AM-TR-OAT-000043

Ameritech Service Control Point Functional Specification

To:	Ameritech and Vendor Community
Priority:	N/A
Effective Date:	July 1989
Issue Date:	Issue 1, July 1989
Expires On:	N/A
Training Time:	N/A
Related Documents:	N/A
Canceled Documents:	N/A
Issuing Department:	N/A
Distribution:	NA
Business Unit:	N/A

Points of Contact:
xxx

Author(s):
xxx

Table of Contents

TECHNICAL REFERENCE NOTICE

This Technical Reference is published by Ameritech to provide a technical description of the Intelligent Network Service Control Point Functional Specification.

Ameritech reserves the right to revise this document for any reason, including but not limited to conformity with standards promulgated by various agencies, utilization of advances in the state of the technical areas, or the reflection of changes in the design of any equipment, techniques or procedures described or referred to herein.

AMERITECH MAKES NO REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, WITH RESPECT TO THE SUFFICIENCY, ACCURACY, OR UTILITY OF ANY INFORMATION OR OPINION CONTAINED HEREIN. AMERITECH EXPRESSLY ADVISES THAT ANY USE OF OR RELIANCE UPON THIS TECHNICAL REFERENCE IS AT THE RISK OF THE USER AND THAT AMERITECH SHALL NOT BE LIABLE FOR ANY DAMAGE OR INJURY INCURRED BY ANY PERSON ARISING OUT OF THE SUFFICIENCY, ACCURACY, OR UTILITY OF ANY INFORMATION OR OPINION CONTAINED HEREIN.

This document is not to be construed as a suggestion to any manufacturer to modify or change any of its products, nor does this document represent any commitment by Ameritech or any Ameritech Operating Company (AOC) to purchase any product, whether or not it provides the described characteristics.

Ameritech does not recommend products, and nothing contained herein is intended as a recommendation of any product to anyone.

Nothing contained herein shall be construed as conferring by implication, estoppel, or otherwise any license or right under any patent, whether or not the use of any information herein necessarily employs an invention of any existing or later issued patent.

Ameritech reserves the right not to offer any or all of these services and to withdraw any or all of them at any future time.

For further technical information regarding this document, contact:

Director, Technical Services — Intelligent Network, Ameritech Services, Inc., 1900 E. Golf Road, Floor 9, Schaumburg, Illinois 60173

Document may be ordered from Ameritech by contacting the Document Order Center at (847) 248-4324.

1. Introduction

1.1. Purpose and Scope

This document details the functionality of the Service Channel Point (SCP) for Ameritech's view of Intelligent Network (IN) Release 0. The IN Release 0 SCP can be viewed as an SCP node as described in *Service Control Point Node Generic Requirements* [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.] as well as platform of additional functionality required to provide IN Release 0 services. The platform includes service independent functionalities that are available to a service designer for building IN Release 0 services as well as additional functionalities (e.g., measurements, data sampling) to support the services. This document will concentrate on the feature functionality of the IN Release 0 SCP [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.] and assume the existence of a basic system as described in *Service Control Point Node Generic Requirements*. [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.] [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] Features for individual subscribers are encoded as Functional Logic customized with subscriber data in the SCP. When a switching system with additional Service Switching Point (SSP) capabilities recognizes a call requiring processing by a feature implemented with the IN Release 0 SCP, it queries the SCP. After interpreting the message and the Functional Logic populated with the subscriber data, the SCP responds with instructions for handling the call.

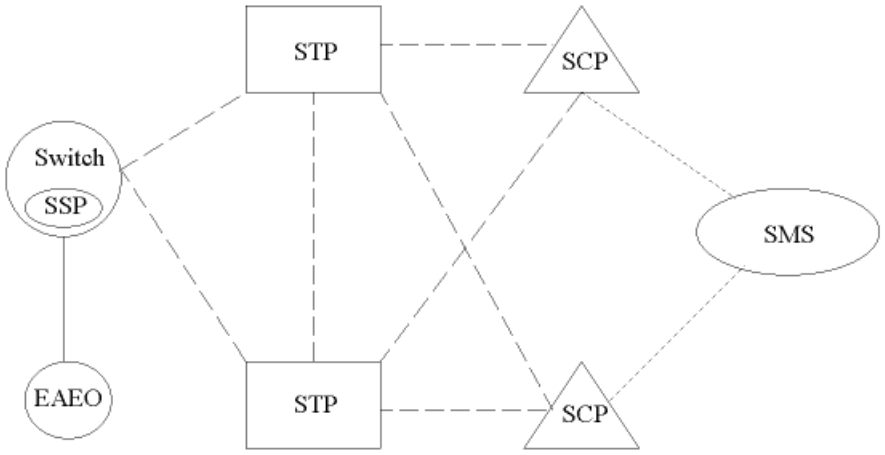
1.1.1. Network Architecture

The SCP and SSP communicate over a Common Channel Signaling (CCS) network. When the SSP recognizes that a call requires IN Release 0 treatment, it sends a message on the CCS network. The destination address in the message identifies a Signal Transfer Point (STP) capable of performing a global title translation for an SCP query. The STP performs the translation to obtain the appropriate IN Release 0 database Subsystem Number (SSN) and the address of the SCP to which the message should be sent. Normally, the global title translation will be performed at the STP; however, an option is provided to perform a global title translation at the SSP. For reliability, the SCP should be deployed in mated pairs.

Each SCP in a mated pair contains the same set of Functional Logic and subscriber data, but the STP translations route calls with certain Global Titles (based on service key) to one SCP and calls with other Global Titles to its mate. If one SCP fails, the STPs redirect *all* traffic to the other SCP. The data in SCPs will be administered by the Service Management System (SMS).

Figure 1 shows the CCS network configuration with the IN Release 0 SCP included.

Figure 1. CCS Network Configuration with Release 0 SCP



Note: SCP referes to the IN Release 0 SCP
Note: EAEO can be either a conforming NAP or a non-conforming NAP

- - - - - - SS7
..... BX.25/X.25
——— Voice

1.1.2. Platform Functionalities

The IN Release 0 platform functionalities can be divided into three categories: general functionalities for providing instructions and/or responses to SSPs, functionalities for monitoring on an ongoing basis and functionalities for defining SCP features.

A. The SCP is responsible for providing the following instructions and/or responses to SSPs in the IN Release 0 network:

- Routing Instructions - can include the following:
  - Carrier ID
  - Alternate Carrier ID
  - Second alternate Carrier ID
  - Routing number (national or international)

  - Outpulse number (network specific)
  - Traveling class mark (for routing over private facilities)
  - Business Customer ID
  - Primary trunk group
  - Alternate trunk group
  - Second alternate trunk group
  - Billing indicators
  - Alternate Billing indicators
  - Second Alternate billing indicators
  - Digits for billing purposes (e.g., PIN, Authorization code, Business Customer ID)
  - Network management controls
  - Request for termination information

- Play announcement - can include the following:
  - Tone or announcement ID
  - Network management controls
- Play announcement(s)/Collect digit(s) - (for intermediate instructions) includes the following:
  - Tone or announcement ID
  - Number of digits expected
- Error handling instructions - can include the following:
  - Error Information

Query and response message are discussed in Sections 2.1.2, 2.1.3 and 2.1.4. Error handling is discussed in Section 2.1.5. For a complete

description of the individual components and the associated parameters, refer to the *Ameritech SCP-SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.].

B. The SCP is expected to monitor the following conditions on an ongoing basis:

- Resource usage information
- Functional Logic errors
- Network management conditions (node and subscriber level)
- Traffic and Performance measurements

Resource usage information is discussed further in Section 2.5.2.7. Feature logic errors are discussed further in Section 2.2.1.6. Network management conditions are discussed further in Sections 2.2.1.4, 2.2.3 and 2.5.1. Traffic and Performance measurements are discussed further in Section 2.5.3.2.

C. There are four categories of functionalities available for defining IN Release 0 Features. [As defined by Ameritech, e.g. originating call screening, etc.]

- Service Key analysis functionalities
- Decision analysis functionalities
- Informational functionalities
- Operations, administration and maintenance (OA&M) functionalities

These are internal platform functionalities which are used in query processing to provide SCP features. [AN SCP feature can include one or more of the IN Release 0 Features as defined by Ameritech.] These functionalities can be combined in various ways and with a number of parameters to define SCP features.

The service key analysis functionalities are those which provide the service key analysis function which results in the identification of the Feature Logic and the subscriber data.

The decision analysis functionalities alter the result of query processing based on time, caller provided information and other conditions. Thus, each query processed

may receive different treatment. Some examples of decision analysis are time of day and percentage routing.

The informational functionalities provide the appropriate information to be included in the conversation or response messages to the SSP. These informational functionalities may be included anywhere in the query analysis process.

Finally, the OA&M functionalities are general functionalities, which do not affect the Functional Logic operation but rather are support functions to be performed on a subscriber level. For example, subscriber statistics on usage or feature logic sample data can be supported using the OA&M functionalities.

An expansion of these functionalities is defined in section 2.2.1.

1.2. User Perspective

1.2.1. Caller

The SCP's intervention on a call is transparent to the caller. Assuming only one query and one response is sent, the delay (i.e., the time until which the user receives some type of call treatment once the user finishes initial dialing) from SSP and SCP interaction is expected to be between 1.5 and 2.5 seconds (see Section 2.7.2 for the SCP expected delay).

1.2.2. Subscriber

The subscriber may have use of the Service Management System (SMS), which will provide the only interface for service definition and changes. The SMS will also control the collection and processing of sample data for subscriber reports.

1.3. Document Organization

This document begins my detailing Call Processing Treatment and Internal Call Processing. Although Signaling and Transmission are identified, they are felt not to be applicable. Finally, Administration, Maintenance, and Performance reviewed.

In addition, Appendix A provides a Feature Definition Example and Appendix B lists the deviations of the functions specified in this document from the Service Control Point Node Generic Requirements [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.].

1.4. Reason for Reissue

Whenever this document is reissued, the reason(s) for reissue will be stated under this heading.

Questions or statements regarding this document should be submitted in writing to:

Director, Technical Services - Intelligent Network, Ameritech Services, 1900 E. Golf Road Floor 9, Schaumburg, Illinois 60173

1.5. Definitions

The following terms are used throughout this document and are defined here so as to avoid confusion with similar or identical terms used in other Intelligent Network documents.

