TRILLIUM®

PSF - ISUP (FT/HA)

Functional Specification

1091146 1.3

PSF - ISUP (FT/HA)

Functional Specification

1091146 1.3

Trillium Digital Systems, Inc.
12100 Wilshire Blvd., Suite 1800
Los Angeles, CA 90025-7118
Phone: +1 (310) 442-9222
Fax: +1 (310) 442-1162
Web: http://www.trillium.com

PSF - ISUP (FT/HA)
Functional Specification
1091146 1.3

Printed in U.S.A.

Preface

Objective

This document provides a functional description of the PSF - ISUP (FT/HA) software (p/n 1000146) designed by Trillium Digital Systems, Inc.

Audience

Trillium assumes that the readers of this document are familiar with telecommunication protocols, specifically SS7 and Trillium's ISUP, PSIF - ISUP, and Fault-Tolerant/High Availability Core products.

Document Organization

This document is organized into the following sections:

Section	Description
1 Introduction	Provides an overview of the product, including the product description and features
2 Environment	Describes assumptions about the operating environment for PSF - ISUP. Explains Trillium's TAPA architecture and how PSF - ISUP fits within the TAPA model.
3 Protocol Characteristics	Specifies the standards to which the software conforms, and the protocol features supported
4 System Characteristics	Defines features not directly related to the protocol, such as the management interface
5 Memory and Performance Characteristics	Provides the performance characteristics and memory size of the PSF - ISUP software, including total code sizes derived under sample compile conditions. This section also gives maximum allowable configurations.

Document Set

The suggested reading order of this document set is:

1. *Functional Specification*

   Contains the features and highlights that describe the protocol and system characteristics. It includes the memory characteristics and conformance details.

2. *Training Course*

   Offers a detailed overview of the features and interfaces of the software. It contains code samples, data flow diagrams, and a list of files.

3. *Service Definition*

   Describes the procedures and layer manager interface used to pass information between the software and other software elements. The Interface Primitives section describes the services of the software. The Interface Procedures section describes and illustrates the flow of primitives and messages across the interfaces.

   Note: *Information on porting the software is contained in the Service Definition.*

