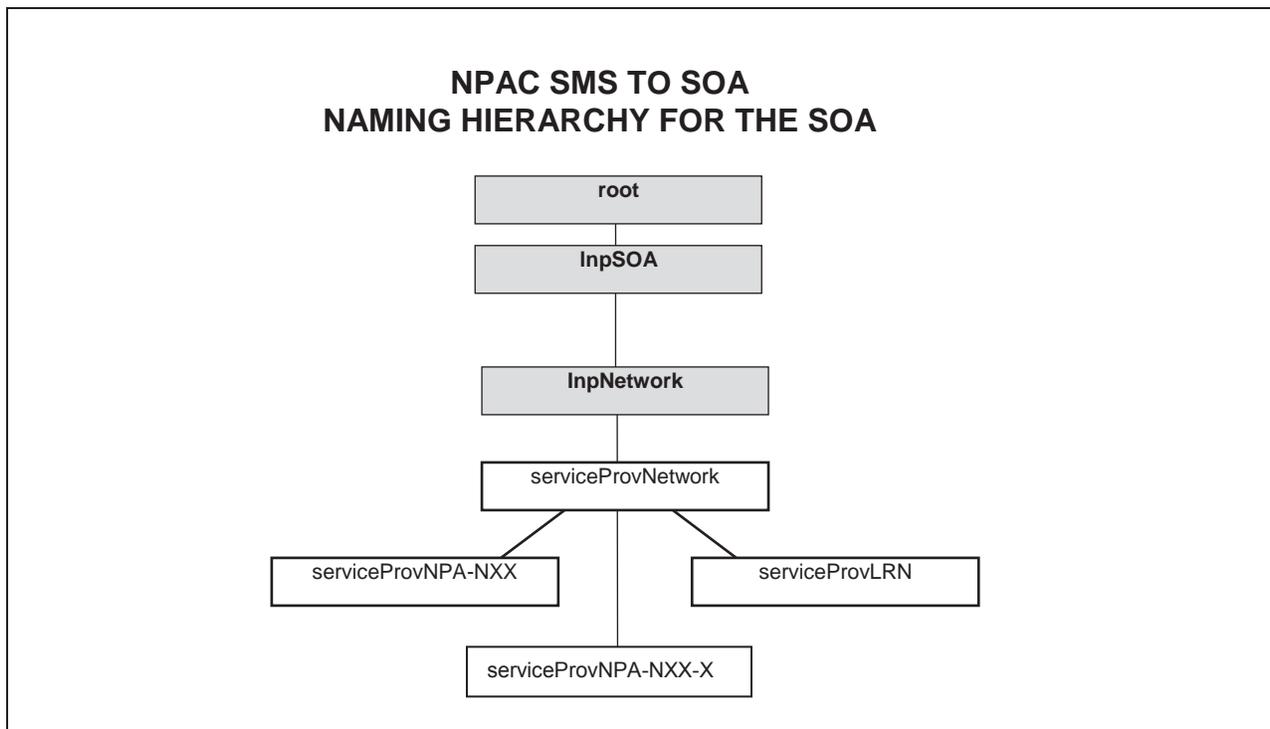


Exhibit 7. NPA SMS to SOA Naming Hierarchy for the SOA.

4 Interface Functionality to CMIP Definition Mapping

4

4.1 Overview

The following tables, Exhibits 8-12, contain the mapping of the interface functionality to managed objects, attributes, actions, and notifications.

4.1.1 Primary NPAC Mechanized Interface Operations

The primary interface functions in support of the NPAC requirements are described in the table below, as well as their corresponding Common Management Information Exchange (CMISE) operation and referenced object type for that operation. This table does not include miscellaneous operations, such as service provider network data querying or downloading, etc. These functions are described in the object behaviors in the GDMO source below.

Exhibit 8. Primary NPAC Mechanized Interface Operations Table

Function	Direction (To/From)	CMIP Operation	Referenced Object Type
Abort/Cancel Audit Request	from SOA	M-DELETE	subscriptionAudit
Audit Complete	to SOA	M-EVENT-REPORT: subscriptionAuditResults	subscriptionAudit
Audit Discrepancy	to SOA	M-EVENT-REPORT: subscriptionAudit-DiscrepancyRpt	subscriptionAudit
Audit Query	from SOA	M-GET	subscriptionAudit
Audit Request SOA	from SOA	M-CREATE	subscriptionAudit
Cancellation Acknowledgement	from SOA (new service provider)	M-ACTION: subscriptionVersionNewSP-CancellationAcknowledge	InpSubscriptions
Cancellation Acknowledgement	from SOA (old service provider)	M-ACTION: subscriptionVersionOldSP-CancellationAcknowledge	InpSubscriptions
Conflict Removal	from SOA (new service provider)	M-ACTION: subscriptionVersionRemoveFromConflict	InpSubscriptions

Function	Direction (To/From)	CMIP Operation	Referenced Object Type
Customer Disconnect Date	to SOA	M-EVENT-REPORT: subscriptionVersionDonorSP-CustomerDisconnectDate or subscriptionVersionRangeDonorSP-CustomerDisconnectDate	subscriptionVersionNPAC or InpSubscriptions
Final Request for Version Create	to SOA (old service provider)	M-EVENT-REPORT: subscriptionVersionOldSPFinalConcurrenceWindowExpiration or subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration	subscriptionVersionNPAC or InpSubscriptions
LSMS Filter NPA-NXX Create	from LOCAL SMS or from SOA	M-CREATE	lsmsFilterNPA-NXX
LSMS Filter NPA-NXX Delete	from LOCAL SMS or from SOA	M-DELETE	lsmsFilterNPA-NXX
LSMS Filter NPA-NXX Query	from LOCAL SMS or from SOA	M-GET	lsmsFilterNPA-NXX
Network Data Download	from LOCAL SMS or from SOA	M-ACTION: InpDownload or M-GET: scoped and filtered for intended serviceProvLRN, serviceProvNPA-NXX serviceProvNPA-NXX-X, service provider attributes	InpNetwork
Network Data Update	from LOCAL SMS or from SOA	M-CREATE	serviceProvLRN, serviceProvNPA-NXX
NPA-NXX Modify	to LOCAL SMS or to SOA	M-SET	serviceProvNPA-NXX
NPA-NXX-X Create	to LOCAL SMS or to SOA	M-CREATE;	serviceProvNPA-NXX-X
NPA-NXX-X Delete	to LOCAL SMS or to SOA	M-DELETE	serviceProvNPA-NXX-X
NPA-NXX-X Modify	to LOCAL SMS or to SOA	M-SET	serviceProvNPA-NXX-X

Function	Direction (To/From)	CMIP Operation	Referenced Object Type
New NPA-NXX	to LOCAL SMS or to SOA	M-EVENT-REPORT: subscriptionVersionNewNPA-NXX	SubscriptionVersionNPAC InpNPAC-SMS
Number Pool Block Change Notification	to SOA	M-EVENT-REPORT attributeValueChange Notification or numberPoolBlockStatusAttributeValueC hange Notification	numberPoolBlockNPAC
Number Pool Block Create	from SOA	M-ACTION: numberPoolBlock-Create	InpSubscriptions
Number Pool Block Create	to LOCAL SMS	M-CREATE: for a single numberPoolBlock	numberPoolBlock
Number Pool Block Modify	from SOA	M-SET: to a single numberPoolBlock	numberPoolBlockNPAC or InpSubscriptions
Number Pool Block Modify	to LOCAL SMS	MSET: to a single numberPoolBlock or scoped and filtered by NPA-NXX-X range for mass update	numberPoolBlock or InpSubscriptions
Number Pool Block Delete	to LOCAL SMS	M-DELETE: for a single numberPoolBlock	numberPoolBlock
Number Pool Block Query	from LOCAL SMS or SOA	M-GET: To a single numberPoolBlockNPAC or scoped and filtered for intended numberPoolBlocks	InpSubscriptions numberPoolBlockNPAC
Number Pool Block Query	to LOCAL SMS	M-GET: scoped and filtered for intended numberPoolBlock	InpSubscriptions
Notification Recovery	from LOCAL SMS or from SOA	M-ACTION: InpNotificationRecovery	InpNPAC-SMS
Recovery Complete	from LOCAL SMS or from SOA	M-ACTION: InpRecoveryComplete	InpNPAC-SMS
Request for Cancellation Acknowledgment	to SOA	M-EVENT-REPORT: subscription VersionCancellationAcknowledgment Request or subscriptionVersionRangeCancellationA cknowledgeRequest	subscriptionVersionNPAC or InpSubscriptions
Request for Version Create	to SOA (new service provider)	M-EVENT-REPORT: subscriptionVersionNewSP-Create Request or subscriptionVersionRangeNewSP- CreateRequest	subscriptionVersionNPAC

Function	Direction (To/From)	CMIP Operation	Referenced Object Type
Request for Version Create	to SOA (old service provider)	M-EVENT-REPORT: subscriptionVersionOldSP-Concurrence Request or subscriptionVersionRangeOldSP-ConcurrenceRequest	subscriptionVersionNPAC or InpSubscriptions
Service Provider Network Creation	to LOCAL SMS or to SOA	M-CREATE	serviceProvNetwork
Service Provider Network Deletion	to LOCAL SMS or to SOA	M-DELETE	serviceProvNetwork
Service Provider Network Service Provider Name Change	to LOCAL SMS or to SOA	M-SET: serviceProvName	serviceProvNetwork
SPID Migration	from LOCAL SMS or from SOA	M-ACTION: InpSpidMigration	InpNetwork
Subscription Version Activate	from SOA	M-ACTION: subscriptionVersionActivate	InpSubscriptions
Subscription Version Cancel	from SOA	M-ACTION subscriptionVersionCancel	InpSubscriptions
Subscription Version Change Notification	to SOA	M-EVENT-REPORT: attributeValueChangeNotification and subscriptionVersionStatusAttributeValue Change or subscriptionVersionRangeAttribute ValueChange subscriptionVersionRangeStatusAttribute ValueChange	subscriptionVersionNPAC or InpSubscriptions
Subscription Version Conflict	from SOA (old service provider)	M-ACTION: subscriptionVersionOldSP-Create setting subscriptionOldSP-Authorization = FALSE	subscriptionVersion
Subscription Version Create	to LOCAL SMS	M-ACTION: subscriptionVersionLocalSMS-Create for multiple creates (i.e., range operations) where the data in the subscription versions is the same M-CREATE: for an individual subscriptionVersion	InpSubscriptions subscriptionVersion
Subscription Version Create	from SOA	M-ACTION: subscriptionVersionOldSP-Create or subscriptionVersionNewSP-Create	InpSubscriptions
Subscription Version Delete	to LOCAL SMS	M-DELETE: scoped and filtered for intended subscriptionVersion criteria	subscriptionVersion

Function	Direction (To/From)	CMIP Operation	Referenced Object Type
Subscription Version Disconnect	from SOA	M-ACTION: subscriptionVersionDisconnect	InpSubscriptions
Subscription Version Download	to LOCAL SMS	M-ACTION: subscriptionVersionLocalSMS-Create or M-CREATE: for an individual subscriptionVersion	InpSubscriptions
Subscription Version Download Request	from LOCAL SMS	M-ACTION: InpDownload or M-GET: scoped and filtered for intended subscriptionVersionNPAC criteria	InpSubscriptions
Subscription Version Modify	from SOA	M-ACTION: subscriptionVersion Modify or M-SET: on relevant subscriptionVersionNPAC attributes for pending and conflict versions	InpSubscriptions
Subscription Version Modify	to LOCAL SMS	M-SET: scoped and filtered for intended subscriptionVersion criteria setting relevant attributes	InpSubscriptions
Subscription Version Query	from SOA from LOCAL SMS	M-GET: scoped and filtered for intended subscriptionVersionNPAC criteria setting relevant attributes	InpSubscriptions
Subscription Version Query	to LOCAL SMS	M-GET: scoped and filtered for intended subscriptionVersion criteria	InpSubscriptions

4.1.2 Managed Object Interface Functionality

The table below contains the mapping of the SOA to NPAC SMS and the Local SMS to NPAC SMS managed objects to the interface functionality.

Exhibit 9. Managed Object Interface Functionality Table

Managed Object Name	Interface Functionality Mapping
InpAudits	Container object used to contain all subscription audit objects on the NPAC SMS and the Local SMS. It is used in the SOA to NPAC SMS interface to support audit functionality.
InpLocal SMS	Container object used to contain all objects on a Local SMS. It is used in the NPAC SMS to Local SMS interface to support NPAC SMS communication to the service provider Local SMS system.

Managed Object Name	Interface Functionality Mapping
InpLogAudit-DiscrepancyRptRecord	Object used to log information from a subscriptionAuditDiscrepancyRpt notification.
InpLogAuditResultsRecord	Object used to log information from a subscriptionAuditResults notification.
InpLogCancellationAcknowledgeRequestRecord	Object used to log information from a subscriptionVersionCancellationAcknowledgeRequest notification.
InpLogDonorSP-CustomerDisconnectDateRecord	Object used to log information from a subscriptionVersionDonorSP-CustomerDisconnectDate notification.
InpLogLocalSMS-ActionResultsRecord	Object used to log information from a subscriptionVersionLocalSMS-ActionResults notification.
InpLogNewNPA-NXXRecord	Object used to log information from a subscriptionVersionNewNPA-NXX notification.
InpLogNewSP-CreateRequestRecord	Object used to log information from a subscriptionVersionNewSP-CreateRequest notification.
InpLogNumberPoolBlockStatusAttributeValueChangedRecord	Object used to log information from a numberPoolBlockStatusAttributeValueChanged notification.
InpLogOldSP-ConcurrenceRequestRecord	Object used to log information from a subscriptionVersionOldSP-ConcurrenceRequest notification.
InpLogOldSP-FinalConcurrenceWindowExpiration	Object used to log information from a subscriptionVersionOldSPFinalConcurrenceWindowExpiration notification.
InpLogOperational-InformationRecord	Object used to log information from a InpNPAC-SMS-Operational-Information notification.
InpLogRangeAttributeValueChangedRecord	Object used to log information from a InpLogRangeAttributeValueChanged notification.
InpLogRangeObjectCreationRecord	Object used to log information from a InpLogRangeObjectCreation notification.
InpLogRangeStatusAttributeValueChangedRecord	Object used to log information from a InpLogRangeStatusAttributeValueChanged notification.
InpLogRangeDonorSP-CustomerDisconnectDateRecord	Object used to log information from a InpLogRangeDonorSP-CustomerDisconnectDate notification.
InpLogRangeCancellationAcknowledgeRecord	Object used to log information from a InpLogRangeCancellationAcknowledge notification.
InpLogRangeNewSP-CreateRequestRecord	Object used to log information from a InpLogRangeNewSP-CreateRequest notification.