Service - A service providers marketed package of features and capabilities which are provided to end users as tariffed offerings.

Feature/Capabilties - Features or Capabilities are incremental elements, perhaps having various options, which can be combined to form services or stand along as a service.

Functional Logic Unit - Network entity (e.g. Service Control Point) logic packages which are used by various features or capabilities and are represented in the SMS as Feature Logic Units; Telco-level or programming.

Vendor Specific Code - Programs provided by the equipment manufacturer, specific to the network entity.

2. Platform Functionality Requirements

This section contains Ameritech's view of functional requirements needed for the IN Release 0 SCP. The requirements assume the existence of and provide additional functionality to a generic database system as described in *Service Control Point Node Generic Requirements*. [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.]

The SCP should be able to perform three major functions in support of the service network:

- Accept and interpret incoming Transaction Capabilities Application Part (TCAP) messages
- Interpret and process Functional Logic using information contained in the incoming TCAP messages
- Provide information for TCAP messages in response to incoming messages.

The content and structure of the TCAP messages is discussed in the sections that follow. The SCP processing required for the messages and the Functional Logic is also discussed.

2.1. Call Processing Treatments

This section describes the CCS messages and common SCP procedures for call processing. *Ameritech SCP-SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.], provides the detailed message formats. Specific procedures determined by Functional Logic and subscriber data or SCP overload and failure conditions are described in Section 2.2.

The SCP should receive call processing requests via query messages as described in Section 2.1.2. The SCP should process the query and send a message to the SSP that originated the query message. For certain calls, the SSP may later send a termination message that reports call termination information. The response and termination messages are described in Section 2.1.4. Messages used for caller interaction are described in Section 2.1.3. The procedures for handling errors encountered in message processing are described in detail in *Ameritech SCP-SSP Interface Specification*. [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.]

2.1.1. SCP Message Processing

The SCP provides functions that are both independent of, and common to, all Functional Logic processing. It serves as an interface to the CCS network by performing the SS7 functions required for message transfer. The TCAP layers of the protocol provide the information required by the Functional Logic and subscriber data to process an incoming query.

In the Signaling Connection Control Part (SCCP) layer, the SCP provides protocol class 0, basic connectionless service. For messages sent to the CCS network, the calling and called party addresses are included in the SCCP data. The SCP formats the SCCP part of the outgoing message as described in *Ameritech SCP - SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] as well as in *Service Control Point Node Generic Requirements*.

For the TCAP layer, the SCP passes the data elements of incoming messages to the Functional Logic and formats outgoing messages based on data elements received from the Functional Logic.

The following sections describe the TCAP information messages exchanged between the SCP and an SSP.

2.1.2. Originating Calls

2.1.2.1. Initial Query Message Contents; SSP INVOKE (Provide Instructions: Start)

The SSP requests instructions for a call by sending a query message to the SCP for processing. The query message may contain the following parameters and respective data elements: [The actual content of the query message is dependent upon the trigger encountered and the information available at the SSP. Details can be found in *Ameritech SSP Functionality Specification* [As defined by Ameritech, e.g. originating call screening, etc.] and *Ameritech SCP-SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.].]

Parameters	Data Elements
Package Type Identifier	Query with Permission
Originating Transaction ID	SSP assigned
Component Type	INVOKE (Last)
Invoke ID	SSP assigned
Operation Code	Provide instructions: Start; reply required
Service Key	Automatic Number Identification (ANI)-(calling) or Dialed (called)
Digits	Originating LATA
Digits	Originating station DN
Digits	Dialed (called)
Digits	Authorization code
Digits	Traveling class mark
Originating Station Type	Global value for IN Release 0 Services
Automatic Call Gap (ACG) Encountered	Type of control encountered and number of digits under control

The SCP should ensure that a conversation or response message be sent to the SSP within 3 seconds of receipt of the query. If query processing takes longer than 3 seconds, a conversation or response message should not be sent, query processing should stop, and all information about the call should be discarded (see Section 2.2.2.3A for further information.)

The Service Key should be used to identify the Functional Logic and subscriber data that should be used to obtain routing information and other details to be provided in the corresponding response message. The data elements from the initial query that can be used in the decision analysis of the Functional Logic are discussed further in Section 2.2.1. If the SCP fails to find the Functional Logic or subscriber data, a response message should be formatted that directs

the SSP to play an announcement and give the call final treatment. If the SCP locates the Functional Logic and subscriber data, call processing should proceed as described in Section 2.1.3.

If the ACG Encountered field is included in the message, it will indicate which type of ACG control was encountered (see Sections 2.2.3 and 2.5.1). The eighth bit is used to indicate SCP overload control (1 = SCP overload control encountered, 0 = no SCP overload control encountered). The seventh bit is used to indicate SMS Originated Code Control (SOCC) [NM control is indicated as “SMS Originated” not “Selective Originated” Code Control because, for IN Release 0, control can be placed on either originating or terminating numbers, depending upon which type of number is used for the service key.] (1 = SOCC encountered, 0 = no SOCC encountered).

When the ACG Encountered field is included in the message, the SCP should determine if the number contained in the Service Key is still under control. Based on the SSP control encoun-

tered and the type of control currently active in the SCP, Table 1 indicates the type of control that should be included in the ACG component of the SCP response (see Section 2.1.4.1B).

Table 1.
ACG Control

SSP Control Encountered	SCP Current Controls	SCP Response ACG Component
SCP Overload	No Control SCP Overload SOCC Both (SCP Overload and SOCC)	Remove SCP Overload Update SCP Overload Remove SCP Overload Initiate SOCC
SOCC	No Control SCP Overload SOCC Both (SCP Overload and SOCC)	Remove SOCC Initiate SCP Overload Update SOCC Initiate SCP Overload
No Control	No Control SCP Overload SOCC Both (SCP Overload and SOCC)	Initiate SCP Overload Initiate SOCC Initiate SCP Overload

2.1.3. Caller Interaction

Functional Logic and subscriber data may dictate the need for caller interaction. In this case, the SCP requests the SSP to play a particular announcement and collect a number of digits, from one to fifteen, as specified by the subscriber data. Transaction IDs and Invoke IDs should be maintained by the SCP to correlate queries with responses, from the SSP, containing the collected digits. On the following page are the query and response messages needed to provide caller interaction.

2.1.3.1. SCP INVOKE (Play announcement and collect digits)

The parameters and respective data elements of this message are:

Parameters	Data Elements
Package Type Identifier	Converstaion with Permission
Originating Transaction ID	SCP assigned
Responding Transaction ID	SSP assigned
Component Type	INVOKE (not Last)
Invoke ID	SCP assigned
Correlation ID	Identical to SSP-assigned Invoke ID in initial query
Operation Code	Caller interaction; reply required
Customized Announcement	Announcement/Tone ID
Number of Digits	Number of digits to be collected

Message for caller interaction can contain only one INVOKE (Play announcement and collect digits) Component. The Number of Digits field should be encoded to specify the number of digits, one to fifteen, to be collected or to specify that a variable number of digits, from one to fifteen, should be collected.

Once the messages is sent, the SCP should start a TA timer (see Section 2.2.2.3B for the timer value). If a response is not received from the network node before the timer expires, the SCP should release resources associated with the call. If the network node returns a response after the timer at the SCP has expired, the SCP cannot continue processing for the call because it has discarded all information concerning the call; therefore, the SCP should return a response message containing a Reject Component to the SSP. (See *Ameritech SCP-SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] for details on error handling.)

2.1.3.2. SSP RETURN RESULT (Collected digits)

The parameters and respective data elements of this message are:

Parameters	Data Elements
Package Type Identifier	Converstaion with Permission
Originating Transaction ID	SSP assigned
Responding Transaction ID	SCP assigned
Component Type	RETURN RESULT (Last)
Correlation ID	Identical to SCP assigned Invoke ID from query
Digits	Caller interaction
Standard User Error	Improper Caller Response

The above message is sent after the appropriate network node plays the announcement and collects the required digits. The message contains one Component. Once the SCP receives the above message, it should ensure that a conversation or response message is sent within 3 seconds to the SSP. Otherwise, the SCP should not send a conversation or response message, query processing should stop, and all information about the call should be discarded (see Section 2.2.2.3 for more information).

If the caller does not dial a sufficient number of digits, then the Standard User Error (Improper caller response) will be included. If the caller should abandon without dialing the requested numbers, the Response message with a Return Result Component indicating a Standard User Error (Called Abandon) will be sent. For further details on error handling procedures, see *Ameritech SCP-SSP Interface Specification*. [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.]

2.1.4. Final Call Handling

If the SCP does not encounter errors in the messages set by the SSP (see Sections 2.1.2 and 2.1.3.3) and is able to formulate routing instructions based on information contained in the message, it should send a response message to the SSP that initiated the query. This message should contain one to three TCAP Components.

A. The INVOKE (Connection Control) Component.

This Component, which is always present in the message, contains the SSP instructions for call handling.

B. The INVOKE (Network Management [NM]: Automatic Call Gapping) Component.

This Component is not present in every message. Procedures for generating this Component are described in Sections 2.2.1.4, 2.2.3 and 2.5.1.2.

C. The INVOKE (Send Notification: Termination) Component.

This Component is not present in every message. It is generated by one or a combination of the following:

- Call sampling function in the Functional Logic and subscriber data (see Section 2.2.1)
- Call management (e.g., Resource Counters) function in the Functional Logic and subscriber data (see Section 2.2.1).

This Component causes the SSP to send the SCP a subsequent termination message, which is described in Section 2.1.4.2.

If the SCP does not encounter errors in the messages sent by the SSP and Functional Logic and subscriber data dictate that no routing instructions should be sent, the SCP should send a response message to the SSP that initiated the query. This message should contain one or two TCAP Components.

A. The INVOKE (Play Announcement) Component.

This Component, which is always present in the message, contains SSP instructions for final call treatment.

B. The INVOKE (NM: Automatic Call Gapping) Component.

This Component is not present in every message. Procedures for generating this Component are described in Sections 2.2.1.4, 2.2.3 and 2.5.1.2.

2.1.4.1. Response Message Contents

All information to be included in the response message is determined by Functional Logic and subscriber data instructions.