4. *Software Test Sample*

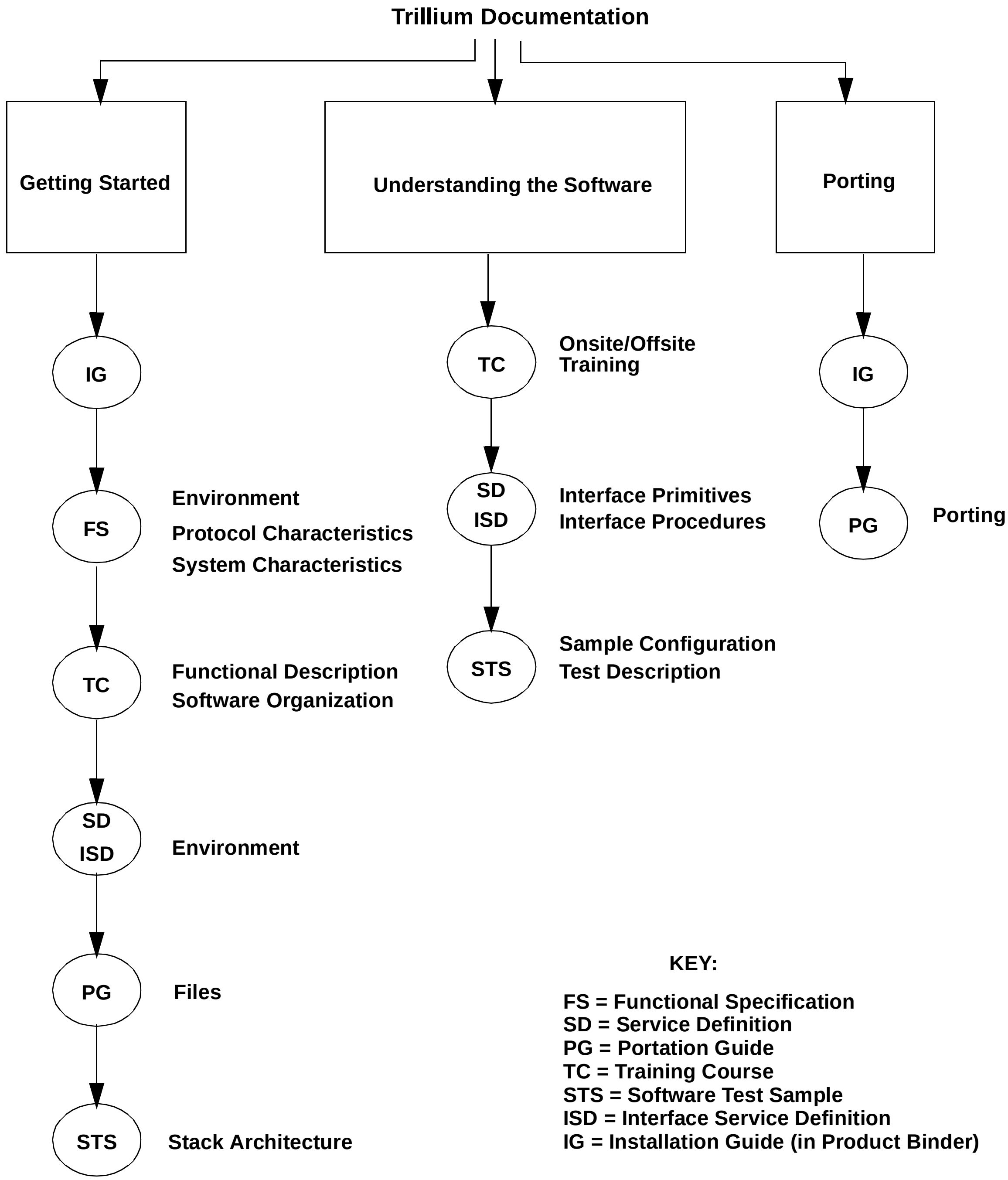
   Describes the sample files delivered with the product and the procedures to build a sample test. This test partially demonstrates the product initialization, configuration, and execution. It may contain data flow diagrams illustrating the correct operation of the software.

In addition to the above PSF documents, the following documents should also be read for a better understanding of the fault-tolerant system:

1. *Fault-Tolerant/High-Availability (FT/HA) Core Functional Specification*
2. *Fault-Tolerant/High-Availability (FT/HA) Core Service Definition*

Using Trillium Documentation

The figure below illustrates the various approaches the user can take when utilizing the software documentation. First time users should read the documents under the **Getting Started** column; important sections and subsections are listed to the right of each document. For users familiar with the documentation but who need to look up certain points concerning the use of the software, the **Understanding the Software** column is suggested. The **Porting** column is for those users who are familiar with Trillium software and related telecommunications protocols and who wish to install the software immediately onto their operating systems.



Notations

This table displays the notations used in this document:

Notation	Explanation	Examples
Arial	**Titles**	**1.1 Title**
Palatino	Body text	This is body text.
Bold	**Highlights information**	**Loose coupling, tight coupling**, **upper layer interface**
ALL CAPS	CONDITIONS, MESSAGES	AND, OR CONNECT ACK
Italics	*Document names, emphasis*	*PSF - ISUP (FT/HA) Functional Specification* This adds *emphasis*.
`Courier New Bold`	**`Code` `Filenames, pathnames`**	**`PUBLIC S16 ZiMiLziCfgReq(pst, cfg)` `Pst *pst;` `CmPFthaMngmt *cfg;`**

Release History

This table lists the history of changes in successive revisions to this document:

Version	Date	Initials	Description
1.3	December 31, 1999	sk	Changes for software release 1.2, including: • Addition of multiple point code support • Addition of NTT and Bellcore variants • FT/HA support for PSIF - ISUP
1.2	November 16, 1998	rs	• Initial release for software version 1.1
1.1	November 03, 1998	rs	• Preliminary release

Contents

Illustrations

1 INTRODUCTION

This document provides a functional description of the Protocol Specific Function - ISDN User Part (PSF - ISUP) software (p / n 1000146) designed by Trillium Digital Systems, Inc. PSF - ISUP adds Fault-Tolerant / High-Availability (FT / HA) functionality to Trillium's ISUP and PSIF - ISUP products. PSF - ISUP can be viewed as a library of functions that ISUP and PSIF - ISUP invoke only in an active / standby fault-tolerant environment.