Managed Object Name	Interface Functionality Mapping
InpLogRangeNewSP-FinalCreateWindowExpirationRecord	Object used to log information from a InpLogRangeNewSP-FinalCreateWindowExpiration notification.
InpLogNewSP-FinalCreateWindowExpirationRecord	Object used to log information from a InpLogNewSP-FinalCreateWindowExpiration notification.
InpLogRangeOldSP-ConcurrenceRequestRecord	Object used to log information from a InpLogRangeOldSP-ConcurrenceRequest notification.
InpLogRangeOldSPFinalConcurrenceWindowExpirationRecord	Object used to log information from a InpLogRangeOldSPFinalConcurrenceWindowExpiration notification.
InpLogStatusAttributeValueChangedRecord	Object used to log information from a subscriptionVersionStatusAttributeValueChanged notification.
InpLogHeartbeat-InformationRecord	Object used to log information from a Heartbeat-Information notification.
InpLogSwimProcessing-RecoveryResultsRecord	Object used to log information from a swimProcessing-RecoveryResults notification.
InpNetwork	Container object used to contain all service provider network data on the NPAC SMS, SOA, and Local SMS. It is used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces to support downloading of network data to the Local SMS and/or SOA and the functionality that allows service providers to create/delete their network data on the NPAC SMS; it is also used to send SPID Migration data to Service Providers that support the information over the interface.
InpNPAC-SMS	Container object used to contain all objects on a NPAC SMS. It is used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces to support NPAC SMS communication from the service provider Local SMS and the SOA systems.
InpServiceProvs	Container object used to contain all service provider data on the NPAC SMS. It is used in the NPAC SMS to Local SMS interface and SOA to NPAC SMS interface to support retrieving of service provider data by the Local SMS and/or SOA and the functionality that allows service providers to update their service provider data on the NPAC SMS. Service providers can only retrieve their service provider data.
InpSOA	Container object used to contain all objects on a SOA. It is used in the SOA to NPAC SMS interface to support NPAC SMS communication to the service provider SOA system.
InpSubscriptions	Container object used to contain all subscription versions and number pool blocks on the NPAC SMS and the Local SMS. It is used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces to support query of subscription and number pool block data on the NPAC SMS and downloading of subscription and number pool block data to the Local SMS.
IsmsFilterNPA-NXX	Object used to represent the NPA-NXX values for which a service provider does not want to be informed of subscription version broadcasts.

Managed Object Name	Interface Functionality Mapping
numberPoolBlock	Object used to represent a number pool block on the Local SMS. These objects are used to support number pool block download from the NPAC SMS to the Local SMS using the NPAC SMS to Local SMS interface.
numberPoolBlockNPAC	Object used to represent a number pool block on the NPAC SMS. These objects are used to support number pool block administration from the SOA using the SOA to NPAC SMS interface. Capability is provided to the SOA for creation and modification. The NPAC SMS can create, modify and delete.
serviceProv	Object used to represent a service provider and its associated data on the NPAC SMS. These objects are used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces to support retrieving of service provider data and the functionality that allows service providers to update their service provider data on the NPAC SMS except serviceProvId and serviceProvType. Service providers can only retrieve their service provider data.
serviceProvLRN	Object used to represent an LRN associated with a service provider on the NPAC SMS, SOA, or Local SMS. These objects are used to support downloading of network LRN data to the Local SMS and/or SOA and the functionality that allows service providers to create/delete their own network LRN data. The service provider will have to add a new object and delete the old one to modify the data.
serviceProvNetwork	Container object used to contain network data for a service provider on the NPAC SMS, SOA or Local SMS. It is used in the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces to support downloading of network data to the Local SMS and the functionality that allows service providers to update their network data on the NPAC SMS.
serviceProvNPA-NXX	Object used to represent an NPA-NXX associated with a service provider on the NPAC SMS, SOA or Local SMS. These objects are used to support downloading of network NPA-NXX data to the Local SMS and/or SOA and the functionality that allows service providers to create/delete their own network NPA-NXX data. NPA splits are supported only through direct contact with NPAC personnel. NPA-NXX Effective Date modification is supported only through direct contact with NPAC personnel.
serviceProvNPA-NXX-X	Object used to represent an NPA-NXX-X associated with a service provider on the NPAC SMS, SOA or Local SMS. These objects are used in number pooling to support downloading of network NPA-NXX-X data to the Local SMS or SOA. Only the NPAC SMS is allowed to create, delete and modify a service provider's NPA-NXX-X data. Local SMS may support this object by setting the "NPAC Customer LSMS NPA-NXX-X Indicator" to TRUE in their service provider profile on the NPAC SMS. SOA may support this object by setting the "NPAC Customer SOA NPA-NXX-X Indicator" to TRUE in their service provider profile on the NPAC SMS.
subscriptionAudit	Object used to represent a subscription audit request on the NPAC SMS. These objects are used to support subscription audit requests from the SOA to the NPAC SMS using the SOA to NPAC SMS interface. The object supports notifications for audit discrepancies found and audit completion results. If the subscription version LNP type is equal to 'pool', the appropriate number pool block will also be audited.

Managed Object Name	Interface Functionality Mapping
subscriptionVersion	Object used to represent a subscription version on the Local SMS. These objects are used to support subscription version download from the NPAC SMS to the Local SMS using the NPAC SMS to Local SMS interface
subscriptionVersionNPAC	Object used to represent a subscription version on the NPAC SMS. These objects are used to support subscription administration from the SOA using the SOA to NPAC SMS interface. Capability is provided for version creation, activation, modification, cancellation, disconnect, and query.

4.1.3 Action Interface Functionality

The table below contains the mapping of the SOA to NPAC SMS and the Local SMS to NPAC SMS actions to the interface functionality.

Exhibit 10. The Action Interface Functionality Table

Action Name	Interface Requirements Mapping
InpDownload	This action is used to support the downloading of subscription, number pool block and network data to the Local SMS from the NPAC SMS. It also supports the downloading of network data to the SOA from the NPAC SMS.
InpRecoveryComplete	This action is used to specify the system has recovered from down time and the transactions performed since the association establishment can now be sent to the Local SMS from the NPAC SMS using the Local SMS to NPAC SMS interface or the SOA from the NPAC SMS using the SOA to NPAC SMS interface.
InpNotificationRecovery	This action is used to support the downloading of notification data to the SOA and/or Local SMS from the NPAC SMS.
InpSpidMigration	This action is used to support the downloading of SPID Migration data to the Local SMS from the NPAC SMS. It also supports the downloading of SPID Migration data to the SOA from the NPAC SMS.
numberPoolBlock-Create	This action is used to support creation of the number pool block object by the block holder service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionActivate	This action is used to support subscription version activation by the new service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionCancel	This action is used to support subscription version cancellation by a service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionDisconnect	This action is used to support subscription version disconnection by the current service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionLocalSMS-Create	This action can be used by the NPAC SMS to create multiple subscription versions via the Local SMS to NPAC SMS interface.
subscriptionVersionModify	This action is used to support subscription version modification by a service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionNewSP-CancellationAcknowledge	This action is used to support the acknowledgment of subscription versions with a status of cancel-pending by the old service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionNewSP-Create	This action is used to support subscription version creation by the new service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.

Action Name	Interface Requirements Mapping
subscriptionVersionOldSP-CancellationAcknowledge	This action is used to support the acknowledgment of subscription versions with a status of cancel-pending by the old service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersionOldSP-Create	This action is used to support subscription version creation by the old service provider from the SOA to the NPAC SMS using the SOA to NPAC SMS interface.
subscriptionVersion RemoveFromConflict	This action is used on the NPAC SMS via the SOA to NPAC SMS interface to set the subscription version status from conflict to pending.
InpNotificationRecovery	This action is used on the NPAC SMS via the SOA to NPAC SMS or Local SMS to NPAC SMS interface to recover notifications.
subscriptionVersionActivateWithError Code	This action is used on the SOA to NPAC SMS interface by the new service provider to activate a subscription version id, TN or range of TNs via the SOA to NPAC SMS interface. This action's reply contains an optional error code to be returned if the action is not successful.
subscriptionVersionCancelWithErrorCode	The action issued on the SOA to NPAC SMS interface by the SOA to cancel a subscription version. This action's reply contains an optional error code to be returned if the action is not successful.
subscriptionVersionNewSP-CancellationAcknowledgeWithErrorCode	This action is used on the SOA to NPAC SMS interface by the new service provider to acknowledge cancellation of a subscriptionVersionNPAC with a status of cancel-pending. This action's reply contains an optional error code to be returned if the action is not successful.
subscriptionVersionRemoveFromConflictWithErrorCode	This action used on the SOA to NPAC SMS interface by either the old or new service provider to set the subscription version status from conflict to pending. This action's reply contains an optional error code to be returned if the action is not successful.
subscriptionVersionOldSP-CancellationAcknowledgeWithErrorCode	This action is used on the SOA to NPAC SMS interface by the old service provider to acknowledge cancellation of a subscriptionVersionNPAC with a status of cancel-pending. This action's reply contains an optional error code to be returned if the action is not successful.

4.1.4 Notification Interface Functionality

The table below contains the mapping of the SOA to NPAC SMS and the Local SMS to NPAC SMS notifications to the interface functionality.

Exhibit 11. The Notification Interface Functionality Table

Notification Name	Interface Requirements Mapping
-------------------	--------------------------------

Notification Name	Interface Requirements Mapping
InpNPAC-SMS-Operational-Information	This notification is used to support the reporting of NPAC SMS scheduled down time. This notification can be issued from the InpNPAC-SMS object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface or from the NPAC SMS to the Local SMS via the NPAC SMS to Local SMS interface.
numberPoolBlockStatusAttributeValueChange	This notification is issued when the number pool block status is modified and can contain the number pool block status and failed service provider list. This notification is issued over the NPAC SMS to SOA interface from the numberPoolBlockNPAC object.
subscriptionAuditDiscrepancyRpt	This notification is used to support the reporting of audit discrepancies found during audit processing. This notification can be issued from an audit object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface.
subscriptionAuditResults	This notification is used to support the reporting of audit processing results. This notification can be issued from an audit object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface.
subscriptionVersionCancellationAcknowledgeRequest or subscriptionVersionRangeNewSP-CancellationAcknowledge	This notification is issued to new and old service providers to request that a cancellation acknowledgment be sent for a subscription version in a cancel-pending state. This notification is issued via the SOA to NPAC SMS interface if the service provider fails to acknowledge the cancellation after a tunable amount of time specified in the NPAC SMS. The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.
subscriptionVersionDonorSP-CustomerDisconnectDate or subscriptionVersionRangeDonorSP-CustomerDisconnectDate	This notification informs the donor service provider SOA that a subscription version is being disconnected. This notification is issued from the NPAC SMS to a SOA via the SOA to NPAC SMS interface. The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.
subscriptionVersionLocalSMS-ActionResults	This notification contains the results of a subscriptionVersionLocalSMS-Create action once all the create requests have been attempted. It is issued from the Local SMS to the NPAC SMS via the NPAC SMS to Local SMS interface.
subscriptionVersionNew-NPA-NXX	This notification informs the Local SMS or SOA of a pending subscription version or new number pool block involving the first use of an NPA-NXX.

Notification Name	Interface Requirements Mapping
subscriptionVersionNewSP-CreateRequest or subscriptionVersionRangeNewSP-CreateRequest	<p>This notification is issued to the new service provider to request that a create request be sent for the subscription version created by the old service provider to provide authorization and/or porting information. This notification is issued via the SOA to NPAC SMS interface if the new service provider failed to authorize porting of a number after a tunable amount of time specified in the NPAC SMS.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
subscriptionVersionNewSPFinalCreateWindow Expiration or subscriptionVersionRangeNewSPFinalCreateWindow Expiration	<p>This notification is issued to the new and old service provider, if they support the Final Create Window Expiration Notification in their Service Provider profile, to inform them of the expiration of the Final Concurrence Window on the NPAC SMS. This notification is issued from the NPAC SMS to the SOA via the SOA to NPAC SMS interface.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
subscriptionVersionOldSP-ConcurrenceRequest or subscriptionVersionRangeOldSP-ConcurrenceRequest	<p>This notification is issued to the old service provider to request that a create request be sent for the subscription version created by the new service provider to provide concurrence for porting. This notification is issued via the SOA to NPAC SMS interface if the old service provider failed to authorize porting of a number after a tunable amount of time specified in the NPAC SMS.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
subscriptionVersionStatusAttributeValueChange or subscriptionVersionRangeStatusAttributeValue Change	<p>This notification is issued when the subscription version status is modified. This notification is issued from the NPAC SMS to the SOA via the SOA to NPAC SMS interface.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
subscriptionVersionOldSPFinalConcurrenceWindow Expiration or subscriptionVersionRangeOldSPFinalConcurrence WindowExpiration	<p>This notification is issued to the old service provider to request for a final time that a create request be sent for the subscription version created by the new service provider to provide concurrence for porting. This notification is issued via the SOA to NPAC SMS interface if the old service provider failed to authorize porting of a number after a tunable amount of time.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>

Notification Name	Interface Requirements Mapping
subscriptionVersionRangeAttributeValueChange	<p>This notification or the Attribute Value Change notification is sent when specified attributes have been updated. This notification is issued via the SOA to NPAC SMS interface.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
subscriptionVersionRangeObjectCreation	<p>This notification or the object creation notification is sent when a subscriptionVersionNPAC object has been created. This notification is issued via the SOA to NPAC SMS interface.</p> <p>The NPAC SMS sends the appropriate notification depending upon the Service Provider's TN Range Notification Indicator.</p>
ApplicationLevelHeartBeat	<p>This notification implements a SOA or LSMS Application Level Heartbeat function. With this functionality the NPAC SMS will send a periodic Heartbeat message when a quiet period between the SOA/LSMS and the NPAC SMS exceeds the tunable value.</p> <p>This notification is prioritized and transmitted according to its SOA Notification Priority tunable in the NPAC SMS when sent over the NPAC SMS to SOA interface.</p> <p>Optionally, this notification may also be implemented on the SOA or Local SMS. With this functionality the SOA/Local SMS will send a periodic Heartbeat message when a quiet period between the SOA/Local SMS and the NPAC SMS exceeds the tunable value.</p> <p>This notification can be issued via the NPAC SMS to SOA and NPAC SMS to Local SMS interfaces. Optionally, this notification can also be issued via the SOA to NPAC SMS and LSMS to NPAC SMS interfaces.</p>
swimProcessing-RecoveryResults	<p>This notification contains the recovery results of a SWIM InpDownload action or SWIM InpNotificationRecovery action from a SOA/LSMS.</p> <p>This notification is issued via the SOA to NPAC SMS interface and the Local SMS to NPAC SMS interface.</p>

4.2 Scoping and Filtering Support

The following section defines the scoping and filtering support for both the SOA to NPAC SMS interface and LSMS to NPAC SMS interface.