A. First Component: INVOKE (Connection Control) - Mandatory if routing instructions are to be provided.

The parameters and respective data elements of this first Component are shown below:

Parameters	Data Elements
Package Type Identifier	Response
Responding Transaction ID	SSP assigned
Component Type	INVOKE (not Last): INVOKE (Last)
Invoke ID	SCP assigned
Correlation ID	Identical SSP-assigned Invoke ID from initial query
Operation Code	Connection control; no reply
Digits	Primary carrier ID
Digits	Alternate carrier ID
Digits	Second alternate carrier ID
Digits	Routing number
Originating Station Type	Global value for IN Release 0 Services
Digits	Traveling class mark
Digits	Outpulse number
Primary Trunk Group	Trunk Group number and call treatment indicator (see Table 2)
Alternate Trunk Group	Trunk Group number and call treatment indicator (see Table 2)
Second Alternate Trunk Group	Trunk Group number and call treatment indicator (see Table 2)
Billing Indicators	AMA call codes Service feature identification
Alernate Billing Indicators	AMA call codes Service feature identification
Second Alternate Billing Indicator	AMA call codes Service feature identification
Overflow Billing Indicator	AMA cal codes Service featuer identification
Digits	Billing number
Digits	Authorization code
Digits	PIN
Digits	Business Customer ID

The Component type should be INVOKE (not Last), if an INVOKE (Send Notification: Termination) Component is included in the message. The Component type should be INVOKE (Last), if otherwise.

The carrier identification and network routing number provide information essential for routing the call and must be present in the message. In addition, the alternate carrier and the second alternate carrier may be included in the message. The alternate carrier may be used by the SSP if no trunks to the primary carrier are available; and the second alternate carrier if no trunks to the alternate carrier are available.

If a call is to be routed over a specific trunk group, a Primary Trunk Group parameter must be included; two alternate trunk groups may also be included. The first octet of a Trunk Group parameter should be the call treatment indicator. The remaining four octets should contain a route index. The number to be outpulsed should be included in either the Routing Number field or the Outpulse Number field; the eighth bit of the call treatment indicator should specify which number is to be used (0 = outpulse number; 1 = routing number). If no outpulsing is required, the eighth bit of the call treatment indicator should be set to 0 and an Outpulse Number field should not be included. The seventh bit should specify if the trunk group to be used is a WATS line (0 = not WATS; 1 = WATS). The call treatment indicator should also specify the action to be taken if there are no idle trunks in a specified trunk group (see Table 2). A Billing Indicator field should be included if an AMA record is to be made when the call is routed over a specific trunk group. There is one Billing Indicator field for each trunk group. If overflow to a public network from a trunk group is permitted, the Overflow Billing Indicator field should be included in the message with the appropriate call code for overflow. Table 2 contains the possible call treatments used when no trunks are idle in a specified trunk group.

Table 2.
Call Treatment Values for Trunk Group Overflow

Treatments	Values
Not used	00000000
No overflow and no return	00000001
Overflow	00000010

Billing indicators and a billing number may be included in the message. The authorization code, dialed number, Business Customer ID or PIN may also be included if needed for billing information (as determined by the Functional Logic and subscriber data). A PIN and authorization code should not be sent together in the response message. If both are included, the SSP will ignore the first parameter and place the second parameter in the billing record. The SCP should be capable

of stripping off some of the digits of a PIN or authorization code when they are to be included in the response message.

B. First Component: INVOKE (Play Announcement) - Mandatory if call is to be terminated to an announcement.

The internal call processing (via Functional Logic and subscriber data) may instruct the network node to play an announcement and terminate a call when certain conditions such as the following occur:

- Blocking because of Resource Counter Overflow
- Call not allowed (e.g., screening restrictions, misroute)
- Disconnected number
- Changed number

The parameters and respective data elements of the response message for these conditions are as follows:

Parameters	Data Elements
Package Type Identifier	Response
Responding Transaction ID	SSP assigned
Component Type	INVOKE (Last)
Invoke ID	SCP assigned
Correlation ID	Identical SSP-assigned Invoke ID from initial query
Operation Code	Caller interaction; no reply
Customized Announcement	Announcement as specified by subscriber data

In order to have the SCP direct the SSP to play an announcement for a disconnected or changed number, a separate set of Functional Logic and subscriber data would have to be placed in the SCP. All queries to that set of logic and data could then be routed to a customized announcement chosen by the subscriber.

C. Second Component: INVOKE (NM: Automatic Call Gapping) - Optional

The parameters and respective data elements of this second Component are:

Parameters	Data Elements
Component Type	INVOKE (Last)
Invoke ID	SCP assigned
Operation Code	Network management: ACG; no reply
Digits	Service Key
ACG Control Cause Indicator	SCP Overload or SMS originated
ACG Control Duraton Indicator	Level (see Table 6 or 7)
ACG Control Gap Indicator	Level (see Table 5)

This component can be sent to the SSP as part of any type of response message. See Sections 2.2.1.4, 2.2.3 and 2.5.1.2 for a description of NM controls for the SCP.

D. Third Component: INVOKE (Send Notification: Termination) - Optional

The parameters are respective data elements of this third Component are:

Parameters	Data Elements
Component Type	INVOKE (Last)
Invoke ID	SCP assigned
Correlation ID	Identical SSP-assigned Invoke ID from initial query
Operation Code	Send notification; reply required
Echo data	SCP assigned

For IN Release 0, the Echo Data field should be used to correlate the information in the termination message with the previous query message to the call. The Echo Data field should be used for resource counter management or call sampling.

Once a request for termination information is sent, the SCP should wait a specified amount of time (**Tr**) for termination information from the SSP. If termination information is not received within the time, the SCP should decrement any resource counters associated with that call and should mark sample data associated with that call if call completion sampling data was requested. If termination information is received after the timeout period (**Tr**), it should be ignored by the SCP (see Section 2.5.2.7 for more information).

2.1.4.2. Termination Message Contents

If a response message contained an INVOKE (Send Notification: Termination) Component, the SSP generates a Unidirectional message containing the Return Result Component containing termination information. The parameters and data elements of this Component are:

Parameters	Data Elements
Package Type Identifier	Unidirectional
Component Type	RETURN RESULT (Last)
Termination NM Control List Overflow Indication	Yes/no
Termination Answer Indication	Yes/no
Termination Error Indication	Yes/no
Echo data	From SCP response message
Connect Time	Minutes; seconds; tenths of seconds
Error Code	Reason for operation failure
Problem Code	SSP assigned
Problem Data	ID and contents of Bad Data field
Standard User Error Code	Error is result of user action

The SSP sets the NM Control List Overflow to “yes” if the response message for the call included an INVOKE (NM: Automatic Call Gapping) Component that caused the SSP control list to overflow.

If the Answer field in the Termination Indicator is set to “no,” a Connect Time field is not included in the message. If the Error Indication is set to “yes,” certain error fields (e.g., Error Code, Problem Code or Problem Data) are included in the message. See *Ameritech SCP - SSP Interface Specification* for more details concerning error handling and the detailed message formats.

The Echo Data field should contain a unique ID that allows the SCP to correlate the information in the termination message with the previous query message for the call. The Echo Data field should contain information pertaining to either the Resource Counter Number used for the call or call sampling data to be collected for the call or both.

2.1.5. Error Handling

Standard TCAP error handling procedures will be used for IN Release 0 messages between the SCP and SSP. The messages used are Unidirectional or Response messages with a Return Error component, which indicates that an invoked operation was not successfully completed; a Unidirectional message with an Invoke (Operation Code: Procedural - Report Error), which indicates that a non-fatal error has been encountered; and a Unidirectional or Response message with a Reject component, which indicates that a message was not understood and cannot be processed.

Ameritech SCP - SSP Interface Specification [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] defines error procedures on the SCP-SSP interface in more detail.

2.2. Internal Call Processing Controls

This section describes SCP internal procedures and data that control feature processing.

2.2.1. Functional Logic

Functional Logic is the query processing logic defined for subscriber(s) to define a feature. Each logic representation is produced with common functionalities and general purpose data in a customized form with subscriber data. The IN Release 0 SCP should support four types of platform functionalities for the definition of IN Release 0 SCP features:

- Service key analysis platform functionalities
- Decision analysis platform functionalities
- Informational platform functionalities
- Operations, Administration and Maintenance (OA&M) platform functionalities

These functionalities should be customizable with a variety of parameters and reusable in new Functional Logic definitions.

Functional Logic can be specific to a subscriber or it can be global in nature. For instance, a global set of Functional Logic may be used for initial processing of queries, after TCAP message processing, using the service key analysis functionality and could include several OA&M functionalities.

To provision a new feature in the IN Release 0 SCP for a subscriber, the service programmer must define the type of service key and its value. Service key analysis should define the feature being used as well as the subscriber data related to it. Then, the decision analysis, informational, and OA&M functionalities are combined to provide query processing for the feature and to populate the appropriate response module with information to be sent to the SSP.

The service programmer may define the new feature and its subscriber data using SMS. Once the feature and subscriber data are defined the feature may be put into service via an administrative command.

The decision, informational, and OA&M functionalities should be able to be combined in any order to define a feature. The only limitations which should exist are as follows:

- The Functional Logic may be described as a logic tree where the nodes are functionalities. No loops are allowed in the tree. This prevents infinite loops in the Functional Logic.

- All branches (results) of a node must be connected to another node. This prevents the incomplete definition of features.

Mode nodes (or functionalities) should be able to have 0-255 possible results. Some special functionalities - like informational functionalities - by definition should have a fixed number of possible results. For example,

- Return functionality has 0 possible results

- Informational functionality has 1 possible result

- Day in Week decision analysis functionality has 7 possible results

The number of allowable results for each functionality is detailed in the following sections.

Additional Functional Logic sequences should be able to be used for query processing when necessary. In order to access the additional Functional Logic sequences, a branching or jump capability should be available, enabling one Functional Logic to point to another Functional Logic. The next Functional Logic to be used in a sequence should be dictated by the subscriber data.

The data for the Functional Logic capabilities may be supplied on a subscriber, global, or call basis. For example, specific valid PIN codes would be expected to be included in subscriber data; time of day may be in global data; caller entered data would be call data.

2.2.1.1. Service Key Analysis Platform Functionalities

Service key analysis should identify Functional Logic specific to a service or to a subscriber and the proper subscriber data.