Trillium's PSF - ISUP software provides interfaces to perform the following basic functions:

- Run-time state update of standby
- Warmstart of an Out-Of-Service (OOS) node to make it standby
- Controlled switchover of active and standby ISUP nodes
- Forced switchover, via the standby node (PSF), on failure of active ISUP node

The PSF - ISUP software is portable C source code that can be compiled to run on any processor, under any operating system, and with an active / standby system architecture. A modular design and simple interfaces allow the PSF - ISUP software to be easily ported to almost any environment.

Note: *It is assumed that the active and standby nodes run simultaneously on the same platform.*

1.1 Terms and Definitions

The following terms are used in this document:

Term	Definition
Active node	A node that executes software to provide the necessary protocol functionality. The active node processes the protocol messages and updates the new state information in the standby node.
Controlled switchover	A procedure that makes a standby node active and an active node standby
Fault-tolerant node	A pair of nodes with replicated protocol layers. A fault-tolerant node can be in an active, standby, or out-of-service state.
Forced switchover	A procedure that makes a standby node active when an active node goes out-of-service
Node	A unit that has a processor(s) with private volatile memory inaccessible to all other nodes, and a private clock governing the execution of instructions on this processor. A node also has a network interface connecting it to a communication network using communication channels. The software governs the sequence of instructions executed on a node.
Out-Of-Service (OOS) node	An off-line node that has the ability to become an active or standby node
Run-time state update	As the active ISUP handles protocol events, which may result in internal ISUP state changes, the active PSF - ISUP updates the standby with the state changes to keep the standby synchronized. PSF - ISUP also updates stable state changes that occur in PSIF - ISUP.
Standby node	A node that acts as a backup to an active node
Warmstart	A procedure that makes an OOS active node standby. An active node updates this new standby node with current information using a bulk update procedure.

1.2 Abbreviations

The following abbreviations are used in this document:

Abbreviation	Definition
ANSI	American National Standards Institute
ETSI	European Telecommunications Standards Institute
FT / HA	Fault-Tolerant / High-Availability
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
ITU	International Telecommunications Union
LM	Layer Manager
MTP3	Message Transfer Part Level 3
OOS	Out-Of-Service
PSF	Protocol Specific Function
PSIF	Protocol Specific Interface Function
SAP	Service Access Point
SCCP	Signalling Connection Control Part
SS	System Services
TAPA	Trillium Advanced Portability Architecture

2 ENVIRONMENT

This section describes design assumptions about the operating environment for the PSF - ISUP software.

2.1 Trillium Advanced Portability Architecture (TAPA)

Trillium’s ISUP product conforms to Trillium Advanced Portability Architecture (TAPA). TAPA can be visualized as a box surrounded by four outer boxes. The box in the center represents the ISUP software, while the four outer boxes represent other software to which ISUP can be connected. The separation between the center box and outer boxes defines the interfaces across which ISUP interacts with the other software.

For example, in Figure 2-1, the lower layer could be Trillium’s MTP Level 3 software and the upper layer could be Trillium’s Interworking Call Control.