4.2.1 Scoping

The NPAC SMS to Local SMS or SOA to NPAC SMS interfaces do not support scoping of CMIP operations of any type by the LSMS or SOA for the following objects:

- root
- InpLocal-SMS
- InpNetwork
- any object with an “empty” filter

NPAC SMS is not required to support Scope other than baseObject Scope for CMIP operations that specify baseManagedObjectClass of one of the following:

- InpNPAC-SMS
- InpServiceProvs

Scoped operations for subscriptionVersions or numberPoolBlocks to the LSMS must be supported on the baseObject (level 0) or from the InpSubscriptions object with a non-empty filter.

The limit in scoping and functionality prevents the NPAC, SOA, and the LSMS systems from having to implement functionality or respond to large requests that are not necessary to support LNP over the mechanized interfaces.

4.2.2 Filtering

Filtering on the NPAC SMS is supported as defined in the GDMO. The NPAC SMS requires the Local SMS to support at a minimum the filter criteria specified below.

Limitations:

- OR and NOT filter support is not required for the Local SMS or SOA.
- NOT filter support is not required for the NPAC SMS.
- Filtering requests with a scope will not be issued to the Local SMS or SOA by the NPAC SMS for any object other than the subscriptionVersion and numberPoolBlock objects. No query will be used that requests both subscription versions and number pool blocks at the same time..
- All authorization rules apply to scoped and filtered operations. For example, a query for data that a service provider is not authorized to view will be failed with a reason of access denied.
- CMISync is not supported for any scoped/filtered CMIP operation.

The following table shows the CMISE primitive filtering support required of the Local SMS by the NPAC SMS.

Exhibit 12 - CMISE Primitive Filtering Support for Local System Objects

CMISE Primitives	Filter Supported	Notes
M-ACTION	N	No filtering is applied to the actions for the subscriptionVersion object.
M-GET	Y	<p>TN Query with greaterOrEqual and lessOrEqual, and equality must be supported for auditing.</p> <p>The fields used with greaterOrEqual and lessOrEqual filters are subscriptionTN and subscriptionActivationTimeStamp.</p> <p>The field used with equality is subscriptionTN.</p> <p>Filters supported contain either a greaterOrEqual and lessOrEqual filter, or equality filter, for subscriptionTN only or a more complex filter.</p> <p>The more complex filter uses two criteria for filtering. The first criteria used is greaterOrEqual and lessOrEqual filters with subscriptionTN. The second criteria uses greaterOrEqual and lessOrEqual filters for subscriptionActivationTimeStamp. Both criteria must be matched for the data being queried (logical “and”).</p> <p>The scope for the filters is level 1 only with a base managed object class of InpSubscriptions.</p> <p>Number Pool Block Query with greaterOrEqual and lessOrEqual, and equality.</p> <p>The fields used with greaterOrEqual and lessOrEqual filters are numberPoolBlockNPA-NXX-X and numberPoolBlockActivationTimeStamp.</p> <p>The field used with equality is numberPoolBlockNPA-NXX-X.</p> <p>Filters supported contain either a greaterOrEqual and lessOrEqual filter, or equality filter, for numberPoolBlockNPA-NXX-X only or a more complex filter.</p> <p>The more complex filter uses two criteria for filtering. The first criteria used is equality filter with numberPoolBlockNPA-NXX-X. The second criteria uses greaterOrEqual and lessOrEqual filters for numberPoolBlockActivationTimeStamp. Both criteria must be matched for the data being queried (logical “and”).</p> <p>The scope for the filters is level 1 only with a base managed object class of InpSubscriptions.</p>
M-SET	Y	<p>TN Modify with greaterOrEqual and lessOrEqual, and equality must be supported for Mass Update or TN range modify requests.</p> <p>The field used with greaterOrEqual and lessOrEqual filters is subscriptionTN.</p> <p>The fields used with equality are subscriptionTN and subscriptionNewCurrentSP.</p> <p>Filters supported contain either a greaterOrEqual and lessOrEqual filter, or equality filter, for subscriptionTN only, or a more complex filter.</p> <p>The scope for the filters is level 1 only with a base managed object class of InpSubscriptions.</p>

CMISE Primitives	Filter Supported	Notes
		<p>Number Pool Block Modify with greaterOrEqual and lessOrEqual, and equality.</p> <p>The field used with greaterOrEqual and lessOrEqual is numberPoolBlockNPA-NXX-X.</p> <p>The field used with equality is numberPoolBlockNPA-NXX-X.</p> <p>The scope for the filters is level 1 only with a base managed object class of InpSubscriptions.</p>
M-DELETE	Y	<p>TN Delete with greaterOrEqual and lessOrEqual, and equality will be supported.</p> <p>The field used with greaterOrEqual and lessOrEqual filters is subscriptionTN.</p> <p>The field used with equality is subscriptionTN.</p> <p>The scope for the filter is level 1 only with a base managed object class of InpSubscriptions.</p>

4.2.3 Action Scoping and Filtering Support

For messages sent to any object, the scope and filter will be checked to ensure it is appropriate for that object class.

- All M-ACTIONS that relate to subscriptions and number pool blocks are targeted to InpSubscriptions.
- The ONLY filters allowed by the GDMO for InpSubscriptions are "equality" and "present" for the single attribute InpSubscriptionsName.
- If any one of the above M-ACTIONS is sent to a subscriptionVerisonNPAC or numberPoolBlockNPAC object you will get a "no such action" error response from that object.
- If you send a scoped/filtered M-ACTION whose scope includes objects of class subscriptionVersionNPAC or numberPoolBlockNPAC, you will receive an error "no such action" from each object specified by the filter.

4.3 InpLocal-SMS-Name and InpNPAC-SMS-Name Values

The following table (Exhibit 13) shows the values to be used for all currently identified NPAC regions for InpNPAC-SMS-Name in the InpNPAC-SMS object. The InpLocal-SMS-Name for the InpLocal-SMS object will be the service provider ID followed by a dash and the InpNPA-SMS Name (e.g., 9999-Midwest Regional NPAC SMS).

Exhibit 13 - Defined InpLocal-SMS-Name and InpNPAC-SMS-Name Values

NPAC Customer Ids	NPAC SMS Region	InpNPAC-SMS-Name
0000	Midwest	Midwest Regional NPAC SMS
0001	Mid-Atlantic	Mid-Atlantic Regional NPAC SMS
0002	Northeast	Northeast Regional NPAC SMS

NPAC Customer Ids	NPAC SMS Region	InpNPAC-SMS-Name
0003	Southeast	Southeast Regional NPAC SMS
0004	Southwest	Southwest Regional NPAC SMS
0005	Western	West Regional NPAC SMS
0006	West Coast	West Coast Regional NPAC SMS
0007	Canada	Region8 NPAC Canada

4.4 OID Usage Information

4.4.1 OIDs Used for Bind Requests

Value	OID
CMIPUserInfo	2:1:1 (per standards and pp.49 IIS1.5)
CMIPAbortInfo	2:1:1 (per standards and pp.51 IIS1.5)
LnpAccessControl	{Inp-attribute 1} = 1:3:6:1:4:1:103:7:0:0:2:1
UserInfo (NpacAssociationInfo)	1:3:6:1:4:1:103:7:0:0:2:105
Application context	2:9:0:0:2 (per standards)

4.4.2 Other OIDs of Interest

Value	OID
AccessControl OID as part of a SMI notification	1:3:6:1:4:1:103:7:0:0:8:1
AccessControl as part of LNP notifications	{Inp-attribute 1} = 1:3:6:1:4:1:103:7:0:0:2:1

4.5 Naming Attributes

Non-zero values are not supported in the auto-instance naming attributes for Local Number Portability objects defined in the IIS.

4.6 Subscription Version M_DELETE Messages

M_DELETE commands are not sent for subscription versions set to old as a result of subsequent porting activity. M_DELETEs for subscription versions are only sent as a result of disconnect or port to original processing. Local SMS systems are responsible for deletion of the subscription versions in their Local SMS database due to the fact that some LSMS implementations may choose to retain old subscription versions in their database.

4.7 Number Pool Block M_DELETE Messages

M_DELETE commands are not sent for number pool blocks set to old as a result of subsequent porting activity. M_DELETEs for number pool blocks are only sent as a result of de-pool. Local SMS systems are responsible for deletion of the number pool blocks in their Local SMS database

due to the fact that some LSMS implementations may choose to retain old number pool blocks in their database.

4.8 Subscription Version Queries

For Service Providers that support the enhanced SV Query functionality (Service Provider SV Query Indicator tunable parameter set to TRUE), the behavior is defined in this section.

If a subscription version query is requested by the SOA/LSMS, and the results are larger than the Maximum Subscription Query tunable value, the NPAC SMS will return subscription versions up to that max value. The SOA/LSMS would accept this message, then use it's contents to send another query to the NPAC SMS, starting with the next TN, and so on until all SVs are returned to the SOA/LSMS. It will be up to the SOA/LSMS to manage the data returned from the NPAC SMS and determine the next request to send to the NPAC SMS in order to get the next set of subscription versions.

The NPAC SMS will continue to return subscription versions that meet the selection criteria. However, the NPAC SMS will not return a "count" to the SOA/LSMS for number of records that match the selection criteria. Service providers should modify their systems to support the following subscription version query operations to the NPAC SMS:

1. When data is returned from a subscription version query and there are exactly n (tunable) records returned, the SP must assume that they didn't get all the data from their query.
2. After processing the first n records, they should send a new query that picks up where the data from the prior query ended.
3. The subscription version data returned from the NPAC SMS for subscription version queries will be sorted by TN and then by subscription version ID so a filter can be created to pick up where the prior query ended.
4. For example, if a SOA query to the NPAC SMS returns exactly 150 records and the last subscription version returned was TN '303-555-0150' with subscription version ID of 1234. The filter used on the next query would be: All subscription versions where ((TN >= 303-555-0151) OR (TN = 303-555-0150 AND subscription version ID >= 1235)).The NPAC SMS does support OR filters.
5. Once the results from the NPAC SMS returns less than 150 records, the SP can assume they received all records in the requested query.

Note: In this situation the NPAC SMS follows the linked replies for the subscription query results with an empty reply (this is an indication that the NPAC SMS is finished sending data for this request).

As an example, a Service Provider's SOA sends a Subscription Version query to the NPAC SMS, There are 225 Subscription Versions that meet the selection criteria. Assuming the Maximum Subscription Query tunable value is set to 150 Subscription Versions, the SOA would receive data from the NPAC SMS in the form of 150 Subscription Versions in 150 linked replies (1 SV per linked reply) followed by a reply (for a total of 151 linked replies). The SOA would then send another query based on the algorithm described above. The SOA would then receive data from the NPAC SMS in the form of 75 Subscription Versions in 75 linked replies (1 SV per linked reply) followed by a reply (for a total of 76 linked replies).

Note: In this situation the NPAC SMS follows the linked replies for the subscription query results with an empty reply (this is an indication that the NPAC SMS is finished sending data for this request).

For Service Providers that DO NOT support the enhanced SV Query functionality (Service Provider SV Query Indicator tunable parameter set to FALSE), a complexityLimitation error is

returned when the number of SVs in a query response exceed the Maximum Subscription Query tunable value.

4.9 NPAC Rules for Handling of Optional Data Fields:

Information is provided on how the NPAC handles the XML string as well as how providers system should deal with Activate and Modify downloads that contain XML optionalData strings. Disconnects are not covered here because they don't contain XML strings.

- Activate - String contains only those fields supported by the provider and specified in the create request.
 - Provider systems should store the fields specified in the message.
- Modify - String contains only those fields supported by the provider and were modified in the modify request.
 - If the modify removed a value from an optional field, it is included in the string with a value of nil.
 - Provider systems should modify only the fields specified in the message. Any other optional fields should be retained.
- Audit - String is included only if there was at least one discrepancy in the fields supported by the provider.
 - Only the OptionalData attribute/parameters supported by an LSMS are audited.
 - Only the OptionalData attribute/parameters supported by the auditing SOA are returned to the SOA in the discrepancy notifications.
 - Audit discrepancy reports contain well formed XML strings (i.e., parse-able) representing the discrepant fields. Fields that are not discrepant will not be included. The SOA needs to parse the XML strings to be able to act on the discrepancies.
 - For Modify downloads that result from an Audit:
 - String contains all fields supported by the provider, regardless of whether or not that individual field was discrepant, and regardless of whether or not the NPAC's subscription version has values for those fields.
 - Fields not supported by the provider are omitted even if they were returned in the Audit query response from the LSMS.
 - Fields supported by the provider but not present in the NPAC's subscription version are included with a value of nil.
 - Provider systems should store the fields as specified above for Activate or Modify downloads.
- Time Based Recovery – Same as Activate.
 - Provider systems should replace all fields with those in the recovery message, including removal of optional fields not provided in the recovery message.
- SWIM Recovery – Individual operations are recovered.
 - Provider systems should store the fields as specified in the message. For both Activate and Modify operations, all attributes in the object (including supported optional data fields that are populated) will be sent to accommodate objection creation in provider systems. If no supported optional data fields are populated,

the Optional Field string is omitted entirely. If a Modify operation removed a value from an optional field, it is included in the string with a value of nil.