The following should be supported as input to the service key field by the IN Release 0 SCP-SSP interface:

1. Dialed Digits (1 to 11 digits)

2. Calling Party Number (1 to 11 digits)

One result of service key analysis should be available for no match to Functional Logic and subscriber data. In the case of no match, an announcement component should be sent back to

the SSP indicating that an unassigned number announcement should be played and the appropriate performance measurements should be made. Otherwise, the SCP should continue query processing with decision analysis, informational and OA&M platform functionalities and subscriber data. If other errors are encountered, e.g., invalid data encountered, then error treatment information should be sent to the SSP.

2.2.1.2. Decision Analysis Platform Functionalities

Decision analysis and informational platform functionalities provide the Functional Logic for every SCP feature. The decision analysis functionalities are platform functionalities which provide analysis and the potential for different results on each query. They should be able to be arranged in any combination in the definition of a feature.

Following is a description of feature analysis functionalities and the number of results available per analysis that should be available for IN Release 0:

1. Time of Day Analysis - should allow decisions based on time of day, in 1 minute increments; up to 25 different results should be available.

2. Day of Week Analysis - should allow decisions based on day of week; up to 7 different results should be available.

3. Date Analysis - should allow decisions based on date, specified as dd/mm/yy; up to 25 different results should be available.

4. TCM Analysis - should allow decisions based on TCM received in initial query; up to 16 different results should be available.

5. Day Type Analysis - should allow decisions based on commonly held information on group oriented dates like bank holidays, federal government holidays, etc., specified as dd/mm/yy, up to 511 days ahead; up to 255 different results should be available. [This information should be found in common tables to allow multiple subscribers to share the same information for analysis. See Section 2.2.2.2 for more information.]

6. Time Type Analysis - should allow decisions based on commonly held information on group oriented times like opening, lunch and closing hours; up to 255 different results should be available. [This information should be found in common tables to allow multiple subscribers to share the same information for analysis. See Section 2.2.2.2 for more information.]

7. Originating LATA Analysis - should allow decisions based on LATA - 221 LATAs defined by LATA ID, up to 221 different results should be available. [This information should be

found in common tables to allow multiple subscribers to share the same information for analysis. See Section 2.2.2.2 for more information.]

8. Origin Analysis - commonly held information to allow decisions based on either 2 or 3 digits of the ANI, up to 255 different results should be available. [This information should be found in common tables to allow multiple subscribers to share the same information for analysis. See Section 2.2.2.2 for more information.]

9. Number Analysis (ANI, Dialed, User Interaction or Service Key digits) - should allow decisions based on one digit of the calling party, dialed (called) number, User Interaction digits or Service Key digits; up to 10 different results should be available.

10. Percentage Distribution - should allow alternating result possibilities for each query. It should specify a percentage of queries to each of the possible results. Up to 100 different results should be available.

11. Call Limiter Analysis - should allow analysis based on the usage within a specified duration, e.g., 100 queries within 10 seconds. Increments should be whole numbers for usage and 1 second increments for duration. This analysis should have a time based result (in 1 second increments) with two results available (one for usage above a subscriber-specified threshold and one for usage below a subscriber-specified threshold.)

12. Resource Usage Analysis - should allow analysis based on a resource counter value. If a resource counter is less than a threshold specified in subscriber data, the SCP should increment the resource counter and the SCP should then include a request for termination notification in the appropriate response message. Queries should continue to be assigned to the resource until the counter reaches a maximum. The counter should be updated by the SCP using termination results from the SSPs (see Section 2.5.2.6.)

13. Caller Interaction Information Collection - should allow SCP to request SSP to prompt for and collect n digits (n from 1 to 15 digits) of caller entered information. It should allow the returned information to be labeled as one of the following: ANI, dialed digits, PIN, User Interaction digits or authorization code, depending upon the needs of the particular Functional Logic and subscriber data. The collected information should then be available for use in further Functional Logic analysis or for use in the appropriate field of a response message. One result should be available for a correct number of digits entered and two error results should be available: one for incorrect number of digits (if request is for a fixed number of digits) and one for no digits entered.

14. PIN Collection and Analysis - should allow for the collection of a fixed number of digits of a PIN from a caller. The PIN should be compared with the valid digits stored in the subscriber data. It should allow for the caller to be requested to reenter digits 1-n times. Three results should be possible: one for correct digits entered, one for incorrect digits entered with no retries remaining, and one for caller abandon.

15. Screening Analysis - should allow the ANI, dialed (called) digits, PIN or Authorization code to be compared against a list of five numbers (of length 1-15 digits.) The analysis should allow each match to have a different result and two results should be available for errors: one result for no match and one result for the condition of no number being available for screening.

16. Intermediate Result Analysis - should allow decisions based on a value stored as an intermediate result; up to 255 different results should be available. Note: All functionalities can fill their results as intermediate results. A number of intermediate result storage locations should be available.

17. Load Distribution Analysis - should allow a uniform distribution of queries across multiple results, where a resource counter is associated with each result. The analysis should allow for hunting for an available resource, beginning with the next resource after the last one that was free. If a resource is available, the resource counter should be incremented and a request for termination notification should be sent with the response message containing routing instructions. One result should be available for the case of no resources available. The counters should be updated by the SCP using termination results from the SSP (see Section 2.5.2.7 for more information on Resource Counter Management).

These functionalities by themselves provide very basic features. But they may also be combined to provide what has been defined as IN Release 0 SCP features, e.g., call distribution by geographic area, alternate host routing, etc. Together with the informational and OA&M platform functionalities, the decision analysis platform functionalities define the disposition of IN Release 0 calls.

2.2.1.3. Informational Platform Functionalities

Informational platform functionalities should provide subscriber based data to the SSPs for IN Release 0 features. This information should be contained in the SCP conversation or response message components to the SSP: Invoke (Connection Control), Invoke (Play Announcement), Invoke (Play announcement and Collect Digits), Invoke (Automatic Call Gapping), and Invoke (Send Notification: Termination).

These functionalities should allow a specified type of data, e.g., subscriber, call or common data, to be set aside for later use in a response or conversation message. During query processing or after the Functional Logic has finished processing the query, a response may be sent to the SSP containing the specified data as needed. Note that there can be response data which may not be sent. For example, billing indicators may be set aside for use in a response message but may never be sent due to the occurrence of an error.

The informational platform functionalities should be able to be included with the decision analysis platform functionalities in any order. If placed at the beginning they should be available to be sent in the response or conversation message regardless of the results of following decision analyses. If they are included after a decision analysis functionality then they should specify the appropriate information based on the preceding analysis. If they are the last functionality in query analysis, then they should specify the final treatment of the call to be included in the response message.

Informational functionalities should have one possible result. A special informational functionality with zero results is a Return functionality. The Return functionality should end the Functional Logic, and thus query processing, and instruct the SCP to send the specific information to the SSP in a TCAP response message.

All informational functionalities should have one possible result. This is what distinguishes them from decision functionalities, which have two or more results.

The set of informational functionalities which should be supported by an IN Release 0 SCP for use in the TCAP response or conversation messages is listed below.

A. The following should be available for inclusion in a conversation or response message and should be available from subscriber data based on analysis:

- Primary Carrier ID - loading of the 3 digit carrier identification.
- Alternate Carrier ID - 3 digits
- Second Alternate Carrier ID - 3 digits
- Routing Number - loading of the 10 to 28 digit routing number. (See note below.)
- Outpulse Number - 0 to 15 digits
- Primary Trunk Group - 8 digits; 1 octet CTI, 8 digit trunk group number.

- Alternate Trunk Group - 8 digits; 1 octet CTI, 8 digit trunk group number.
- Second Alternate Trunk Group - 8 digits; 1 octet CTI, 8 digit trunk group number.
- Billing Indicator - AMA call type (3 digits) and Service Feature Identification (3 digits.)
- Alternate Billing Indicator - AMA call type (3 digits) and Service Feature Identification (3 digits.)
- Second Alternate Billing Indicator - AMA call type (3 digits) and Service Feature Identification (3 digits.)
- Overflow Billing Indicators - AMA call type (3 digits) and Service Feature Identification (3 digits.)
- Billing Number - loading of a 10 digit alternate billing number.
- Business Customer ID
- Customized Announcement ID - load the customized announcement ID between 1 and 255.
- Number of Digits - load the number of digits expected by the SCP.
- ACG Indicators - load control cause indicator, duration and gap level, number of digits under control. For cause indicator, duration and gap values, see Section 2.2.3 and Section 2.5.1.

B. The following should be available for possible inclusion in a response message and should be available from caller input:

- PIN - variable length
- Authorization code - variable length

C. The following should be available for inclusion in a response message and should be available, as TCAP echo, from the initial query:

- Traveling Class Mark - 1 or 2 digits, value 0-15

- Originating Station Type - 1 octet (See note below.)

NOTE: The Routing Number parameter may also be available from the Dialed (called) Digits parameter from the initial query. The Originating Station Type parameter may also be an overwrite to a value assigned to all IN Release 0 services.

The parameters which may be contained in each informational platform functionality for Release 0 are bounded by the fields supported for SCP responses to SSPs (see *Ameritech SCP-SSP Interface Specification* [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] for additional details on the parameter values).

2.2.1.4. Operations, Administration and Maintenance (OA&M) Platform Functionalities

OA&M platform functionalities provide the service programmer the statistics, sampling and network and resource management support functionality needed in the definition and provisioning of new services. Some functionalities are independent functionalities that can be used in Functional Logic while others are combinations of several other functionalities that have been previously discussed. These functionalities may be used just as any other platform functionality. They should be able to be included in the Functional logic in any order and with a variety of parameters.

The OA&M functionalities that should be available for IN Release 0 are as follows:

- Call Gapping
- Sampling
- Special Studies
- Statistics

2.2.1.4.1 Call Gapping

The Call Gapping functionality should allow network management controls to be placed on a subscriber basis. It should allow a limit to the number of queries through a branch of Functional Logic during a defined time period to be specified. If the limit is exceeded within the time period, call gapping should be automatically initiated on the next query passing through that functional logic branch (by sending an ACG component with the response message), based on a number of digits of the service key (fixed at 3, 6, 7, 8, 9, or 10 digits) as specified in the subscriber data. The maximum observation time should be 30 minutes. The maximum number of attempts should be 10,000.