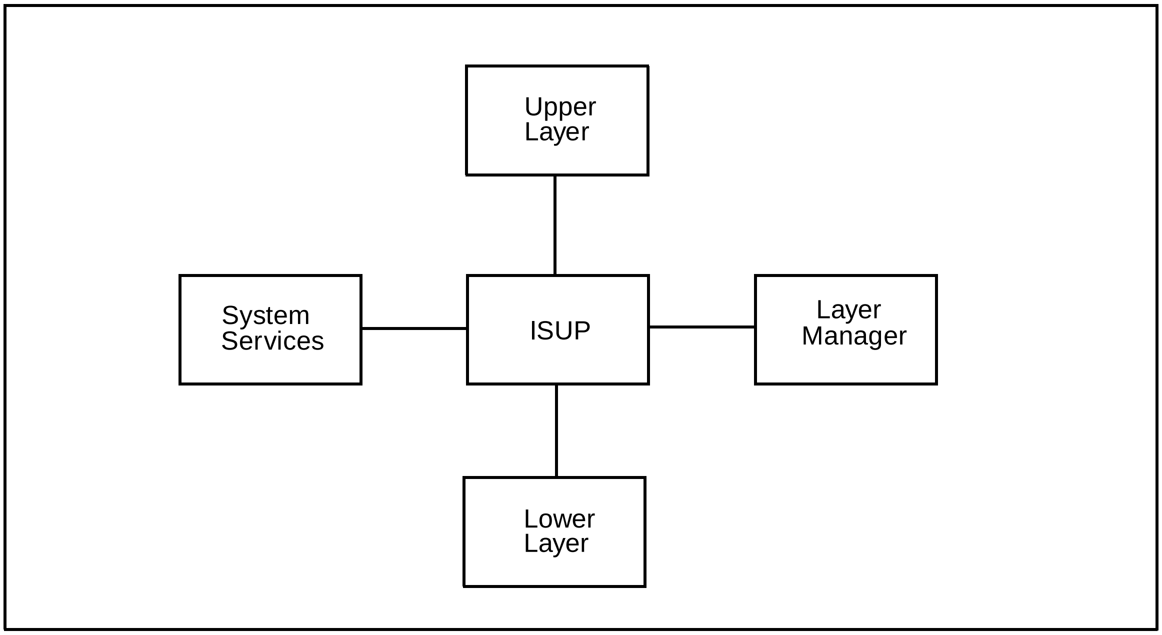


Figure 2-1: Trillium Advanced Portability Architecture (TAPA)

2.2 PSF - ISUP Architecture

PSF - ISUP architecture differs from the architecture of the Trillium products that conform to TAPA, as PSF - ISUP does not have an upper or lower layer. However, PSF - ISUP does provide the standard Layer Manager (LM) and System Services (SS) interfaces.

Figure 2-2 illustrates the architecture of the PSF - ISUP software environment.