- Notifications –
 - For a create notification (Number Pool Block only), string contains only fields supported by the provider and specified in the create request.
 - For an AVC (Number Pool Block only), string contains only those fields supported by the provider that were modified. If a supported field is removed, it is included in the string with a value of nil.
- BDD - Each field supported by the provider has a position in the BDD record.
 - For fields supported by the provider but not present in the NPAC's subscription version, the field is included in the string with an empty value (two adjacent pipe characters).
 - For fields not supported by the provider, no field placeholder is included in the string (no adjacent pipe characters).
 - Provider systems should replace all fields with those in the BDD.

5 *Secure Association Establishment*

5

5.1 Overview

This section describes the security, the association management and recovery procedures for the service provider SOAs and Local SMSs to follow, and how error information will be passed between interfaces.

The first section describes the security and authentication procedures used in the NPAC SMS interface. The second section describes the NPAC SMS's behavior and error handling and suggests how a service provider SOA or Local SMS should proceed when establishing an association.

5.2 Security

This section describes the security processes and procedures necessary for service provider SOA systems and Local SMSs to establish a secure association and maintain secure communication with the NPAC SMS. Security threats to the NPAC SMS include:

- Spoofing - An intruder may masquerade as either the SOA, Local SMS, or NPAC SMS to falsely report information.
- Message Tampering - An intruder may modify, delete, or create messages passed.
- Denial or Disruption of Service - An intruder may cause denial or disruption of service by generating or modifying messages.
- Diversion of Resources - An intruder may generate or modify messages that cause resources to be diverted to unnecessary tasks.
- Slamming - An intruder may generate or modify messages that cause customer's service to be moved between service providers.

Security threats are prevented in the NPAC SMS by use of the following methods:

- Strong two way authentication at association.
- Insuring data integrity by detection of replay, deletion, or modification to a message.
- Insuring non-repudiation of data by guaranteeing integrity and supporting data origination authentication for each incoming message.
- Implementation of access control and application level security that allows only authorized parties to cause changes to the NPAC SMS database.

5.2.1 Authentication and Access Control Information

The following access control information definition will be used in the AccessControl field of the association and CMIP PDUs to ensure a secure communication for both the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface:

```

LnpAccessControl ::= SEQUENCE {
    systemId          [0]  SystemID,
    systemType       [1]  SystemType,
    userId           [2]  GraphicString60 OPTIONAL,
    listId           [3]  INTEGER,
    keyId            [4]  INTEGER,
    cmipDepartureTime [5]  GeneralizedTime,
    sequenceNumber   [6]  INTEGER (0...4294967295),
    function         [7]  AssociationFunction,
    recoveryMode     [8]  BOOLEAN signature
    signature        [9]  BIT STRING
}

ServiceProvId ::= GraphicFixedString4

SystemID ::= CHOICE {
    serviceProvID [0] ServiceProvId,
    npac-sms [1] GraphicString60
}

SystemType ::= ENUMERATED {
    soa(0),
    local-sms(1),
    soa-and-local-sms(2), -- value not supported

    npac-sms(3)          --value is only valid for AccessControl
                        definition
}

AssociationFunction ::= SEQUENCE {
    soaUnits [0] SoaUnits,
    lsmsUnits [1] LSMSUnits
}

SoaUnits ::= SEQUENCE {
    soaMgmt [0] NULL OPTIONAL,
    networkDataMgmt [1] NULL OPTIONAL,
    dataDownload [2] NULL OPTIONAL
    notificationDownload [3] NULL OPTIONAL
}

LSMSUnits ::= SEQUENCE {
    dataDownload [0] NULL OPTIONAL,
    networkDataMgmt [1] NULL OPTIONAL,
    query [2] NULL OPTIONAL
}
    
```

Exhibit 4. Access Control

5.2.1.1 System Id

The system Id is the unique Id for the system using an interoperable interface and must be specified in the systemId field. For a service provider using the SOA and/or Local SMS interfaces, this is the Service Provider ID. For the NPAC SMS, it is the unique identifier for the regional SMS.

In cases where a service provider is providing SOA services for an associated service provider, the primary service provider must establish the association with their System Id set to their primary Service Provider ID. PDUs that are subsequently sent to the NPAC SMS may contain the primary or associated Service Provider Ids of the requesting service provider. Associated Service Provider Ids are sent in the System Id when actions are being taken on behalf of an associated service provider by the service provider providing SOA services (the primary service provider). The Service Provider ID specified in the access control for PDUs sent after association establishment, whether it's the primary or secondary Service Provider ID, is considered the requesting service provider and all validations will use this Service Provider ID.

5.2.1.2 System Type

The system type that indicates the type of system using the interoperable interface must be specified in the systemType field. The valid types are SOA and/or Local SMS and NPAC SMS.

5.2.1.3 User Id

The user Id of the user of the interface can optionally be specified in the userId field for the SOA interface. This is the 60 character graphics string user identifier for a user on a SOA system. It is not validated on the NPAC SMS, however, it is used for logging purposes.

5.2.1.4 List Id

The list Id must be specified as an integer in the listId field to identify a key list. This key list is one of the key lists exchanged outside of the interface process that is known to both the NPAC SMS and the Local SMS or SOA system it is communicating with.

NPAC key lists and service provider key lists are to be managed based upon service provider id and presentations layer address (P-selector) of the service provider's SOA system and/or Local SMS system. Also, a given service provider id and P-selector value can exist for one or more Network Service Access Points (NSAP).

The NPAC SMS must generate and maintain NPAC key lists based upon the service provider's service provider id and P-selector value of the system(s) that support its SOA and LSMS interfaces. In addition, service providers(SOA systems and Local SMS systems) must also manage the NPAC's key lists. Each side of the interface must support multiple NPAC key lists per service provider id and P-selector value.

Service providers (SOA system and Local SMS system) must generate and maintain key lists based upon the service provider's service provider id and P-selector value of the system(s) that support its SOA and LSMS interfaces. Furthermore, the NPAC SMS must also manage the service provider's key lists. Each side of the interface must support multiple service provider(SOA system and Local SMS) key lists per service provider id and P-selector value.

In cases where a service provider is providing SOA services for an associated service provider, key lists are only exchanged with the primary service provider using the primary service provider id.

5.2.1.5 Key Id

The key Id of a key in the key list must be specified as an integer in the keyId field. This uniquely identifies the key in the key list used to create the digital signature. The size of the modulus for the key is variable between 600 and 2048 bits.

Since key lists are to be managed based upon service provider id and the P-selector value of a service provider's SOA system and/or Local SMS system, keys are to be treated independently at the presentation layer for an association. By using presentation layer support of a key list, SOA and Local SMS systems can have one key or unique keys to support the SOA and LSMS interfaces. The following situations are supported:

1. If a service provider has one process supporting the SOA and LSMS interface, then the process has one P-selector value supporting both interfaces. The SOA/Local SMS system would use the same key list and the same key for all associations created for the both the SOA and LSMS interface. The NPAC SMS would in turn have one NPAC key list and key to support both interfaces.
2. If a service provider has two processes supporting the SOA and LSMS interface, then each process would have different P-selector values. The SOA and Local SMS systems would use separate key lists and keys per interface. In detail, the SOA system would use a key list and key for all associations involving the SOA interface and the Local SMS system would use a different key list and key for all associations involving the LSMS interface. The NPAC SMS would also manage separate key lists and keys per the SOA and LSMS interface. Furthermore, the NPAC SMS would use the same key list and key for all associations within a given interface.
3. If a service provider has an SOA system or a Local SMS system that consists of multiple processes, then each processes would have different P-selector values. Therefore, each process would manage separate key lists and separate keys per process. The NPAC SMS would also manage separate key lists/keys per process. For example, if a Local SMS system consists of 2 processes (one process supporting subscription data and the other supporting network/query data), the processes would have separate P-selector values and use separate key lists/keys per association. The NPAC SMS would also manage separate key lists and keys per process within the LSMS interface.

Note: In cases where a service provider is providing SOA services for an associated service provider, keys are used from primary service provider key lists

If the service provider determines their key is compromised they should change their own private key and list. If the NPAC determines that their key is compromised then they should change their own private key and list. The NPAC should not invalidate a service provider's key and vice versa. However, should either side of the industry interfaces (SOA and Local SMS interface) change keys, the remote side is expected to mark the previously used key as

used (key expiration). Previously used keys (ListId/KeyId combinations) are considered expired and result in a security violation across the industry interface when re-used.

5.2.1.6 CMIP Departure Time

The CMIP departure time must be specified in GeneralizedTime in the cmipDepartureTime field as the time the PDU departed the sending system. The universal time format (YYYYMMDDHHMMSS.0Z) is used. In order to ensure data integrity and no-repudiation the NPAC SMS system must be synchronized to within five minutes of the Local SMS and SOA systems that it communicates.

5.2.1.7 Sequence Number

The sequence number is an integer that must be specified in the sequenceNumber field. It should be specified as zero at association time and incremented by one for every message sent over the association. Once the sequence number reaches 4294967295 the counter will be reset to one for the association. Please note that each sender independently keeps its own counter for the sequence number of messages sent and received. For example, after association is established, a Local SMS could send three messages to the NPAC SMS with sequence numbers 1, 2, and 3 respectively. The NPAC SMS when sending its first message to the Local SMS would use sequence number 1, not sequence number 4.

5.2.1.8 Association Functions

The Association Function(s) must be specified on the initial association request (AARQ PDU). The following table lists the possible Association Functions that can be specified for each of the Association Request Initiators and the associated bit mask value:

Exhibit 135 Association Functions

Association Request Initiator	SOA	Local SMS
Association Function SOA Management (Audit and Subscription Version) Classes: InpSubscriptions numberPoolBlock numberPoolBlockNPAC subscriptionAudit subscriptionVersion subscriptionVersionNPAC	0x01	

Service Provider and Network Data Management Classes: InpNetwork InpNPAC-SMS InpServiceProvs IsmsFilterNPA-NXX serviceProv serviceProvLRN serviceProvNetwork serviceProv-NPA-NXX serviceProvNPA-NXX-X	0x02	0x04
LSMS Network and Subscription Data Download Classes: InpNetwork InpSubscriptions		0x08
SOA Network Data Download Classes: LnpNetwork	0x20	
Query Outbound from the NPAC SMS Classes: All		0x10
SOA Notifications (only applicable for SOAs supporting a separate notification association) Classes: InpNPAC-SMS InpSubscriptions numberPoolBlockNPAC subscriptionAudit subscriptionVersionNPAC	0x40	

The association functions specified upon association are stored. Then all subsequent operations performed by that associations are then validated against that data to verify that they are 'legal'. All outbound messages from the NPAC are also validated against the association functions and if a service provider does not have the correct masking set, they will not receive the transmission. Note that the multiple Association Functions can be specified for an association. For example, a Local SMS can establish an association for both the process audit and network and subscription data download association functions.

SOA Notifications have been separated out to support SOAs that wish to implement a separate SOA Channel for Notifications. Based on the Service Provider tunable (SOA Notification Channel Service Provider Tunable), this function may be included in a SOA association, even if the Service Provider does not bind with that function mask. This allows SOA notifications to be sent down a single SOA channel.

5.2.1.9 Recovery Mode

The recovery mode flag is set to TRUE when a Local SMS or SOA is establishing a connection after a downtime. This flag indicates to the NPAC SMS to hold all current transactions until the Local SMS or SOA sends the Recovery Complete action. Once an association is established in recovery mode by a Local SMS, the Local SMS should request service provider, subscription and network downloads and notifications that occurred during downtime. Once an association is established in recovery mode by a SOA, the SOA should request service provider and network downloads and notifications that occurred during downtime. After these steps are complete, the Local SMS or SOA should submit the Recovery Complete action. The NPAC SMS will respond to the recovery complete action, send all updates that occurred since association establishment and then normal processing will resume. See *Appendix B, Section 7.1*.

Service Provider Local SMS and SOA systems recover data independently. SOA systems can recover their information before, after, or concurrently with an LSMS using the same Service Provider Id.

A service provider providing SOA services for associated service providers can recover notifications for the primary and each associated service provider id prior to issuing the Recovery Complete action.

Alternatively, Service Provider Local SMS and SOA systems can recover data using the SWIM method. Refer to section 5.3.4 (Recovery) for more information.

5.2.1.10 Signature

The signature field contains the MD5 hashed and encrypted systemId, the system type, the userId, the cmipDepartureTime, and sequenceNumber without separators between those fields or other additional characters. Before hashing and encryptions, character fields are ASCII format and integer fields are 32 bit big endian. Encryption is done using RSA encryption using the key from the key list specified. Validation of this field ensures data integrity and non-repudiation of data. The following is additional information about how the information should be represented for digital signature encoding:

Field	Format	Contents
SystemID	ASCII	
SystemType	Integer	e.g. local-sms = 1
UserId	ASCII	
cmipDepartureTime	ASCII	"YYYYMMDDHHMMSS.OZ" format
sequenceNumber	Integer	

5.2.2 Association Establishment

Strong two way authentication at association is done for both the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface. This secure association establishment is done at the application level using the access control field described

above. The access control information used during association set-up is sent in the association control messages. Association establishment can be done by the SOA to NPAC SMS or Local SMS to NPAC SMS. The NPAC SMS cannot initiate an association. The initiator of the association specifies its information in the AARQ PDU message and the responder in the AARE PDU.

When the SOA or LSMS initiate an association with the NPAC the NSAP and P-selector values will be validated to ensure that they are valid for the service provider initiating the association. The following is an example of the information exchanged in the AARQ and AARE PDUs and the processing involved. Assume for the example:

- A Local SMS is making an association with the NPAC SMS.
- The Local SMS systemId is “9999.”
- The NPAC SMS systemId is “NPAC SMS User Id.”
- The listId for the key list is 1.
- The keyId is 32.
- The key in listId 1 with a keyId of 32 is “ABC123.”
- The sequence number is 0 (as required).