In addition to specifying the number of digits of the service key to be placed under control, the gap level and duration level must be specified. The subscriber should specify the gap interval level (1-15, see Table 5) and the duration interval level (1-13, see Table 4) to be included in the ACG request. The control cause indicator should be set to “SOCC.” The SSP will use the SOCC control cause indicator to route blocked calls to 60 ipm.

2.2.1.4.2 Sampling

A sampling functionality may be needed in any location of the Functional Logic. For sampling, the sampling functionality should specify the percentage and the information to be collected on the query and/or termination result.

Sample data should be able to be obtained on subscriber basis. In order to obtain sample data, the functionality for sampling must already exist in the Functional Logic. Requests should be implemented by updating the sampling parameters in the subscriber data. The request from SMS must include a Sampling ID. The Sampling ID should include all the information necessary to uniquely define the sampling to be done. Sample data can be based on either call attempt or call completion. Following is the data that should be collected:

Part 1:

The following data should be collected for both call attempt and call completion samples.

A. Sampling ID

B. Service Key

C. Time

D. Sample rate

E. LATA

F. ANI (calling number)

G. Dialed number

H. Routing number

I. Authorization code

J. PIN

K. Outpulse number

L. Traveling class mark

M. Carrier ID

N. Final disposition

1. Announcement ID number
2. Returned routing number
3. Functional Logic error.

In addition, for call completion samples, a sample number should be included with the above information.

Part 2:

The following information should be collected for call completion samples only, where the SCP returned a routing number.

- Sample number
- Answer indication
- Connect time
- Error code.

The sample number should be used by the SMS to correlate the call attempt data with the call completion data.

If the SCP receives no termination information from the SSP within a specified time (**Tr,** where **Tr** should be telco assignable), the sampling information should not be lost but should be marked to indicate the error.

The SCP should gather sample data and store it: it should not collate call completion sample data. Data should be gathered into files for transfer to the SMS.

2.2.1.4.3 Special Studies

It should be possible to perform special studies by the IN Release 0 SCP. The special studies functionality is a temporary call monitor requested by SMS that produces reports when calls

meet certain trap criteria. The Special Studies functionality must be present in the Functional Logic to perform special studies.

The contents of an SMS Special Study request are:

A. Time limit

B. Trap limit

C. Special Studies ID

D. Trap criteria - The following may be trap criteria:

1. Service Key
2. Dialed number
3. ANI (calling number 0
4. Primary Carrier ID
5. Primary trunk group
6. Routing number
7. Authorization code
8. PIN
9. Tone/Arrangements
10. Outpulse Number

The time limit set an upper limit on the life of the study. The study should also end if the number of calls trapped meets the trap limit. The selection possibilities for data collection are the same as in sampling (see Section 2.2.1.4.2.) The SMS can issue a command to cancel a Special Study.

2.2.1.4.4 Statistics

There are two types of statistical functionalities which should be definable in the Functional Logic. First, a counter with a threshold should be defined which may be placed at any location in the functional logic. On reaching the threshold, or at a specified time, the counter information should be sent to the system statistical function for output. Second, a counter with termination notification information which may be placed anywhere in the functional logic sequence should

be available. This functionality should collect termination notification information along with counter information. The information on threshold or timer should be available for output to SMS.

2.2.1.5. Platform Functionality Variables

Data or variables should be supported for the SCP Functional Logic on a subscriber, common or call basis. Following is a description of the data types and some of the values that should be supported.

Subscriber data should be used by the Functional Logic to provide the subscriber specific values in query processing. Subscriber data should be contained within a subscriber data unit and defined for use only for a specific subscriber. This data should specify the following information:

- Conditions and values for Decision Analysis Functionalities
- Values for Information Functionalities
- Values for OA&M Functionalities

Common data may be used by a variety of subscribers and functional logic in query processing. Typical examples of common data are time of day or nodal statistics parameters. For more information on common data, see Section 2.2.2.1.

Call data should be populated during query processing. The data is either present in the incoming query from the SSP, generated during query processing, or returned to the SCP during call processing. For example, the response data from informational functionalities should be stored as call data before the response is sent to the SSP. Following is the call information that should be available to the SCP from the SSP messages:

- Service Key - Type and Value
- Originating LATA
- Originating Station DN
- Dialed Digits or Calling Number
- PIN
- Authorization code
- Traveling Class Mark

- Termination Indicators
- Caller interaction
- Connect Time
- Standard User Error Code

The actual values for the call data are specified in *Ameritech SCP-SSP Interface Specification*. [Any changes to, additions to or modifications of *Service Control Point Node Generic Requirements* are included in Appendix B of this document.] See this document for allowable data values.

The informational and service key analysis functionalities are also directly related to the TCAP signaling interface and are therefore bounded by that specification. Other data values, for example, those which populate the subscriber and functional logic data, are functions of the SCP's decision analysis and OA&M functionalities. Section 2.2.1.4 contains specific information on the OA&M platform functionalities and Section 2.2.1.2 contains specific information on the decision analysis functionalities.

2.2.1.6. Query Processing Routine Errors

The SCP should monitor for processing errors in the SCP Functional Logic. This may include monitoring for infinite loops in the Functional Logic, lack of response indication information at the end of a logic branch, and any other mechanisms necessary. These monitors or controls depend on the representation of the logic in the SCP. Therefore, the vendor should provide adequate checks based on their particular implementation. If an infinite loop is detected in the query processing logic, the feature should be taken out of service and an error message sent to SMS where an Exception Report will be generated. For any other logic error encountered, the feature need not be taken out of service, but an error message should be generated.

2.2.2. Common Procedures and Data

The SCP can use the data described in this section in query processing. This information resides in the SCP such that it is available to the Functional Logic at any point in the query processing.

2.2.2.1. General Purpose Data

General Purpose data may be included in each the Functional Logic script or may be "called" from a common data location. While it is not required that frequently/commonly used data reside in a common data location, such a data structure may realize efficiencies in system design

as well as from the service programmers perspective. Information to be contained in a common data location is the following:

- time of day
- day of week
- date

2.2.2.2. Common Procedures

Common procedures are those which may be shared among many or all IN Release 0 SCP features. These procedures may be defined for each feature or can be included in a common set which may be accessed when required by the Functional Logic. Common procedures for query processing should include the following:

- Day Type (bank holidays, federal government holidays) Analysis
- Time Type (bank hours, retail store hours) Analysis
- Origin Analysis (based on two or three digits of calling party's number)
- Originating LATA Analysis

See Section 2.2.1.2 for more information.

Common procedures that are executed only if specified by the Functional Logic and subscriber data or if requested by SMS are the following:

- Counters for statistical purposes, special studies, routing statistics, etc. (See Section 2.2.1.4.5 for more information.)
- Collection of traffic and performance measurements (See Section 2.5.3.2 for more information.)
- Network management controls on a subscriber level. (See Section 2.5.1 for more information.)

Some of these procedures/functions can be considered node functions while others may relate specifically to features but are common across many different features. Any of the platform functionalities previously described could also be considered common procedures in the case where multiple subscribers use the same data (e.g., network resource counters.)

2.2.2.3. Timer Expiration Values

All TA (Transaction Age) timer values for the SCP discussed in this section should be recent changeable on an office by office basis from an external system (SMS.) The TA timers are described below and nominal value are given.

A. The SCP should ensure that a conversation or response message is sent in response to queries from the SSP within a specified amount of time. The nominal value for that time is 3 seconds.

B. A TA timer expiration value should be used by the SCP to time SSP response to SCP queries involving caller interaction (see Section 2.1.3.3A). The nominal value for this timer is 5 minutes.

C. A timer, **Tr**, should be used by the SCP to time SSP responses to SCP requests for termination information (see Section 2.1.4.1D). The value of this timer should be telco-assignable.

Performance measurements on the number of TA timer expirations should be maintained by the SCP (see Section 2.5.3.2).

2.2.3. Automatic Network Management Controls for SCP Overloads.

When the calling rate for SCP IN Release 0 services becomes excessively high, the SCP can be driven into an overload condition. To protect the SCP from the effects of overloads, an automatic NM control is required, which will be called the SCP overload control. The SCP overload control should operate as follows. The SCP should detect overloads by measuring query processing loads, based on a measure that is an indication of query processing delay, on the system at a node level. When the SCP determines it is in overload, certain actions should be taken to help alleviate the overload. The actions are a function of the level of overload.

When the SCP initially enters an overload condition, its first actions should be to process higher priority messages. If the overload condition continues to exist, the SCP should begin sending INVOKE (NM: ACG) Components in response messages that contain routing instructions or that instruct the SSP to play an announcement. If the above actions are not sufficient to alleviate the overload condition, the SCP should begin dropping incoming messages.

When the SCP begins returning ACG requests to the SSP, all incoming queries to the SCP should result in the SCP sending an ACG command. The ACG command should include the first six digits of the service key on which control is to be initiated, the gap interval level and the duration interval level. There are 16 possible gap interval levels (see Table 3) and 13 possible

duration levels (see Table 4). To remove a code from the control list at the SSP, Gap interval level 16 and Duration level 0 should be used.

Table 3.
Gap Interval Levels

Gap Interval Level	Nominal Interval (seconds)
0	0
1	3
2	4
3	6
4	8
5	11
6	16
7	22
8	30
9	42
10	58
11	81
12	112
13	156
14	217
15	300

The ACG command instructs the SSP to limit traffic contributing to the overload. The ACG request should instruct the SSP to cut back future messages from originating or terminating IN Release 0 callers with the same first six digits as the service key of the current query. The ACG request should also contain the gap interval and duration maintained by the SCP. The control cause indicator in the ACG should be set to indicate a SCP overload. The SSP will use this control cause indicator to route blocked calls to No Circuit Announcement (NCA).

In addition to the SCP overload requirements specified in this section, NM requirements are also specified in three other sections of this document, Sections 2.2.1.4, 2.5.1 and 2.5.4. Section 2.2.1.4 contains details on the Call Gapping OA&M functionality available to subscribers in their Functional Logic. Section 2.5.1 contains the requirements for SOCC controls. Section

2.5.4 contains the requirements for performing special studies. The concept of a special study was developed for service maintenance, but it can also be a useful tool for NM.