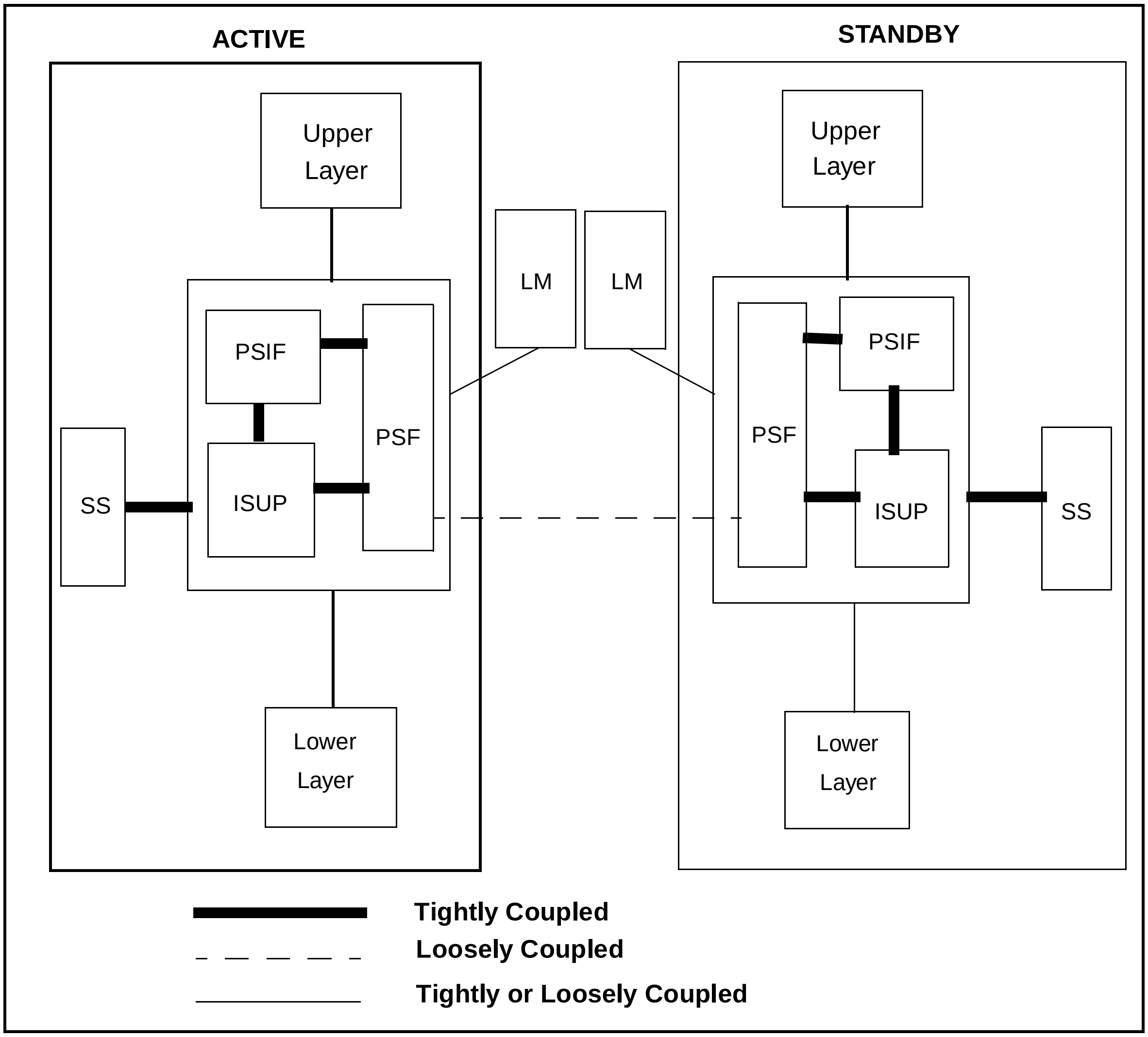


Figure 2-2: PSF - ISUP environment

The interfaces in Figure 2-2 are:

Interface	Description
System Services (SS)	This interface provides the system functions required by PSF - ISUP, including, initialization, timer management, memory management, message and queue management, date and time management, and resource checking. The System Services interface is always tightly coupled.
Layer Manager (LM)	This interface provides the functions required to control and monitor PSF - ISUP. In addition, this interface provides the functions to initialize and modify PSF - ISUP configuration parameters. This interface can be tightly or loosely coupled.
Peer PSF - ISUP	This interface provides functions for the active PSF - ISUP to communicate with the standby PSF - ISUP, and vice versa. These functions involve state updates from active to standby during run time, warmstart, and controlled switchover. This interface is always loosely coupled.
PSF - ISUP and ISUP (PSIF - ISUP)	PSF - ISUP updates the stable state changes during run-time state update. In order to avoid excessive overhead, all transient states are updated during warmstart and controlled switchover, but not during run time. This interface is always tightly coupled.

PSF - ISUP interacts with the layer manager and the peer PSF - ISUP using a set of primitive functions. These primitives, taking the form of requests, indications, responses, and confirms, completely define the interaction between the layers. Information flows between layers through the exchange of primitives across Service Access Points (SAPs).

3 PROTOCOL CHARACTERISTICS

This section describes PSF - ISUP features and primitives.

PSF - ISUP implements the functionality required to allow Trillium's ISUP (PSIF - ISUP) layer to function in a Fault-Tolerant / High-Availability (FT / HA) system.

Only the active copy of ISUP (PSIF - ISUP) participates in protocol execution. While handling an event, the active ISUP (PSIF - ISUP) can modify its internal states. The active PSF - ISUP updates the state changes to the standby to keep the standby synchronized with the active. In order to reduce the run-time update overhead and the complexity involved in the update procedure, the active PSF - ISUP updates only high-level, stable ISUP (PSIF - ISUP) states to the standby.

Transient states, those states that last for a relatively small duration, are updated only at the time of a controlled switchover and not during run time. During a forced switchover, any transient state information existing at that time is lost. Therefore, the calls in the transient state are lost. The standby resumes operation with its current, stable states.