The Local SMS initiates the association request by creating and sending an AARQ PDU to the NPAC SMS. This AARQ PDU contains the following access control information in the syntax described above:

- The systemId of “9999”.
- The listId of 1.
- The keyId of 32.
- The current Local SMS GMT time in the cmipDepartureTime.
- A sequence number of 0.
- The signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key “ABC123” as found in key list 1 with key id 32.
- And all BOOLEAN items are set to FALSE in the functional groups field, except for the LSMSUnit of Query item which is set to TRUE.

Once the AARQ PDU is sent, the sender (in this case the Local SMS), starts a tunable timer (with a default value of 2 minutes). If the timer expires before the AARE PDU is received then the Local SMS will terminate the association attempt.

When the NPAC SMS receives the association request it validates the data received. The data is validated as follows:

- Ensure the systemId is present and valid for the association.
- Ensure the sequence number is 0.
- Ensure the cmipDepartureTime is within 5 minutes of the current NPAC SMS GMT time.
- Find the key specified and decrypt the signature insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.

- The functional groups requested are valid for the system type that requested the association. In this example, the system type must be “local-sms(1)” {“soa-andlocal-sms(2)” value is to be removed from a future version of the IIS}.

If validation of the AARQ PDU fails then an A-ABORT will be issued by the NPAC SMS with an error of access denied. If the validation of the AARQ PDU is successful then an AARE PDU would be sent back to the Local SMS. This AARE PDU contains the following access control information in the syntax described above:

- The systemId of “NPAC SMS User Id.”
- The listId of 1.
- The keyId of 32.
- The current NPAC SMS GMT time in the cmipDepartureTime.
- A sequence number of 0.
- And the signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key “ABC123” as found in key list 1 with key id 32.

The NPAC SMS may choose to optionally specify a new listId and keyId if for any reason it wants to make a key change. Should either side of the interface change its listId/keyId values, both sides of the interface must mark the previously used keyId as used.

When the Local SMS receives the association response it validates the data received. The data is validated as follows:

- Ensure the systemId is present and valid for the association. (Note: the userId field is not required for Local SMS and NPAC SMS associations).
- Ensure the sequence number is 0.
- Ensure the cmipDepartureTime is within 5 minutes of the current Local SMS GMT time.
- Find the key specified and decrypt the signature insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.

If validation of the AARE PDU fails then an A-ABORT will be issued by the Local SMS. If validation is successful then a secure association has been established.

5.2.3 Data Origination Authentication

For M-GET, M-SET, M-CREATE, M-DELETE, and M-ACTION, the access control field described above is used for data origination authentication. Please note that any of the messages sent between manager and agent must be sent in confirmed mode. The following is an example of the information exchanged in the CMIP PDUs and the processing involved. Assume for the example:

- A SOA is making an association with the NPAC SMS.
- The SOA system provides SOA functionality for another Service Provider.
- The SOA systemId is “9999” for the primary Service Provider Id and is “8888” for an associated Service Provider Id.
- The NPAC SMS systemId is “NPAC SMS User Id.”
- The listId for the key list is 1.

- The keyId is 32.
- The key in listId 1 with a keyId of 32 is “ABC123.”
- The sequence number is 1.

The SOA sends an M-GET to the NPAC SMS. The M-GET PDU contains the following access control information in the syntax described above:

- The systemId of “8888.”
- The listId of 1.
- The keyId of 32.
- The current Local SMS GMT time in the cmipDepartureTime.
- A sequence number of 1.
- And the signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key “ABC123” as found in key list 1 with key Id 32.

Once the M-GET is sent, the sender (in this case the SOA), starts a tunable timer (with a default value of 2 minutes). If the timer expires before the M-GET CMISE service response is received then the SOA will regenerate the sequenceNumber, cmipDepartureTime and signature and resend the request. The SOA should resend a default of 3 times and abort the association if no response is received. If a response is received after the timeout period, it should be discarded. If an error message is received on a retry request, it should be evaluated to see if the request was processed or the error was received for other reasons. For example, an error of “duplicateObjectInstance” for an M-CREATE request most likely indicates a successful create.

When the NPAC SMS receives the M-GET request it validates the data received. The data is validated as follows:

- Ensure the systemId is present and valid for the association. For the SOA the systemId can be the primary or associated Service Provider Id depending on the requestor.
- Ensure the sequence number is the next sequence number expected. (In this case 1).
- Ensure the cmipDepartureTime is within 5 minutes of the current NPAC SMS time.
- Find the key specified and decrypt the signature, insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.

If validation of the M-GET PDU fails then an A-ABORT will be issued by the NPAC SMS without any additional information to prevent tampering and unauthorized use of network resources by intruders. If the validation of the M-GET PDU is successful then the NPAC SMS would get the data requested and send back an M-GET Response to the SOA.

Since CMIP notifications (M-EVENT-REPORT) do not have access control fields, all notifications defined contain the access control information in the notification definition. ObjectCreation, ObjectDeletion, and AttributeValueChange should use the “information” attribute, which is an ANY DEFINED BY to contain the access control field. The values and authentication for the notification access control fields are the same as above. For range ObjectCreation and AttributeValueChange notifications the access control would not be placed in the information attribute but rather in the access control attribute defined.

This would allow for the access control information to only be present once in the range notifications.

When the NPAC sends a notification, the destination service provider is uniquely identified in the distinguishedName of the M-EVENT-REPORT. The InpLocalSMS-Name attribute value(2.17) is appended to the service provider's id and is used to populate the value of the first element of the EventReportArgument's managedObjectInstance distinguishedName. This allows primary service providers to distinguish notifications destined for themselves and for each secondary service provider.

5.2.4 Audit Trail

Audit trails will be maintained in logs on the NPAC SMS for the following association information:

- Association set-up messages.
- Association termination messages.
- Invalid messages:
 - Invalid digital signature.
 - Sequence number out of order.
 - Generalized time out of range.
 - Invalid origination address.
- All incoming messages regardless of whether or not they cause changes to data stored in the NPAC SMS.

This information will be made available for report generation on the NPAC SMS system. It will not be made available through the NPAC SMS Interoperable Interface.

5.3 Association Management and Recovery

5.3.1 Establishing Associations

5.3.1.1 NpacAssociationUserInfo

The following structure will be used to report the status of a login attempt or the current state of the NPAC SMS:

```
NpacAssociationUserInfo ::= SEQUENCE {
    error-code [0] IMPLICIT ErrorCode,
    error-text [1] IMPLICIT GraphicString(SIZE(1..80))
}
```

```
ErrorCode ::= ENUMERATED
{
    success (0),
    access-denied (1)
    retry-same-host (2)
    try-other-host (3)
}
```

}

Bind Requests and Responses

For AARQ (M-Bind requests) the NPAC SMS will be ignoring the CMIPUserInfo userInfo field. The SMASEUserInfo will be ignored by the NPAC SMS.

In order to validate a successful login, the AARE (M-Bind response) from the NPAC SMS will contain the NpacAssociationUserInfo as the “userInfo” field of the CMIPUserInfo that is contained on the AARE. The ErrorCode will be set to “success”.

The following structure will be used for CMIPUserInfo:

```
CMIPUserInfo ::= 2:9:1:1:4
--{joint-iso-ccitt(2) ms(9) cmip(1) cmip-pci(1)
abstractSyntax(4)}

CMIPUserInfo ::= SEQUENCE {
    protocolVersion [0] IMPLICIT ProtocolVersion
    DEFAULT {version1-cmip-assoc},
    functionalUnits [1] IMPLICIT FunctionalUnits DEFAULT {},
    accessControl [2] EXTERNAL OPTIONAL
    userInfo [3] EXTERNAL OPTIONAL
}
```

5.3.1.2 Unbind Requests and Responses

The NPAC SMS will never be issuing the RLRQ (M-Unbind request), but will respond to them from the SOA or Local SMS.

5.3.1.3 Aborts

For unsuccessful logon attempts or situations where the NPAC SMS application must abort all associations, the ABRT CMIPAbortInfo structure’s “userInfo” will contain the NpacAssociationUserInfo structure. The ErrorCode will be set to one of the enumeration values.

The following structure will be used for CMIPAbortInfo:

```
CMIPAbortInfo ::= 2:9:1:1:4
--{joint-iso-ccitt(2) ms(9) cmip(1) cmip-pci(1)
abstractSyntax(4)}

CMIPAbortInfo ::= SEQUENCE {
    abortSource [0] IMPLICIT CMIPAbortSource,
    userInfo [1] EXTERNAL OPTIONAL
}
```

5.3.1.4 NPAC SMS Failover Behavior

Under normal conditions, the primary NPAC SMS will be responding by accepting association requests while the secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of TRY_OTHER_HOST.

When the primary NPAC SMS needs to go down for a short period of time (secondary will not take over), the primary NPAC SMS will either not be responding (if down) or be denying association requests with an error code of RETRY_SAME_HOST (if partially up). The secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of TRY_OTHER_HOST.

When the primary NPAC SMS goes down (scheduled or unscheduled) and the secondary NPAC SMS is re-synchronizing to become active, the primary NPAC SMS will be denying association requests with an ABRT and error code of TRY_OTHER_HOST. The secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of RETRY_SAME_HOST. Once the secondary NPAC SMS is done re-synchronizing, it will then start accepting association requests.

5.3.1.5 Service Provider SOA and Local SMS Procedures

The following is an algorithm that can be used by a service provider SOA or Local SMS when trying to establish an association with the NPAC SMS:

try to establish an association on the primary NPAC SMS if a response was obtained

```

{
  if the response was an ABRT and the ABRT is from the NPAC
  Application
  {
    switch (error code)
    {
      case ACCESS_DENIED
        find out what is causing the error and fix it
        retry the association on the primary NPAC SMS
      case RETRY_SAME_HOST
        wait X seconds
        retry the association on the primary NPAC SMS
      case TRY_OTHER_HOST
        wait X seconds
        execute this algorithm again substituting
        "secondary" for "primary"
    }
  }
  else
  {

```

```

        if the response was an ABRT and from the PROVIDER
        (not application)
            find out what is causing the error and fix it
            retry the association on either the primary or
            secondary NPAC SMS
        }
else
{
    # timeout - some type of network error has occurred
    # a number of different things can be done:
    #
    #   wait X seconds
    #   retry primary
    #
    #       or
    #
    #   find out what is causing the error and fix it
    #   retry the association on the primary NPAC SMS
    #
    #       or
    #
    #   wait X seconds
    #   execute this algorithm again substituting
    #   "secondary" for "primary"
}
    
```

5.3.2 Releasing or Aborting Associations

Any of the systems, NPAC SMS, service provider SOA or Local SMS can abort an association at any time. Only the SOA and Local SMS can perform an RLRQ request. Once a scheduled outage has arrived, the NPAC SMS will abort associations (error code of “Try Other Host” or “Retry Same Host” depending on the type of outage).

5.3.3 Error Handling

5.3.3.1 NPAC SMS Error Handling

The NPAC SMS will issue errors to the Local SMS and SOA interfaces based upon the definitions and mappings in Appendix A. The NPAC SMS expects the SOA and Local SMS to support the same error definitions when both issuing (with the exception of a sending processingFailure as defined in ILL 130) and receiving error responses for the operations each interface supports.

The NPAC SMS will attempt to interpret an error returned from a SOA or Local SMS. The NPAC SMS will log the error. If the request is not resent and the error response was returned from a Local SMS and related to a subscription

version broadcast (M-CREATE or Create Action, M-DELETE, M-SET), a broadcast failure will be noted for the service provider on the subscription version. If a service provider does not have an active Local SMS association at the time of a broadcast, the broadcast will be automatically failed for the service provider.

The Local SMS and SOA are expected to recover themselves with the NPAC SMS when their association is reestablished. Thus it is the responsibility of the Local SMS and SOA to request the necessary data to rectify the failed transmission of M-EVENT-REPORTs, network data updates and non-broadcast oriented subscription version updates.

If the NPAC SMS sends a request to a Local SMS or SOA and receives no response from the CMISE service within the tunable period, the NPAC SMS will resend the message according to the tunable retry periods for the specific message type. If a response is received after the timeout period, it will be discarded. If the NPAC SMS receives no response, the NPAC SMS will assume the association is down and abort the connection. The Local SMS and SOA systems should assume the same behavior with the NPAC SMS.

5.3.3.2 Processing Failure Error

The NPAC SMS will use the Service Provider profile flags (SOA Action Application Level Errors Indicator, SOA Non-Action Application Level Errors Indicator, LSMS Action Application Level Errors Indicator, and LSMS Non-Action Application Level Errors Indicator) to determine the handling of Processing Failure errors. When they are not supported:

In addition to the standard CMIP error reporting mechanisms, the following attribute will be passed in the SpecificErrorInfo structure on CMIP errors that return a PROCESSING FAILURE error. This structure will be used to detail errors not covered by the standard CMIP error codes.

GDMO Definition

```
lnpSpecificInfo ATTRIBUTE
    WITH ATTRIBUTE SYNTAX LNP-ASN1.LnpSpecificInfo;
    MATCHES FOR EQUALITY;
    BEHAVIOUR lnpSpecificInfoBehavior;
    REGISTERED AS {lnp-attribute 8};
```

```
lnpSpecificInfoBehavior BEHAVIOUR
```

```
DEFINED AS !
```

```
This attribute is used to return more detailed
error text information upon a CMIP Processing
Failure error.
```

```
!;
```

ASN.1 Definition

```
LnpSpecificInfo ::= GraphicString(SIZE(1..256))
```

When the Service Provider profile flags (SOA Application Level M-ACTION Errors Indicator, SOA Non-Action Application Level Errors Indicator, LSMS Application Level Errors M-ACTION Indicator, and LSMS Non-Action Application Level Errors) are

supported, the Processing Failure error will contain an `InpSpecificErrorCode` instead of `InpSpecificInfo`.