Table 4.
Duration Level

Duration Level	Nominal Control Duration (seconds)
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128
9	256
10	512
11	1024
12	2048
13	infinity

2.2.4. Failure Recovery

The SCP may experience varying levels of failure. After any restart, any queries which were being processed are lost. The SSP will time out and provide error treatment for the call. Resource counters should maintain their pre-failure values. Functional Logic, subscriber data and all common data described in Section 2.2.2 should not be affected by the failure.

2.3. Signaling

Not applicable.

2.4. Transmission

Not applicable.

2.5. Administration

2.5.1. Network Management

The SCP should contain automatic NM controls to alleviate the effects of SCP overloads on a feature or subscriber basis.

The NM strategy for IN Release 0 services is different from the other new services, e.g., 800 Service and ABS. The NM strategy for IN Release 0 is based on controlling on originating OR terminating calling.

A primary reason for this is that routing of database queries for IN Release 0 is done based on the service key, which may be a calling number (e.g., in the case of a subscribed trigger) or a called number (e.g., in the case of a Directory Number trigger.) [A description of these triggers is given in the *Ameritech SSP Functionality Specification*.]

The administration of the automatic control parameters in the OA&M Call Gapping functionality, activation and deactivation of the manual control, and obtaining Exception and Scheduled Reports will all be handled by SMS. The SMS functions as an intelligent conduit between the SCP and the network manager. The SCP should provide the following functions for SMS:

- Change SCP automatic control parameters for Functional Logic and subscriber data on request (see Section 2.5.1.1)
- Activate manual control in the SCP on request (see Section 2.5.1.2)
- Send Special Study Reports to SMS on request (see Section 2.5.4.2).
- Send Exception Reports to SMS.

2.5.1.1. Automatic Control Parameters

Two types of automatic overload controls exist in the SCP, those that are present in the Functional Logic and the subscriber data using the Call Gapping OA&M functionality and those that provide a system level overload control called SCP overload control. The Call Gapping OA&M Functionality is discussed further in Section 2.2.1.4. The SCP overload control is discussed further in Section 2.2.3.

The gap levels and duration levels used in the Call Gapping functionalities can be loaded from the SMS. The network manager, via the SMS, should be able to replace new values of gap interval and/or duration level. Those values can be assigned from Tables 4 and 5 as described in Section 2.2.1.4.

The Release 0, the SMS will not be able to change the values of gap and duration levels that are used in the INVOKE (NM: ACG) Component that is sent as a result of system level overloads.

2.5.1.2. SMS Originated Code Control (SOCC)

The SCP should provide a SMS Originated Code Control (SOCC) for NM use. With SOCC, the network manager, via SMS, should be able to control calls based on the service key, either an originating number or a dialed number. It should be possible for the network manager to place a 3, 6, 7, 8, 9, or 10-digit code under manual control. The controlled code should have the form NPA, NPA-NXX, NPA-NXX-X, NPA-NXX-XX, NPA-NXX-XXX, and NPA-NXX-XXXX. The service key is the code to be controlled (e.g., NPA-NXX-X). SMS specified the gap interval level (1-15, see Table 5), the control duration (1-13, see Table 4), the control cause indicator and the number of service key digits under control. The network manager will use Table 5 to select the desired gap interval level (e.g., 1), and will use this level in the SOCC activate message to SMS. SMS will send the selected value from Table 5 along with the other control parameters, e.g., code, to SCP. The SCP should use the gap interval level for generating ACGs that it sends to the SSP. To remove a code from the control list, Gap Interval level 0 and Duration level 0 should be used. To deactivate a manual control, SMS will specify the code to be removed. Note that the SOCC control uses a different set of gap intervals from the SCP overload control. Because SOCC is a manual control and it can be applied to up to 10 digits, it is necessary to have a finer set of gap interval levels in the small gap interval range to afford the network manager greater flexibility in choosing the proper gap interval.

Table 5.
NM SOCC Control Gap Intervals

Gap Interval Level	Average Gap Interval (seconds)
0	Remove Code from Control List
1	0.0
2	0.1
3	0.25
4	0.5
5	1.0
6	2.0
7	5.0
8	10.0

9	15.0
10	30.0
11	60.0
12	120.0
13	300.0
14	600.0
15	infinity

NOTE: The above provides the control level field values and the corresponding average gap intervals for SCP to SSP communication.

A network manager can put a zero gap interval SOCC on a code. Although this should not limit the calling rate from the given code, it should cause counts to be accumulated (on a 5-minute basis) for calls from the code. These counts should be available to the network manager as a scheduled report (see Section 2.5.1.3), and should provide a valuable surveillance functionality, e.g., to detect excessive calling to/from certain numbers or groups of numbers.

The network manager should also be able to override code controls using the zero gap interval. For example, assume a code control is on a given 6-digit control, e.g., 201-758. Then, if the network manager places a zero gap interval SOCC on, e.g., 201-758-2, the 2000 series of lines in the 758 NXX will be removed from control at the SSP. This will occur because for control activation purposes, the SSP will first look at longer codes. The network manager, via SMS, would send the selected code and control values to the SCP and the SCP would then send an ACG component to the SSP to initiate the control.

When a manual control is received in the SCP, from the SMS, every query received in the SCP that contains the controlled code as the service key should result in the SCP sending an ACG message to the SSP that originated the query. The ACG should contain the controlled code, the gap interval level index, the duration level, and a control cause indicator. The control cause indicator should be set to SMS initiated. The SSP will use this control cause indicator to route blocked calls to 60 ipm.

When the SCP receives a command from SMS to deactivate a manual control, the specified code should be removed from the manual control in the SCP. The next query from an SSP for this code should cause the SCP to return an indication to the SSP to remove the code from control.

2.5.1.3. Error Messages and Special Study Reports

A. Error Messages

The SCP will send an error messages to the SMS indicating a change in SCP overload level.

B. Special Study Reports

As part of the service maintenance functionalities, it is required that the SCP have a Special Study Report functionality. The requirements for special studies are given in Section 2.5.4.2. The special study functionality is a temporary call monitor that produces reports when calls meet certain trap criteria (see Section 2.5.4.1). Although the special study concept was primarily developed for service maintenance, it also provides a very useful tool for network management. Therefore, it is required that network managers, via SMS, have access to the special study functionality covered in Section 2.5.4.1.

2.5.2. Functional Logic and Subscriber Data Administration

This section describes administration of subscriber-specific data: Functional Logic, subscriber sample data, and Resource Usage Counters.

2.5.2.1. Functional Logic and Subscriber Data

The same Functional Logic should be able to be chosen by one or more subscribers. To perform a certain service, the subscriber should provide the specific data that is needed by the functionalities in the Functional Logic. For example, the time-result combinations are required data for population of the time of day functionality.

In addition to the active Functional Logic and the active subscriber data, sets of pending Functional Logic and pending subscriber data can be maintained to allow the activations without disrupting the service to a customer.

2.5.2.2. Functional Logic Commands

The administrative commands to introduce new Functional Logic should contain the following information:

- Functional Logic name
- Type of command (general introduction and delete commands, retrieve commands, etc.)

Before the Functional Logic should be accepted in the SCP, a check for completeness should be made. (For example, no capabilities with open results or loops should be allowed.)

2.5.2.3. Subscriber Record Commands

The administrative commands to introduce or remove a subscriber should contain the following information:

- Subscriber name
- Command type (introduce/remove a subscriber, connect/disconnect a Functional Logic to a subscriber)

2.5.2.4. Subscriber Data Commands

The administrative commands to introduce, update or delete subscriber data should contain the following information:

A. Subscriber name (not used for common data)

B. Capability type

C. Capability sequence number

D. Command type (enter/modify/update/delete/connect)

One command affects the data of one capability.

2.5.2.5. Subscriber Retrieve Operations

Both the general subscriber properties (name and Functional Logic name) can be used by the subscriber to retrieve the subscriber data. The commands should provide the possibility to retrieve the following information:

- the specific data for one capability of a subscriber
- all data for one type of capability of a subscriber
- all data of a subscriber.

2.5.2.6. Sampling Data and Special Studies

A subscriber can request sampling data or special study reports by changing the data of the respective capability. A specific subset of the data mentioned in 2.2.1.4.3 can be selected.

2.5.2.7. Resource Counter Management

The SCP should ensure that the accuracy of resource counters be maintained. If query processing causes a counter to be incremented, the SCP should include an INVOKE (Send Notification: Termination) Component in a response message containing routing instructions. If multiple counters are incremented, only one termination notification request should be sent. If, after further processing of a query, routing instructions are not sent as final treatment (e.g., the SCP instructs the SSP to play an announcement), the SCP should decrement all the appropriate resource counters.

The SCP should wait a specific time (**Tr**) for the Unidirectional message containing termination information from the SSP. If the termination information is received within time (**Tr**), the SCP should use the echo data to decrement the appropriate resource counters. After the SCP time out period (**Tr**) for waiting for the SSPs termination notification, the involved resource counters should be decremented. The value for **Tr** should be determined by the administrator.

The limits for the resource usage counters can be changed by recent change and update commands.

2.5.3. Common Procedures Data Updates

This section covers the administration of common SCP data.

2.5.3.1. General Purpose Data Updates

Table entries are added or deleted with an Update Table command. A Retrieve Table command returns all entries in the table. Update and Retrieve commands are available for the common data tables discussed in Section 2.2.2.

2.5.3.2. Traffic and Performance Measurements

The SCP should maintain the following measurements (peg counts) on CCS messages and query processing:

1. Query messages requesting IN Release 0 service
2. Call not allowed because no match of service key to Functional Logic and subscriber data
3. Incoming messages dropped because of an overload condition
4. Query messages with error in data

5. Query messages not responded to because the SCP timer expired for a SCP response
6. Caller interaction messages sent to collect digits
7. VCA responses sent because of Functional Logic error
8. SCP TA timer expired for a response from the SSP (caller interaction)
9. Response messages sent with routing instructions
10. Requests for a termination message to be sent
11. Termination messages received
12. Error messages received with termination indicators
13. Error messages received with termination indicators caused by caller abandon or SSP failure
14. Error messages received without termination indicators

These measurements, should be maintained in 30-minutes intervals. The measurements, along with the date and time at which the interval ended, should be stored along with other measurements, detailed in *Service Control Point Node Generic Requirements* [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.], that are maintained for IN Release 0.

The SCP is also capable of maintaining traffic and performance measurements (see Section 2.2.1.4) for a particular subscriber. These additional measurement capabilities are dictated by the individual sets of Functional Logic and subscriber data.