3.1 Basic Features

PSF - ISUP supports the following basic features:

1. Warmstart of the peer ISUP (PSIF - ISUP) for making it standby from Out-Of-Service (OOS). The active PSF - ISUP updates all stable states to the peer as part of this procedure.
2. Run-time state updates to keep the standby ISUP (PSIF - ISUP) synchronized with the active ISUP (PSIF - ISUP). Only the high-level, stable states are updated at run time to avoid update overhead during run-time operation of ISUP (PSIF - ISUP).

   The following states in ISUP are updated at run time:

State	**Description**
Connection states	• Idle • Answered
Maintenance and hardware blocking (circuit) states	• Idle • Locally blocked • Remotely blocked • Locally and remotely blocked
Upper/lower SAP states	• Bind enable • Unbind disable
Interface control block	• Available • Unavailable

In addition, state update for the following connection states (related to unanswered calls) can be turned on/off based on the value of the update option described in Section 4.1.1.1, "General Configuration":

- Incoming call – address complete indication sent
- Outgoing call – address complete received (waiting for answer)

The following transient states in ISUP are not updated at run time. However, they are all updated during controlled switchover:

State	Description
Connection states	• Waiting for continuity • Waiting for continuity report • Waiting for circuit reservation acknowledgment (ANSI only) • Waiting for continuity report and circuit reservation acknowledgment (ANSI only) • Waiting for digits • Waiting for address complete • Waiting for release complete • Waiting for release complete and release response
Maintenance and hardware blocking states	• Waiting for block acknowledgment • Waiting for block response • Waiting for unblocking acknowledgment • Waiting for unblocking response • Waiting for reset acknowledgment • Waiting for reset response

The following states in PSIF - ISUP are updated at run time:

- Upper/lower SAPs:
    - Bind enable
    - Unbind Disable

None of the states for the following entities in PSIF - ISUP are updated at run time. However, they are all updated during control switchover:

- Interface maintenance control block
- Group maintenance control block
- Maintenance release control block

3. No loss of information during controlled switchover. The transient states related to all ISUP (PSIF - ISUP) procedures are updated to the standby before the controlled switchover.
4. Ability to let ISUP (PSIF - ISUP) handle protocol events during warmstart. The active PSF - ISUP performs the warmstart state update in multiple scheduling of ISUP (PSIF - ISUP) so that other processes do not starve; therefore, protocol events can be handled by ISUP (PSIF - ISUP) between state updates.
5. Forced switchover when the active ISUP (PSIF - ISUP) becomes OOS. Upon switchover, the standby PSF - ISUP becomes active and resumes operation with current, stable states.
6. The functionality to abort an ongoing warmstart state update or controlled switchover state update.
7. The disabling of run-time state update from the active node to the standby node when the standby becomes OOS.
8. Shutdown procedure that resets PSF - ISUP states and deallocates the memory allocated by PSF - ISUP for its operation up to the time of the shutdown.
9. Alarm generation on finding failures during PSF - ISUP operation.
10. Debug print generation, if enabled by the layer manager.
11. Multiple point codes implementation , which updates the interface ID information to the standby side.
12. PSF - ISUP supports the following ISUP variants:
    - ITU Q.761-Q.764 (1988 and 1992)
    - Q.730, Q.766, and Q.767
    - ANSI T1.113 (1988 and 1992)
    - ETSI ETS 300-356
    - German Telecom FTZ 163-TR-75.95
    - Italian Telecom ISUP-S
    - Singapore Telecom SS7 Specification
    - NTT (Japan) specification
    - Bellcore

3.2 Primitives

The following primitives interface PSF - ISUP with the peer PSF - ISUP and the layer manager.

3.2.1 Layer Manager Primitives

The following groups define the primitives that interface PSF - ISUP with the layer manager.

3.2.1.1 Configuration

The following primitives configure PSF - ISUP:

Name	Description
`ZiMiLziCfgReq`	Configuration request
`ZiMiLziCfgCfm`	Configuration confirm

3.2.1.2 Solicited Status

The following primitives gather information to determine the current state of the PSF - ISUP software:

Name	Description
`ZiMiLziStaReq`	Status request
`ZiMiLziStaCfm`	Status confirm

3.2.1.3 Unsolicited Status

The following primitive provides information indicating a change in status of the PSF - ISUP software:

Name	Description
`ZiMiLziStaInd`	Status indication

3.2.1.4 Control

The following primitives control PSF - ISUP operation:

Name	Description
`ZiMiLziCntrlReq`	Control request
`ZiMiLziCntrlCfm`	Control confirm

3.2.2 Peer PSF - ISUP Primitives

The following group defines the primitives that interface PSF - ISUP with the peer PSF - ISUP.