5.3.3.3 NPAC SMS Detailed Error Codes

The NPAC SMS will issue detailed error codes to the supporting SOA and Local SMS interfaces based upon the definitions and mappings in Appendix A. The Service Provider profile flags (SOA Application Level Errors Indicator, LSMS Application Level Errors Indicator) will indicate whether application level errors are supported across the SOA/LSMS interfaces. When they are supported:

- The SOA/LSMS will utilize ACTIONs that support detailed error codes (e.g., M-ACTION `subscriptionVersionActivateWithErrorCode`), as defined in Exhibit 10. The SOA/LSMS may still utilize ACTIONs that do not support detailed error codes.
- All other CMIP messages (e.g., M-CREATE `serviceProvNPA-NXX`) will be supported through a `processingFailure` response that will contain the detailed error code, instead of the other CMIP standard errors.

This allows all messages to be covered for the detailed error codes for SOA/LSMS interfaces that support this features.

For SOA/LSMS interfaces that do not support this feature, an ACTION that supports error codes will result in a no-such-action error response.

5.3.4 Recovery

The SOA and Local SMS associations are viewed to be permanent connections by the NPAC SMS. Thus when the association is broken for any reason, the system connecting to the NPAC SMS must assume responsibility to recover and resynchronize themselves with the NPAC SMS. One association should be established for recovery and no other associations should be established in normal mode until recovery is complete.

During the recovery processing, other messages may be generated at the NPAC SMS that are intended for the recovering SOA or LSMS. These messages are queued on the NPAC SMS until the SOA or LSMS finishes the recovery process and sends an `InpRecoveryComplete` action to the NPAC SMS. Additionally, during the recovery process, the “x by y” retry functionality (where “x” is the number of attempts, and “y” is the interval in number of minutes in between attempts) continues on the NPAC SMS, but message sending is suspended to the SOA or LSMS, and the retry attempts counter is not decremented, as long as the SOA or LSMS is still in recovery mode. Therefore, a Subscription Version could stay in a “sending” status for a period of time longer than expected, since the retry logic will not transition the status to “partial failure” or “failed” as long as a Service Provider is in recovery mode.

While recovering subscription data, the NPAC SMS excludes Subscription Versions with a status of failed. The value in the Broadcast Timestamp field in each Subscription Version is used to determine whether or not a Subscription Version is included in the recovering LSMS’s requested criteria.

The SOA or LSMS is capable of recovering data based on the association functions. The SOA recovers service provider, network data and notification data using the network data management association function (`networkDataMgmt`). The LSMS recovers notifications and subscription data using the data download association function (`dataDownload`), and

recovers service provider and network data using the network data management association function (networkDataMgmt).

Service Provider and Notification recovery requests can only be sent to the NPAC when the SOA/LSMS is in recovery mode, otherwise an abort message is returned.

NPAC data may be recovered in three ways, ‘time-based’, ‘record-based’, or ‘Send What I Missed (SWIM)-based’ criteria. Based on the type of data being recovered, additional criteria may also be specified. The table below show the type of data that can be recovered, and the criteria that may be used for each type.

Data Type	Criteria	Additional Criteria
Network Data	Time Based	Time Range (consisting of Start time, End time)
	Record-Based	NPA-NXX range all NPA-NXX data NPA-NXX-X range all NPA-NXX-X data LRN range all LRN data all network data
	SWIM	conditional Action ID (indicating receipt of previous response for <Network> data)
Service Provider Data	Time Based	Time Range (consisting of Start time, End time)
	Record Based	Service Provider ID
		All Service Providers
SWIM	conditional Action ID (indicating receipt of previous response for <Service Provider> data)	
Subscription Data	Time-Based	Time Range (consisting of Start time, End time)
	Record-Based	TN TN range
	SWIM	conditional Action ID (indicating receipt of previous response for <Subscription> data)
Number Pool Block Data	Time-Based	Time Range (consisting of Start time, End time)
	Record-Based	NPA-NXX-X NPA-NXX-X range

	SWIM	conditional Action ID (indicating receipt of previous response for <Number Pool Block> data)
Notification Data	Time-Based	Time Range (consisting of Start time, End time)
	Record-Based	Not Available
	SWIM	Time Range (consisting of Start time, End Time) is ignored in a SWIM recovery request Conditional Action ID (indicating receipt of previous response for <Notification> data)

‘Time-Based’ Recovery Requests

All 'time-based' recovery requests specifying time range criteria are limited to the NPAC SMS tunable, “Maximum Download Duration”. When the SOA or LSMS issues a recovery request (whether Service Provider, Network, Subscription, Number Pool Block, or Notification Data) with time-based criteria, the NPAC SMS will compare the time range indicated in the request to the “Maximum Download Duration” tunable.

For service providers that do not support linked replies, Subscription data 'time-based' recovery requests specifying time range criteria are also limited to the number of TNs specified in the Service Provider specific tunable, “Maximum TN Download in Recovery Request”. Therefore, a valid request will fall within both the duration and quantity tunable values.

For service providers that do not support linked replies, Notification data 'time-based' recovery requests specifying time range criteria are also limited to the number of notifications specified in the NPAC SMS tunable, “Maximum Number of Download Notifications”. Therefore, a valid request will fall within both the duration and quantity tunable values.

For service providers that do not support linked replies, for all types of 'time-based' recovery requests, where the tunable value is exceeded, an appropriate error message is issued over the interface from the NPAC SMS to the originating system. This applies to both duration overages (“Maximum Download Duration”), and number of record overages (“Maximum TN Download in Recovery Request” for subscription data, and “Maximum Number of Download Notifications” for notification data).

‘Record-Based’ Recovery Requests

For service providers that do not support linked replies, all ‘record-based’ recovery requests specifying other criteria (for example, TN/TN range, NPA-NXX/NPA-NXX range) are limited by the number of records specified in the NPAC SMS tunable, “Maximum Number of Download Records”. When the SOA or LSMS issues a network data recovery request or the LSMS issues a subscription version data recovery request, using 'record-based' criteria, the NPAC SMS will compare the records indicated in the request to the “Maximum Number of Download Records” tunable. If the number of records exceeds this tunable value, an appropriate error message is issued over the interface from the NPAC SMS to the originating system.

‘SWIM-Based’ Recovery Requests

‘SWIM-based’ recovery requests allow for the recovery of service provider, network, subscription, number pool block, and notification data where the NPAC SMS replies to the originating SOA/LSMS with the missed data, by using linked replies. The NPAC SMS will keep track of messages destined for a SOA/LSMS that were NOT successfully responded to by the SOA/LSMS, when the service provider system supports SWIM recovery (SP Profile Flags, SOA SWIM Indicator = TRUE and LSMS SWIM Indicator = TRUE). Missed messages will be stored based on the limits of the SOA SWIM Maximum and LSMS SWIM Maximum tunables. SWIM based recovery requests can only be sent to the NPAC when the SOA/LSMS is in recovery mode, otherwise an abort message is returned.

During SWIM based recovery, the SOA/LSMS issue recovery requests for each type of data, and the NPAC SMS will issue recovery responses based on the SP Profile flags for ranges, and notification types for the missed messages and limit the responses by the respective Linked Reply Blocking Factor and Maximum Linked Recovered Object tunables for each data type. Each response from the NPAC SMS will contain a status and ACTION_ID. If the Service Provider system returns an invalid ACTION_ID, the NPAC SMS will abort the association. The status will be one of the following:

- Success

An NPAC SMS response that includes a status of Success indicates that SWIM recovery for the data type specified can be completed in either a single reply (with a status of Success and an ACTION_ID) or multiple linked replies (each with a status of Success and the same ACTION_ID in each reply, except for the last linked reply which will be empty – indicating the end of the linked reply data). In this case the Service Provider system must issue an M-EVENT-REPORT notification including the ACTION_ID in order for the NPAC SMS to clear the SWIM list for this data type and continue the recovery processing.

- Failed

An NPAC SMS response that includes a status of Failed indicates the NPAC failed to process the recovery request. An ACTION_ID is included, however the Service Provider system does not need to issue an M-EVENT-REPORT notification. The Service Provider system should re-start the recovery process with a new recovery request.

- No-Data-Selected

An NPAC SMS response that includes a status of No-Data-Selected indicates there isn’t SWIM data for the requested data type to recover. The response will include an ACTION_ID and the Service Provider system must issue an M-EVENT-REPORT notification including the ACTION_ID to continue the recovery processing.

- Swim-More-Data

An NPAC SMS response that includes a status of Swim-More-Data indicates that the SWIM recovery for the data type specified includes an amount of data greater than the Linked Reply Maximum and requires subsequent download request in order to recover all the data on the SWIM list.

When the Service Provider system receives an NPAC SMS ACTION response with linked replies that include an ACTION_ID and status of Swim-More-Data in each of the replies, the Service Provider system should issue a subsequent recovery request including this ACTION_ID. The NPAC SMS will issue an ACTION Response for the next set of data and clear the SWIM list for the (linked reply) data already downloaded and processed. This subsequent ACTION response from the NPAC SMS will include a new ACTION_ID (the same in each of the linked replies for this

response) and a status of either Swim-More-Data in each reply or Success in each reply followed by an empty reply. The Service Provider system and the NPAC SMS will continue this message exchange until the NPAC SMS ACTION Response indicates a status of Success (for each linked reply in that response).

After the Service Provider system receives an ACTION Response from the NPAC SMS indicating a status of Success and an ACTION_ID, the Service Provider system must issue an M-EVENT-REPORT notification including the most recent ACTION_ID in order for the NPAC SMS to clear this last set of (linked reply) data that was downloaded and processed, from the SWIM list for this data type and continue the recovery processing. If the Service Provider system returns an invalid ACTION_ID, the NPAC SMS will abort the association.

After the Service Provider system receives an ACTION Response from the NPAC SMS indicating a status of either Success or No-Data-Selected and an ACTION_ID, the Service Provider system must issue an M-EVENT-REPORT notification including the most recent ACTION_ID. If the Service Provider system returns an invalid ACTION_ID, the NPAC SMS will abort the association. The M-EVENT-REPORT reply from the NPAC SMS will contain one of the following responses:

- Success

An NPAC SMS M-EVENT-REPORT reply that includes a status of Success indicates that the recovery request has been completed for this data type and if there was data downloaded, that data has been cleared from the SWIM list.

- Failed With an Error Code

An NPAC SMS M-EVENT-REPORT reply that includes a status of Failed with an Error Code indicates that the recovery request failed and the Service Provider system should repeat recovery for that data type.

- Failed With an Error Code and a Stop-Date (timestamp)

An NPAC SMS M-EVENT-REPORT reply that includes a status of Failed with an Error Code and a stop-date (timestamp) indicates that the SWIM Maximum has been exceeded and the Service Provider’s SWIM indicator was changed from ON to OFF as of the time in the stop-date timestamp. The stop-date (timestamp), also indicates the time of the last SWIM entry onto the SWIM list. In this situation the Service Provider should perform further, ‘time-based’ recovery based upon the stop-date timestamp in order to recover all potentially missed messages for each data type they support. The Service Provider system may complete SWIM recovery for each data type and then request further time-based recovery for each data type:

For example:

SWIM (SP Data) – SWIM (Network Data) - SWIM (Subscription Data) –
 SWIM (NPB Data) – SWIM (Notification Data)
 – time-based (SP Data) – time-based (Network Data) – time-base
 (Subscription Data) – time-base (NPB Data) – time-based (Notification
 Data)

OR upon performing SWIM recovery and receiving the stop-date timestamp they may immediately perform time-based recovery for that same data type then SWIM based recovery for the next data type followed by time-based recovery for the same data type:

For example:

SWIM (SP Data) – time-based (SP Data) – SWIM (Network Data)
 - time-based (Network Data)
 etc.

Service Providers can continue to use the existing recovery mechanism/messages to recover data between the SOA/LSMS and the NPAC, using the ‘time-based’ or ‘record-based’ methods. However, if the Service Provider supports SWIM recovery, it is important that they first recover using the SWIM criteria. When the Service Provider supports SWIM recovery, their SWIM list is not “cleared” until successful SWIM recovery occurs, thus recovering by either time-based or record-based criterion first and SWIM subsequently may cause data integrity issues.

Upon completion of recovery, the SOA/LSMS should issue an `InpRecoveryComplete` message indicating the end of the missed data, and processing between the SOA/LSMS and NPAC SMS will resume normal mode.

5.3.4.1 Local SMS Recovery

To recover, the Local SMS starts by setting the `recoveryMode` flag of the access control parameter. This flag signals the NPAC SMS to hold all data updates to this Local SMS. The Local SMS should then request the service provider, network, subscription and number pool block data downloads and the notifications that occurred during downtime. Once this is complete, the Local SMS should issue the `InpRecoveryComplete` action to turn off the `recoveryMode` flag. After the NPAC SMS responds to the `InpRecoveryComplete` action it will send to the LSMS any other messages that have occurred since the association was established.

5.3.4.2 SOA Recovery

To recover, the SOA starts by setting the `recoveryMode` flag of the access control parameter. This flag signals the NPAC SMS to hold all data updates to this SOA. The SOA should then request the service provider, network data downloads and notifications that occurred during downtime. Once this is complete, the SOA should issue the `InpRecoveryComplete` action to turn off the `recoveryMode` flag. After the NPAC SMS responds to the `InpRecoveryComplete` action it will send to the SOA any other messages that have occurred since the association was established.

5.3.4.3 Linked Action Replies during Recovery

Linked Reply functionality applies to Service Providers that have their SOA Linked Replies Indicator set to TRUE, or their Local SMS Linked Replies Indicator set to TRUE.

For service provider that support linked replies the Maximum TN Download in Recovery Request, the Maximum Number of Download Notifications and Maximum Number of Download Records tunables do not apply to recovery processing.

Linked replies will be returned as the response to an `InpDownload` action request for *network* data if the number of messages returned exceeds the "Network Data Linked Reply Blocking Factor" tunable but is less than the "Network Data Maximum Linked Recovered Objects" tunable. If the number of network data objects to be returned exceeds the "Network Data Maximum Linked Recovered

Objects" tunable, a "criteria-too-large" error will be returned to the requesting SOA/LSMS.

Linked replies will be returned as the response to an InpDownload action request for *subscription* data if the number of objects returned exceeds the "Subscription Data Linked Reply Blocking Factor" tunable but is less than the "Subscription Data Maximum Linked Recovered Objects" tunable. If the number of subscription data objects be returned exceeds the "Subscription Data Maximum Linked Recovered Objects" tunable, a "criteria-too-large" error will be returned to the requesting LSMS.