2.5.3.3. Service Sample

To the SCP, a “service” is a collection of Functional Logic and subscriber data. In order to achieve a service sample, SMS must then indicate which Functional Logic (via a Functional Logic ID) and subscriber data (via the service keys) for which the SCP must sample. Sampling for IN Release 0 services should be controlled by start and stop messages from the SMS. The SMS should maintain a list of the IN Release 0 services, the customers that subscribe to each and the service keys associated with those subscribers. The start message would then indicate to the SCP which service keys to sample on in order to complete a service sample. Requested service sample data can be based on either call attempt or call completion information. The start message should include the following information:

- the sample rate
- sample type
- Sampling ID
- the Service Key(s) or Functional Logic on which the SCP should sample

The particular sample rate should be engineered such that it does not cause SCP overload conditions. The sampling identification should consist of the following information: subscriber name, type of functionality, sequence number of the functionality.

As with the subscriber sampling, the functionality for sampling must already be in place in the Functional Logic in order to achieve a service sample. Further information concerning the data collected for a sample is covered in Section 2.2.1.4.

The SCP should gather the service sample data and store it: it should not collate call completion sample data. Data should be gathered into files for transfer to SMS.

2.5.4. Service Maintenance

The SCP should provide a monitoring function with Exception Reports and an analysis function with Special Studies.

2.5.4.1. Error Messages

The SCP should send error messages to the SMS when the following events occur:

A. Functional Logic errors

B. Service Key in query message does not match Functional Logic and subscriber data

C. Network management condition (see Section 2.5.1.3A).

2.5.4.2. Special Studies

A Special Study is a temporary call monitor that produces reports for SMS for calls meeting trap criteria. The Special Studies functionality must already be in place in the functional logic in order for SMS to request a Special Study. Details of Special Studies are included under OA&M functionalities in Section 2.2.1.4.

2.5.5. Administrative Data Transfer Procedures

The SCP and SMS exchange three classes of administrative data:

- Traffic and performance measurements
- Customer and network sample data
- Control data

The sections that follow describe the high-level data flows and procedures for each class.

2.5.5.1. Measurements

The SCP must be able to support the capability to maintain traffic and performance measurements at 30 minutes intervals. If the SCP fails or reinitializes, measurements for the current 30-minute period are lost. If the SCP enters an overload state, measurement collection is unaffected.

The SCP combines the IN Release 0-specific measurements with other common measurements it has taken in the same time interval and stores them for later dispatch to SMS.

2.5.5.2. Sample Data

The SCP should collect customer sample and network sample data for the services supported by the SCP and store them in separate files for the SMS. The SCP will generate a system alarm when threshold for remaining disk space is reached.

2.5.5.3. Control Data

The SCP processes three types of control data:

- SMS commands
- SCP responses
- SCP reports.

SMS commands are SMS messages sent to update or retrieve SCP administrative data. SCP responses should be sent to each SMS command. An SCP report should be sent in response to some event other than the receipt of an SMS command. The SCP report messages should be either error messages or Special Study reports.

2.6. Maintenance

In addition to functions detailed in the *Service Control Point Node Generic Requirements* [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.], the Release 0 SCP should have the following functions available to the SCP maintenance terminal.

2.6.1. Traffic and Performance Measurements

A. Error messages received with termination indications caused by SSP failure

B. Error messages received because of unexpected data

C. Error messages received because of a protocol error

D. Error messages sent because of unexpected data

E. Incoming messages dropped because of an SCP overload

F. Misrouted messages (no Functional Logic located)

2.6.2. Reports

The maintenance terminal receives automatic Daily Reports and requested Day-to-Hour Reports on the measurements listed in Section 2.6.1 (see the *Service Control Point Node Generic Requirements* for a detailed description of these reports).

2.6.3. Controls

Commands for setting traffic and performance measurement maximums specified in this section can be entered from the maintenance terminal.

2.6.4. Backup and Recovery

The SCP has its own facilities for backing up SCP data without SMS intervention. However, if larger amounts of Functional Logic or subscriber data are lost, the SCP should be able to accept a magnetic tape of Functional Logic and subscriber data updates written by SMS.

The SCP can also selectively back up Functional Logic and all data during feature modifications. As part of modification of a feature, a time can be specified for old logic and data to be saved. The time period allowed should be up to 72 hours. This is specified via an administrative command. See *Service Control Point Node Generic Requirements* [The IN Release 0 SCP

will be referred to as the SCP throughout the remainder of this document.] for more details on the SCP backup and recovery.

2.7. Performance

2.7.1. Availability

The total downtime objective for services offered by the SCP is 3 minutes per year for all faults caused by hardware, software, and procedural errors. Since each SCP will be deployed in pairs in a mated configuration, a downtime objective of 20.9 hours per year for an individual SCP meets these objectives.

A mated configuration is one in which a SCP application is deployed at two (normally geographically separate) SCP nodes.

The supplier should follow the requirements specified in Section 12.5 of the *LATA Switching System Generic Requirements (LSSGR)* [AN SCP feature can include one or more of the IN Release 0 Features as defined by Ameritech.] for a duration of downtime and for guidelines on the allocation of downtime between faults caused by hardware error and faults caused by software and procedural errors.

2.7.2. Network Response Time

Network response time for the SCP is defined as the interval beginning at the receipt of a query message and ending at the transmission of a response message. During normal conditions, the mean response time should be less than 1.0 seconds and should not exceed 1.5 seconds for 99 percent of all messages. For the 10-second period after a SCP first assumes the load of its mate, the mean response time should be less than 1.5 seconds and should not exceed 2.0 seconds for 99 percent of all messages.

2.7.3. Administrative Response Time

The administrative response time for the SCP is defined as the interval beginning from the receipt of an SMS command to the transmission of an acknowledgment from the SCP. The mean response time objective should be in the 4-to-8 second range, depending on the type and size of the SMS command message.

2.7.4. Query Processing Capacity

To allow for expected growth (see Section 2.9.2), an SCP should be able to process at least 200 transactions per second. A transaction is the processing of a message from the SSP through the time a response is sent to the SCP. There may be many transactions in the process of completing an IN Release 0 call.

2.7.5. Administrative Capacity

Details on Administrative Capacity for the IN Release 0 SCP will be provided by Ameritech.

2.8. Interactions

Not applicable.

2.9. Limitations and Restrictions

Details on Limitations and Restrictions of the IN Release 0 SCP will be provided by Ameritech.

2.9.1. Functional Logic Limitations

2.9.2. Resource Counters

2.10. Timing and Tolerances

Not applicable.

3. List of Acronyms

AMA	Automatic Message Accounting
ACG	Automatic Call Gap
ANI	Automatic Number Identification
CCS	Common Channel Signaling
EAEO	Equal Access End Office
IC	Interexchange Carrier
INC	International Exchange Carrier
LATA	Local Access and Transport Area
NCA	No Circuit Announcement
NPA	Numbering Plan Area
NM	Network Management

OA&M	Operations, Administration and Maintenance
PIN	Personal Identification Number
SCCP	Signaling Connection Control Part
SCP	Service Control Point
SOCC	SMS Originated Code Control
SMS	Service Management System
SS7	Signaling System 7
SSN	Subsystem Number
SSP	Service Switching Point
STP	Signal Transfer Point
TA	Transaction Age
TCAP	Transaction Capabilities Application Part
VCA	Vacant Code Announcement

4. References

1. *Service Control Point Node Generic Requirements*, Issue 2, TA-TSY-000039. Telcordia (formerly Bellcore). December, 1986.

2. *Ameritech Service Control Point - Service Switching Point Interface Specification.* Issue 1, AM TR-OAT-000044. July, 1989.

3. *Ameritech Service Switching Point Functional Specification.* Issue 1, AM TR-OAT-000042. Ameritech. July, 1989.

4. *LATA Switching System Generic Requirements (LSSGR).* Issue 2, TR-TSY-000064. Telcordia (formerly Bellcore). July, 1987.

APPENDIX A - Appendix A: A Feature Definition Example

This section describes how a service programmer might use the IN Release 0 SCP platform functionalities to provide a feature. The following describes the Functional Logic and subscriber data needed in the SCP to provide the Access Authorization Verification feature.

Access Authorization Verification is a feature which verifies a caller entered code and if correct, routes the caller to the desired destination. If, the caller entered code is incorrect, the caller can reenter the code a specified number of times before being played an announcement and disconnected. To deploy this feature in an IN Release 0 environment, the SCP would be responsible for performing the following functionalities:

1. service key analysis (service key = dialed digits - 10 digits)
2. OA&M functionality (sampling)
3. decision analysis functionality (PIN collection and analysis)
4. OA&M functionality (call gapping)
5. informational functionality (routing number, billing indicator)
6. informational functionality (announcement ID)
7. OA&M functionality (statistical counter)

See Figure A-2 for a logic representation of the feature.