Linked replies will be returned as the response to an InpDownload action request for *Number Pool Block* data if the number of objects returned exceeds the "Number Pool Block Data Linked Reply Blocking Factor" tunable but is less than the "Number Pool Block Data Maximum Linked Recovered Objects" tunable. If the number of Number Pool Block data objects be returned exceeds the "Number Pool Block Data Maximum Linked Recovered Objects" tunable, a "criteria-too-large" error will be returned to the requesting LSMS.

Linked replies will be returned as the response to an InpNotificationRecovery action request for *notification* data if the number of notifications returned exceeds the "Notification Data Linked Reply Blocking Factor" tunable but is less than the "Notification Data Maximum Linked Recovered Notifications" tunable. If the number of notifications to be returned exceeds the "Notification Data Maximum Linked Recovered Notifications" tunable, a "criteria-too-large" error will be returned to the requesting SOA/LSMS.

As an example, a Service Provider's SOA was down, and is now performing notification recovery. During the downtime, 90 notifications were issued. Assuming the Notification Blocking Factor is set to 50 notifications, the recovering SOA would receive data from the NPAC SMS in the form of three linked replies. The first reply would contain 50 notifications, the second reply would contain 40 notifications, and the third reply would be empty (this is an indication that the NPAC SMS is finished sending data for this recovery request). In the case where the amount of data to be returned is less than or equal to the associated Blocking Factor, the M-ACTION response will be a normal response (i.e., non-linked response) and will not be a linked reply.

Below are the tunables that specify the download size:

Download Criteria	Tunable Name
Network data download request maximum linked reply size	Network Data Linked Replies Blocking Factor
Subscription download request maximum linked reply size	Subscription Data Linked Replies Blocking Factor
Number Pool Block download request maximum linked reply size	Number Pool Block Data Linked Replies Blocking Factor
Notification download request maximum linked reply size	Notification Data Linked Replies Blocking Factor
Total number of network data objects returned for a single download request	Network Data Maximum Linked Recovered Objects
Total number of subscription data objects returned for a single download request	Subscription Data Maximum Linked Recovered Objects
Total number of Number Pool Block data objects returned for a single download request	Number Pool Block Data Maximum Linked Recovered Objects

Total number of notification data notifications returned for a single download request	Notification Data Maximum Linked Recovered Objects
----------------------------------------------------------------------------------------	----------------------------------------------------

Linked replies will be returned as the response to an action request from the recovering SOA/LSMS. The entire operation is considered complete once all linked replies and the final empty reply are returned as the response to the original request. Timeout processing is expected to start when the initial request is sent by the recovering SOA/LSMS, and terminate upon receipt of the final empty reply by the recovering SOA/LSMS.

5.4 Congestion Handling

The following sections define NPAC SMS behavior when in congestion and the NPAC handling of Local SMS and SOA congestion. The recommendation for Congestion Control follows the “Flow Control” mechanism and is described in OSI Communication Reference Model (ISO/IEC 7498). The two types of flow control defined are:

1. Peer Flow Control
2. Inter-Layer Flow Control

Peer Flow Control can be used when two peer layers of the OSI Stack talk to each other. The most common form of Peer Flow Control is the sliding window protocol. This protocol is implemented by TCP. This is the flow control approach used by the NPAC SMS.

5.4.1 NPAC SMS Congestion

Once the number of incoming messages to be queued to the NPAC SMS is exceeded at the transport layer, TCP/IP, an indication will be sent to the sender from the transport layer, TCP/IP, that congestion is occurring. Upon clearing of the congestion situation, the transport layer, TCP/IP will indicate to the sender that congestion has been cleared. As the receiver, the NPAC SMS application will not be aware that it is congested. The NPAC SMS application will be continually processing the information being sent as quickly as possible. Only the sender will be aware that the NPAC SMS is congested due to the fact that it can not send any more information to the NPAC SMS via the transport layer, TCP/IP. Implementation of functionality to handle NPAC congestion situations is at the discretion of SOA and LSMS vendors.

5.4.2 NPAC Handling of Local SMS and SOA Congestion

The NPAC SMS application must be able to handle congestion when attempting to send out a message to a SOA or LSMS system. When receiving indications of congestion via the transport layer from a SOA or LSMS the NPAC SMS application stops dispatching messages for the SPID (primary or associated) and SOA or LSMS interface that returned congestion. Note: If a SOA system returns congestion it will not affect the LSMS for the same service provider and vice versa. When the NPAC SMS stops dispatching messages to a congested SOA or LSMS, the retry attempts and retry timer values and the behavior associated with them apply to the messages not dispatched. The NPAC will abort the SOA or LSMS association once the retry attempts are exhausted. Any unacknowledged messages at the NPAC SMS application layer will be handled as failures as they are when an association is aborted today, for example for security reasons.

Once the NPAC SMS gets an indication via the transport layer that a SOA or LSMS system that was previously congested is ready to receive information, the NPAC SMS resumes sending of messages to that system. Note that the NPAC SMS will use the sequence number for the message it sends first that was the sequence number on the

message that was sent when congestion indication was received. This is done since the SOA or LSMS system did not receive this message. If the sequence number were incremented this would cause the SOA or LSMS to abort the association due to the sequence number value being larger than expected. SOA and LSMSs should use the same sequence number as well when communicating with the NPAC to prevent the NPAC from aborting the association due to the sequence number value being larger than expected.

5.4.3 Out-Bound Flow Control

Under normal conditions the NPAC SMS sends messages to the associated SOA/LSMS and the SOA/LSMS is able to keep up with the NPAC, and Flow Control is not encountered. However, under load conditions, the SOA/LSMS is not able to keep up with the messages sent from the NPAC SMS and Flow Control may be encountered.

For a SOA/LSMS that is currently in a normal state (not in Flow Control), the NPAC SMS monitors the number of outstanding, non-responsive messages sent to that system. If the number of outstanding, non-responsive messages is less than the Flow Control Upper Threshold (tunable value), NPAC sends the current message it is handling, and continues with normal processing. If the number of outstanding, non-responsive messages is equal to the Flow Control Upper Threshold tunable, the NPAC sends the current message it is handling, and sets the Flow Control flag to TRUE. In this situation Flow Control is encountered.

During Flow Control the NPAC SMS verifies the Flow Control flag setting for the destination SOA/LSMS to determine if it's OK to send each message. If the flag is FALSE, the message is sent; if the flag is TRUE the message is held/queued. In a Flow Control state, the NPAC SMS monitors the number of outstanding, non-responsive messages sent to that SOA/LSMS. If the number of outstanding, non-responsive messages is greater than the Flow Control Lower Threshold, no action is taken. When the number of outstanding, non-responsive messages is less than or equal to the Flow Control Lower Threshold (tunable value), the NPAC SMS resumes sending messages (whether queued or normal). A SOA/LSMS that is in a Flow Control state will have outstanding, non-responsive messages. For all outstanding, non-responsive messages that were sent, NPAC response timers and abort behavior will apply. For all messages NOT sent but held because the Flow Control flag is set to TRUE, NPAC response timers and abort behavior will NOT apply.

Flow Control is implemented on the NPAC SMS side of the CMIP interface and it is optionally implemented on the SOA/LSMS. The implementation of Flow Control by the sending system is independent of any implementation of Flow Control by the receiving system and is applicable on a per association basis. Flow Control applies to both normal mode and recovery mode and is applicable for service provider, network, number pool block, subscription version and notification data.

5.5 Abort Processing Behavior

The NPAC exchanges messages with the SOA/LSMS. For every request from the NPAC, a response is required from the SOA/LSMS. A SOA/LSMS that fails to respond to a message is subject to Abort Processing Behavior (APB).

The NPAC sends messages to the associated SOA/LSMS. For every message sent, abort behavior is initiated, and a Roll-Up Activity (RAT) timer is started. The initial abort timer is based on existing retry functionality. The RAT timer is either the Roll-Up Activity-Single (RAT Single) tunable value or the Rollup Activity Timer Expire SVRange (RAT Range) tunable value. The secondary abort timer is the Abort Processing Behavior Upper Threshold tunable window. The NPAC allows a SOA/LSMS to fall

behind in processing messages. However, the limit is defined by the Abort Processing Behavior Upper Threshold tunable window, upon which when this value is reached the association is aborted.

The NPAC SMS “rolls-up” downloaded data (e.g. SV activate to LSMSs) to reflect the status of porting activity. Abort behavior and roll-up behavior are separate items, but often confused because both can happen at the same time when a timer expires.

During message exchange between the NPAC SMS and the SOA/LSMS the response from the SOA/LSMS is one or more of the options below, based on the tunable settings:

- All SOAs/LSMSs respond to the NPAC SMS message before the end of the retry window and RAT timer expiration.
 - In this instance the NPAC SMS expires the RAT timer for that event and with a successful response, the NPAC SMS considers the responding SOA/LSMS as “successful” to the request (e.g. the SPID is not placed on the Failed-SP List).
- All SOAs/LSMSs do NOT respond to the NPAC SMS before the end of the retry window (e.g. end of the “X by Y” window).
 - The retry timer has expired based on the applicable retry value. The NPAC SMS determines if any messages/responses were received from this SOA/LSMS during the retry window.
 - If at least one message/response is received from the SOA/LSMS, processing continues.
- All SOAs/LSMSs do NOT respond to the NPAC SMS before the end of the RAT timer expiration. The RAT timer has expired based on the applicable value (either single or range).
 - The NPAC SMS performs “roll-up” activities for all messages sent to the SOAs/LSMSs on this event (status is set, Failed-SP List(s) is updated appropriately and notifications are sent to respective SOAs).
- SOA/LSMS responds to the NPAC SMS AFTER the expiration of the RAT timer.
 - The NPAC SMS updates status/Failed-SP List, and sends notifications to respective SOAs.
- SOA/LSMS does NOT respond to the NPAC SMS before the end of the secondary abort (Abort Processing Behavior Upper Threshold tunable) window.
 - The NPAC SMS aborts the association to the SOA/LSMS and the SOA/LSMS must re-associated to the NPAC SMS.
 - The SOA/LSMS goes through recovery processing (recovery based on SOA/LSMS linked replies indicator) and the NPAC SMS updates the status/Failed-SP List, and sends notifications to the SOAs.

Abort processing behavior applies to both normal and recovery modes. Service provider, network, number pool block, subscription version and notification messages are subject to Abort processing behavior.

5.6 Single Association for SOA/LSMS

A SOA/LSMS system may connect to the NPAC SMS with one association for the same function (same bit mask). The NPAC SMS will abort any previous associations that use that same function.

5.7 Separate SOA Channel for Notifications

A SOA system may connect to the NPAC SMS with multiple SOA channels (i.e., associations) for different functions (different bit masks), specifically request/response data versus notification data. The NPAC SMS will distribute transactions across these SOA associations based on functionality (different bit masks). This allows for additional throughput for the SOA as a result of two associations.

6 *GDMO Definitions*

6

The latest version of the GDMO interface definitions is available on the NPAC website (www.npac.com, under the documents section).

7 General ASN.1 Definitions

7

The latest version of the LNP ASN.1 Object Identifier definitions is available on the NPAC website (www.npac.com, under the documents section).

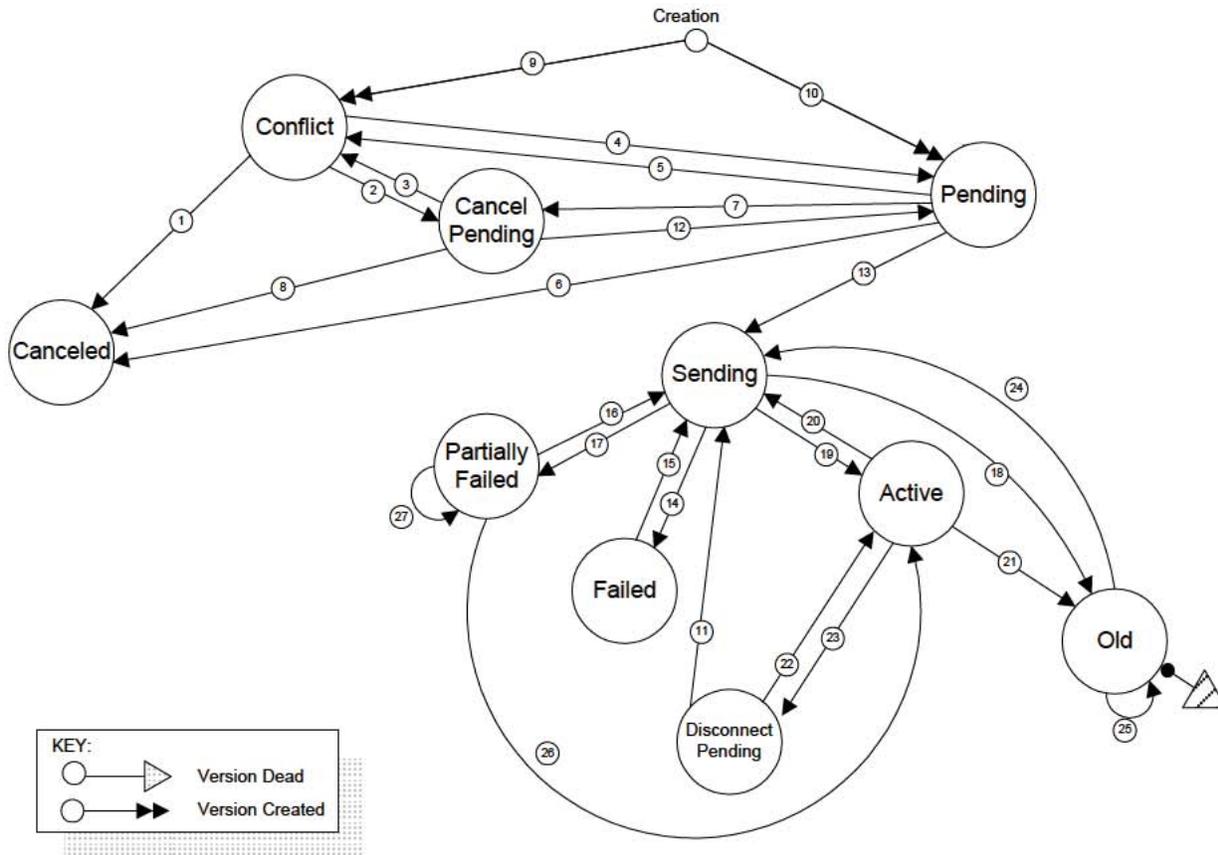
8 *LNP XML Schema*



The latest version of the LNP XML schema is available on the NPAC website (www.npac.com, under the documents section).