These functionalities structured in the manner described above provide a feature skeleton - i.e., it lacks the subscriber data to perform in the network. An example of the subscriber data which could be used with this skeleton is the following (matching the capabilities used above):

1. dialed digits = 214–904–1000
2. sample rate = .02; sample based on call attempt
3. number of digits = 4

screening list = 1022, 1023, 5022, 5023, and 9800

announcement ID = 24

number of retries: max = 3

4. call limit = 100, time = 10 seconds, duration = 30 seconds

gap interval level = 3, duration interval level = 4, # digits to control = 7

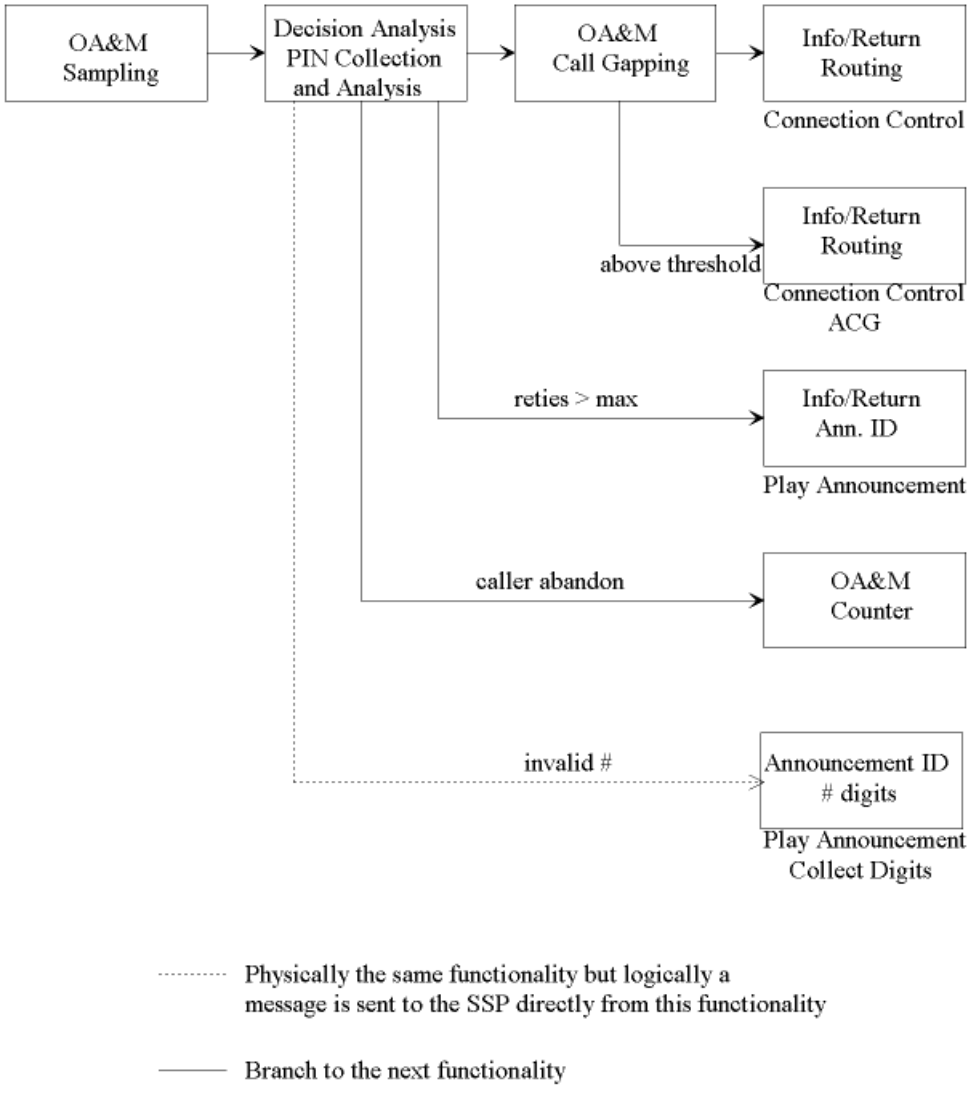
5. routing number = 214-997-1234, billing indicator = AMA call type and Service Feature ID

6. announcement ID = 25

7. counter threshold = 20

This data, together with the Functional Logic, provides the IN Release 0 feature in the SCP.

Figure A-2. Logical Representaion of Access Authorization Verification Feature



APPENDIX B - Appendix B: Deviations from *Service Control Point Node Generic Requirements*

The purpose of this appendix is to reflect the variations on or deviations to the functionality of the Service Control Point (SCP) Node as described in *Service Control Point Node Generic Requirements* [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.] in the Ameritech IN Release 0 SCP. In general, the IN Release 0 SCP (or SCP) does not consist of a node co-located with one or more applications as described in the Introduction of *Service Control Point Node Generic Requirements* but will, in addition to the functionality described in the main text of this document, provide the functionality as detailed in *Service Control Point Node Generic Requirements*. Therefore, any mention of the node monitoring an application, the node passing messages to an application, an application passing messages to the node, etc., does not apply for the Ameritech IN Release 0 SCP. In addition, there will be no interface to an X.25 network. All mention of support of or communication with that network will not be applicable in the Release 0 time frame. Finally, there will be no direct interface to the Signaling, Engineering, and Administration System (SEAS) for Release 0. Any reference to an SCP Node/SEAS Interface Specification should be ignored. All information normally sent to SEAS will be sent to either Switching Control Center System (SCCS) or to Service Management System (SMS.) [Reference to an SCP/SMS Interface Specification should also be ignored. That interface will remain proprietary for Release 0.]

Following is a list of the first and second level headings as they are found in the *Service Control Point Node Generic Requirements*. [The IN Release 0 SCP will be referred to as the SCP throughout the remainder of this document.] Under each section heading is a description of any deviations to or variations on the functionality described in that section for Release 0. If a section is left blank, it should be assumed that the IN Release 0 SCP will perform the functionality as described in that section.

B.1. Introduction

B.1.1. Purpose and Description

The SCP will not contain a distinction between the node and applications for Release 0.

B.1.2. Definitions

B.1.3. Organization

B.2. Network Plan

B.2.1. General

- The SCP will not interface to an X.25 service network for Release 0.
- The SCP will not interface directly to SEAS for Release 0.

B.2.2. Common Channel Signaling Network

B.2.3. BCC Public Packet-Switched Network

The SCP will not interface to the Public Packet-Switched Network for Release 0.

B.2.4. Service Management System

The interface between the SCP and SMS will be a proprietary interface for Release 0.

B.2.5. SEAS(TM) System

The SCP will not interface directly to SEAS for Release 0.

B.2.6. Maintenance System and Local Maintenance Interfaces

B.3. System Architecture

B.3.1. Node/Application Distinction

This section is not applicable for the Ameritech IN Release 0 SCP.

B.3.2. Multiple Applications

This section is not applicable for the Ameritech IN Release 0 SCP.

B.3.3. User Programmability

References to a separate node and application are not applicable for the Ameritech IN Release 0 SCP.

B.4. Features

B.4.1. Feature List and Functional Guide

Discussion of multiple applications is not applicable for the Ameritech IN Release 0 SCP.

B.4.2. Feature Definitions/Descriptions

This section is not applicable for the Ameritech IN Release 0 SCP.

B.4.3. Node Capabilities

The SCP will not interface to the X.25 service network for Release 0.

B.5. Message Processing

B.5.1. General

B.5.2. Message Treatment

- Discussion of the handling of X.25 messages is not applicable for Release 0.
- Discussion of message distribution (to the appropriate application) is not applicable for Release 0.
- **What application identifier will the SCP attach to outgoing messages?**
- There will be a variation on overload actions for Release 0.
  - There will be no distinction between application and node.
  - The SCP will not inform SMS of its overload level.
  - The SCP will not have the same levels of overload.
  - Overload may be determined by something more than message response time.

B.5.3. Internal Call-Processing Functions

B.6. Signaling

B.6.1. Signaling System 7

B.6.2. X.25

This section is not applicable for the Ameritech IN Release 0 SCP.

B.7. Transmission

B.8. Administration

B.8.1. Billing

B.8.2. Traffic Measurements

- Measurements discussed in this section will be sent to either SCCS or SMS for Release 0.
- Collection of measurements for the X.25 service-related network will not be supported for Release 0.
- Component measurements of operations system messages originating/terminating will be maintained separately for SMS and SCCS.
- Disk subsystem component measurements will not be supported for Release 0.
- Exception Reports will be sent to SCCS for Release 0.
- Record base measurements will be sent to SCCS or an administration terminal for Release 0.
- If SCP storage capacity is nearing its maximum, as alarm will be sent to SCCS.

B.8.3. Service Measurements

- SCP unavailability (not SCP node unavailability) measurements will be maintained for Release 0.
- Application unavailability measurements are not applicable for Release 0.
- Disk measurements are not applicable for Release 0.

- Exception notices will be sent to SCCS for Release 0.
- Exception notices for change in application/node status are not applicable for Release 0. Instead, an exception notice will be sent for a change in system status.
- Exception notices for messages received for a non-existent application are not applicable for Release 0. Instead, an exception notice will be sent for messages receives for a non-existent feature.

B.8.4. Database Backup and Recovery

B.8.5. SCP Data Provisioning

- There will be no SCP interface to SEAS for Release 0. All access to the SCP database will be via SMS and possibly via SCCS.
- NSPEC data will be maintained by the SCP as it applies to the individual system. (E.g., it will contain an Application System ID rather than a Node Software Generic ID.)
- An ASPEC file is not applicable for the IN Release 0 SCP. Instead, similar information will be maintained on a feature basis. Note that there will not be individual application (or feature) timers. All queries for Release 0 will have the same response time requirements.
- Specification data will be administered by SMS for Release 0.

B.8.6. Generic Program Alteration

B.8.7. Security

B.8.8. Supplier Support

B.8.9. System Testing and Integration

B.9. Maintenance

B.9.1. SCP System Maintenance

- Database trouble detection will be provided for the system (not for a node and service application) for Release 0.
- Maintenance alarms will be sent to SCCS for Release 0.

B.9.2. CCS Signaling Link Maintenance

B.9.3. Data Link Maintenance

- Data link maintenance will not be provided for an X.25 service network.

B.9.4. Maintenance Measurements

- Maintenance measurements will be sent to a MOC and SCCS for Release 0.
- Service measurements will be maintained as they apply but there will be no distinction between service measurements made by the node or by the application for Release 0.

B.9.5. Remote Maintenance

B.9.6. Network Maintenance

B.10. System Interfaces

B.10.1. CCS Interface

B.10.2. X.25 Interface

- An X.25 network interface is not supported for Release 0.

B.10.3. Operations System Interface

- The SCP/SMS Interface will remain proprietary for Release 0.
- The interface to SEAS will not be supported for Release 0.
- The interface to SCCS will be provided in one of two ways: the asynchronous interface as described in LSSGR FSD 35-08-0100 or the synchronous interface as described in LSSGR FSD 35-08-0200.

B.11. Service Standards

- References to X.25 network response time are not applicable for Release 0.

B.12. Reliability and Quality

B.12.1. Introduction

B.12.2. Availability

B.12.3. Data Integrity

B.12.4. Additional Reliability and Quality Requirements

B.13. Power

B.14. Equipment

B.15. Electromagnetic and Electrical Environment

B.16. Network Traffic Management

- Counts maintained for network management will be sent to SCCS, not SEAS, for Release 0.

B.17. System Capacity

B.17.1. General

B.17.2. Messages

B.17.3. Adding Capacity

B.18. Synchronization

B.19. Documentation

B.19.1. General

B.19.2. Required Documentation

B.20. Data

- This section is not applicable for Release 0 as it applies to an X.25 service network.