9 Subscription Version Status

9



Subscription Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
1	Conflict to Canceled	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version in conflict directly to canceled after it has been in conflict for a tunable number of calendar days.

Subscription Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
		SOA to NPAC SMS Interface or NPAC Operations Interface - NPAC Personnel	The old Service Provider User (or NPAC personnel acting on behalf of the Service Provider) sends a cancellation request for a Subscription Version created by that Service Provider with a status of conflict that has not been concurred by the other new Service Provider.
2	Conflict to Cancel Pending	NPAC Operations Interface - NPAC Personnel	User cancels a Subscription Version in conflict or cancels a Subscription Version that was created by or concurred to by both Service Providers.
		SOA to NPAC SMS Interface	User sends a cancellation request for a Subscription Version that was created by or concurred to by both Service Providers.
3	Cancel Pending to Conflict	SOA to NPAC SMS Interface or NPAC SOA Low-tech Interface	Service Provider User sends an un-do cancel-pending request for a Subscription Version with a status of cancel-pending for which the same Service Provider previously issued a cancel request.
		NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version with a status of cancel pending to conflict if cancel pending acknowledgment has not been received from the new Service Provider within a tunable timeframe.
4	Conflict to Pending	NPAC Operations Interface - NPAC Personnel and SOA to NPAC SMS Interface - Old Service Provider	User removes a Subscription Version from conflict.
		SOA to NPAC SMS Interface - New Service Provider	New Service Provider User removes a Subscription Version from conflict. This action can only occur if a tunable number of hours have elapsed since the Subscription Version was placed in conflict.
5	Pending to Conflict	NPAC Operations Interface - NPAC Personnel	<ol style="list-style-type: none"> 1. User sets a Subscription Version with a status of pending to conflict. 2. User creates a Subscription Version for an existing pending Subscription Version for the old Service Provider and does not provide authorization for the transfer of service.
		SOA to NPAC SMS Interface - Old Service Provider	Old Service Provider sends a Subscription Version creation or modification request for a Subscription Version with a status of pending, which revokes the old Service Provider's authorization for transfer of service. This action can only be taken once, and must be taken a tunable number of hours prior to the new Service Provider due date.
6	Pending to Canceled	NPAC Operations Interface - NPAC Personnel	User cancels a Subscription Version with a status of pending that has not been concurred by both service providers.
		SOA to NPAC SMS Interface	Service Provider User sends a cancellation request for a Subscription Version created by that Service Provider with a status of pending that has not been concurred by the other Service Provider.

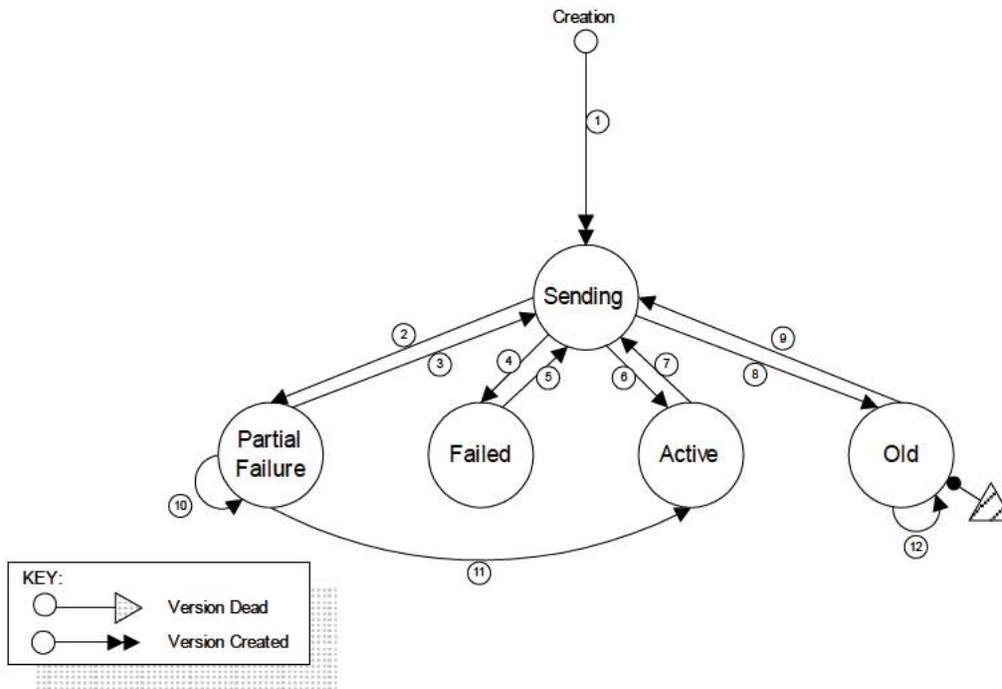
Subscription Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
		NPAC SMS Internal	<ol style="list-style-type: none"> NPAC SMS automatically sets a pending Subscription Version to canceled after authorization for the transfer of service has not been received from the new Service Provider within a tunable timeframe. NPAC SMS automatically sets a pending Subscription Version to canceled if an activation request is not received a tunable amount of time after new Service Provider due date.
7	Pending to Cancel Pending	NPAC Operations Interface - NPAC Personnel	User cancels a Subscription Version with a status of pending that has been created/concurred by both Service Providers.
		SOA to NPAC SMS Interface	Service Provider User sends a cancellation request for a Subscription Version with a status of pending that has been concurred by the other Service Provider.
8	Cancel Pending to Canceled	NPAC SMS Internal	NPAC SMS automatically sets a cancel pending Subscription Version to canceled after receiving cancel pending acknowledgment from the concurring Service Provider, or the final cancellation concurrence window has expired without cancel concurrence from the old Service Provider.
9	Creation - Set to Conflict	NPAC Operations Interface - NPAC Personnel	User creates a Subscription Version for the old Service Provider and does not provide authorization for the transfer of service.
		SOA to NPAC SMS Interface - Old Service Provider	User sends an old Service Provider Subscription Version creation request and does not provide authorization for the transfer of service.
10	Creation - Set to Pending	NPAC Operations Interface - NPAC Personnel	User creates a Subscription Version for either the new or old Service Provider. If the create is for the old Service Provider and authorization for the transfer of service is not provided, refer to # 9, <i>Creation - Set to Conflict, NPAC Operations Interface</i> .
		SOA to NPAC SMS Interface	User sends a Subscription Version creation request for either the new or old Service Provider. If the create is for the old Service Provider, and authorization for the transfer of service is not provided, refer to # 9, <i>Creation - Set to Conflict, SOA to NPAC SMS Interface</i> .
11	Disconnect Pending to Sending	NPAC SMS Internal	NPAC SMS automatically sets a deferred disconnect pending Subscription Version to sending after the effective release date is reached.
12	Cancel Pending to Pending	SOA to NPAC SMS Interface or NPAC SOA Low-tech Interface	Service Provider User sends an un-do cancel-pending request for a Subscription Version with a status of cancel-pending for which the same Service Provider previously issued a cancel request.
13	Pending to Sending	NPAC Operations Interface - NPAC Personnel	User activates a pending Subscription Version for a Subscription Version with a new Service Provider due date less than or equal to today.
		SOA to NPAC SMS Interface - New Service Provider	New Service Provider User sends an activation message for a pending Subscription Version for a Subscription Version with a new Service Provider due date less than or equal to today.
14	Sending to Failed	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version from sending to failed after all Local SMSs fail Subscription Version activation after the tunable retry period expires.
15	Failed to Sending	NPAC Operations Interface - NPAC Personnel	User re-sends a failed Subscription Version.

Subscription Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
16	Partially Failed to Sending	NPAC Operations Interface - NPAC Personnel	User re-sends a partial failure Subscription Version.
17	Sending to Partially Failed	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version from sending to partial failure after one or more, but not all, of the Local SMSs fail the Subscription Version activation after the tunable retry period expires.
18	Sending to Old	NPAC SMS Internal	NPAC SMS automatically sets a sending Subscription Version to old after a disconnect or “porting to original” port to all Local SMSs successfully completes. Disconnects that fail on one or more, but not all, Local SMSs will also be set to old.
19	Sending to Active	NPAC SMS Internal	<ol style="list-style-type: none"> 1. NPAC SMS automatically sets a sending Subscription Version to active after the Subscription Version activation is successful in all of the Local SMSs. 2. NPAC SMS automatically sets a sending Subscription Version to active after the Subscription Version modification is successfully broadcast to any of the Local SMSs after all have responded. 3. NPAC SMS automatically sets a sending Subscription Version to active after a failure to all Local SMSs on a disconnect.
20	Active to Sending	NPAC Operations Interface - NPAC Personnel	User disconnects an active Subscription Version and does not supply an effective release date, or User modifies an active Subscription Version or resends a failed disconnect or modify.
		SOA to NPAC SMS Interface - Current Service Provider	User sends a disconnect request for an active Subscription Version and does not supply an effective release date, or User modifies an active Subscription Version.
21	Active to Old	NPAC SMS Internal	NPAC SMS automatically sets the currently active Subscription Version to old once a currently active subscription version is superceded by a pending subscription version, due to the fact that the current version is set to old when an activate occurs. The new pending version is set to sending and then to active, partially failed, or old. On a disconnect the sending state occurs before the old.
22	Disconnect Pending to Active	NPAC Operations Interface - NPAC Personnel	User cancels a Subscription Version with a disconnect pending status.
		SOA to NPAC SMS Interface - New Service Provider	User sends a cancellation request for a disconnect pending Subscription Version.
23	Active to Disconnect Pending	NPAC Operations Interface - NPAC Personnel	User disconnects an active Subscription Version and supplies a future effective release date.
		SOA to NPAC SMS Interface - Current Service Provider	User sends a disconnect request for an active Subscription Version and supplies a future effective release date.
24	Old to Sending	NPA Operations Interface – NPAC Personnel	User re-sends a partial failure of a disconnect or partial failure or failure of a port-to-original Subscription Version.

Subscription Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
25	Old to Old	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version from old to old after one or more previously failed Local SMSs successfully disconnect a Subscription Version, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.
26	Partially Failed to Active	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version from partial failure to active after all previously failed Local SMSs successfully activate a Subscription Version, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.
27	Partially Failed to Partially Failed	NPAC SMS Internal	NPAC SMS automatically sets a Subscription Version from partial failure to partial failure after one or more, but not all previously failed Local SMSs successfully activate a Subscription Version, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.

10 Number Pool Block Status

9



Number Pool Block Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
1	Creation - Set to Sending	NPAC SMS Internal	NPAC SMS creates a Number Pool Block for the Block Holder Service Provider.
		NPAC Operations Interface - NPAC Personnel	User sends a Number Pool Block creation request for the Block Holder Service Provider.
		SOA to NPAC SMS Interface - Block Holder Service Provider	The Service Provider User sends a Number Pool Block creation request for itself (the Block Holder Service Provider).
2	Sending to Partial Failure	NPAC SMS Internal	NPAC SMS automatically sets a Number Pool Block from sending to partial failure after one or more, but not all, of the Local SMSs fail the Number Pool Block activation after the tunable retry period expires.

Number Pool Block Version Status Interaction Descriptions			
#	Interaction Name	Type	Description
3	Partial Failure to Sending	NPAC Operations Interface - NPAC Personnel	User re-sends a partial failure Number Pool Block.
4	Sending to Failed	NPAC SMS Internal	NPAC SMS automatically sets a Number Pool Block from sending to failed after all Local SMSs fail Number Pool Block activation after the tunable retry period expires.
5	Failed to Sending	NPAC Operations Interface - NPAC Personnel	User re-sends a failed Number Pool Block.
6	Sending to Active	NPAC SMS Internal	<ol style="list-style-type: none"> 1. NPAC SMS automatically sets a sending Number Pool Block to active after the Number Pool Block activation is successful in all of the Local SMSs. 2. NPAC SMS automatically sets a sending Number Pool Block to active after the Number Pool Block modification is broadcast to all of the Local SMSs and either all have responded or retries have been exhausted. 3. NPAC SMS automatically sets a sending Number Pool Block to active after a failure to all Local SMSs on a de-pool.
7	Active to Sending	NPAC Operations Interface - NPAC Personnel	<ol style="list-style-type: none"> 1. User de-pools an active Number Pool Block. 2. User modifies an active Number Pool Block. 3. User resends a failed de-pool or modify Number Pool Block.
		SOA to NPAC SMS Interface - Block Holder Service Provider	User modifies an active Number Pool Block.
8	Sending to Old	NPAC SMS Internal	<ol style="list-style-type: none"> 1. NPAC SMS automatically sets a sending Number Pool Block to old after a de-pool to all Local SMSs successfully completes. 2. NPAC SMS automatically sets a sending Number Pool Block to old after a de-pool that fails on one or more, but not all Local SMSs.
9	Old to Sending	NPA Operations Interface – NPAC Personnel	User re-sends a partial failure of a de-pool.
10	Partial Failure to Partial Failure	NPAC SMS Internal	NPAC SMS automatically sets a Number Pool Block from partial failure to partial failure after one or more, but not all previously failed Local SMSs successfully activate a Number Pool Block, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.
11	Partial Failure to Active	NPAC SMS Internal	NPAC SMS automatically sets a Number Pool Block from partial failure to active after all previously failed Local SMSs successfully activate a Number Pool Block, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.
12	Old to Old	NPAC SMS Internal	NPAC SMS automatically sets a Number Pool Block from old to old after one or more previously failed Local SMSs successfully de-pools a Number Pool Block, as a result of an audit or LSMS recovery. The Failed_SP_List is updated to reflect the updates to the previously failed SPs.