3.2.2.1 Data Transfer

Data transfer primitives carry out state updates between the active and the standby PSF - ISUP:

Name	**Description**
`ZiPiOubDatReq`	Outbound data request
`ZiPiInbDatReq`	Inbound data request
`ZiPiOubDatCfm`	Outbound data confirm
`ZiPiInbDatCfm`	Inbound data confirm

4 SYSTEM CHARACTERISTICS

This section describes the features of PSF - ISUP not directly related to PSF - ISUP's state update functionality.

4.1 Management

The layer manager consists of the configuring, monitoring, and controlling functions that ensure the error-free and efficient operation of PSF - ISUP.

4.1.1 Configuration

The layer manager performs the general and peer PSF - ISUP SAP configuration.

Configuration parameters are received from the layer manager. Individual parameters cannot be configured selectively—that is, without specifying the rest of the configuration parameters. The specified ranges can be changed by modifying the appropriate type definitions. The actual values that can be assigned depend on the availability of system resources, such as memory.

Some of the configuration parameters can be reconfigured in a running system without affecting PSF - ISUP operation.

4.1.1.1 General Configuration

The following parameters are configurable for the entire PSF - ISUP software:

Parameters	**Reconfigurable?**	**Allowable Values**
Timer resolution	No	0 to 32767
Virtual processor ID of the ISUP node	No	0 to 65535
Memory region for allocating message buffers for mailing a message to PSF - ISUP	No	0 to 255
Memory pool for allocating message buffers for mailing a message to PSF - ISUP	No	0 to 255
Layer manager post structure for reporting alarms	Yes	See note below table
`updateOption` — Configuration parameter to determine if state updates should be done for unanswered calls	Yes	**`0`** - Do not update **`1`** - Update

Note: *Refer to the System Services Interface Service Definition for the range of allowable values for the post structure.*

4.1.1.2 Peer SAP Configuration

This option configures the SAP towards the peer PSF - ISUP:

Parameter	**Reconfigurable?**	**Allowable Values**
Memory region for allocating message buffers for mailing a message to the peer PSF - ISUP	Yes	0 to 255
Memory pool for allocating message buffers for mailing a message to the peer PSF - ISUP	Yes	0 to 255
Peer PSF - ISUP's physical processor ID	Yes	0 to 65535
Peer PSF - ISUP's entity ID	Yes	0 to 255
Peer PSF - ISUP's instance ID	Yes	0 to 255
Priority for the post structure of the peer SAP	Yes	User defined
Route for the post structure of the peer SAP	Yes	Default
Selector for the post structure of the peer SAP	Yes	Not currently used
Value for the timer started by the active PSF - ISUP to wait for confirm from the standby for warmstart or controlled switchover state update	Yes	0 to 65535
Maximum update message size	Yes	X to (2^{32} - 1) (See note below table)

Note: *The minimum size of the update message must be greater than the maximum size of a table to be packed by PSF - ISUP.*

4.1.2 Control

Control functions change the status of the active/standby ISUP (PSIF - ISUP) and perform warmstart and controlled switchover procedures. Control can be exerted at any time by the layer manager.

The following table describes the control functions:

Control Function	**Description**
Go active	Makes ISUP (PSIF - ISUP) active with peer SAP enabled or disabled, depending on whether the peer is standby or OOS. This action can be used in the following scenarios: • When both nodes on which ISUP (PSIF - ISUP) resides are OOS, one node is made active (initialization) • When an OOS ISUP (PSIF - ISUP) node must be made active • When a standby ISUP (PSIF - ISUP) must be made active for forced or controlled switchover • When a current active node must remain active (see note below table)
Go standby	Makes ISUP (PSIF - ISUP) standby with the peer SAP enabled. This action can be used in the following scenarios: • When an OOS ISUP (PSIF - ISUP) node must be made standby • When an active ISUP (PSIF - ISUP) must be made standby for controlled switchover • When a current standby node must remain standby (see note below table)
Warmstart	Enables and warmstarts the SAP towards the peer (that is, sends all stable state information to the standby). This action is used by the layer manager when an OOS ISUP (PSIF - ISUP) node must be made standby. This action is valid only for the active PSF - ISUP.
Synchronize	Updates the standby with transient state information. This action is used by the layer manager during a controlled switchover of an ISUP (PSIF - ISUP) node. The active PSF - ISUP sends all transient states (which are not updated during run time) to the standby. This action is valid only for the active PSF - ISUP.
Disable peer SAP	Disables the peer SAP. This action is used by the layer manager when a standby ISUP (PSIF - ISUP) node becomes OOS. The active ISUP (PSIF - ISUP) stops sending run-time state update messages after the peer SAP is disabled. This action is valid only for the active PSF - ISUP.
Shutdown	Shuts down PSF - ISUP operation. PSF - ISUP deallocates all allocated memory and goes into the state that PSF - ISUP maintains after initialization. PSF - ISUP can then be reconfigured.

Control Function	Description
Abort	Aborts the ongoing warmstart or controlled switchover procedure. The layer manager uses this action to abort an ongoing procedure after the layer manager has sent a warmstart or synchronization request to the active. This action is valid only for the active PSF - ISUP.
Disable alarm indication	Disables alarm generation (unsolicited status indications)
Enable alarm indication	Enables alarm generation
Disable debug print generation	Disables debug print generation
Enable debug print generation	Enables debug print generation

Note: *The layer manager can request the active PSF - ISUP to synchronize the standby ISUP (PSIF - ISUP) to perform a controlled switchover. When the active PSF - ISUP completes synchronization, the layer manager sends a control request to the active ISUP (PSIF - ISUP) to become standby and another control request to the standby ISUP (PSIF - ISUP) to become active in order to complete the switchover. After synchronization, the layer manager may decide not to continue with the controlled switchover because of synchronization failure or because the user wants to abort the controlled switchover. In that case, the layer manager must send a control request to the active ISUP (PSIF - ISUP) to remain active and a control request to the standby ISUP (PSIF - ISUP) to remain standby, in order to resume protocol operation.*

4.1.3 Alarms

PSF - ISUP generates alarms (unsolicited status indications) to the layer manager to indicate an abnormal condition. The following table lists several examples of alarms:

Alarm	Description
Memory failure alarm	PSF - ISUP fails to allocate a buffer for sending an update message to the standby
Update message error alarm	The PSF - ISUP on the standby node detects an error in a state update message received from the active
Sequence error alarm	The PSF - ISUP on the standby node detects a sequence error in the state update message received from the active. This can occur if a state update message from active node to standby is lost.

4.1.4 Debug Print Generation

Debug prints can be generated from PSF - ISUP at different levels. These prints can be enabled or disabled individually by the layer manager at run time. The following debug prints are supported by PSF - ISUP:

Debug Prints	Description
Protocol layer interface debug prints	• Generated whenever a function is called at the PSF - ISUP and ISUP (PSIF - ISUP) interface. • Indicates the function name and the parameters passed to the function
Peer PSF - ISUP interface debug prints	• Generated whenever a function is called at the PSF - ISUP and the peer PSF - ISUP interface • Indicates the function name and the parameters passed to the function
Layer manager interface debug prints	• Generated whenever a function is called at the PSF - ISUP and the layer manager interface • Indicates the function name and the parameters passed to the function
Pack debug prints	• Generated whenever a pack function is called by the active PSF - ISUP to pack the states in an update message
Unpack debug prints	• Generated whenever the standby PSF - ISUP calls an unpack function to unpack the states from an update message received from the active

4.1.5 Status

Status information indicates the current state of the software. Status information may be gathered at any time by the layer manager. Collection of status information does not change any of the information examined.

4.1.5.1 General

The layer manager issues a status request to determine the ISUP (PSIF - ISUP) state. ISUP (PSIF - ISUP) can be in an active, standby, or OOS state.

4.1.5.2 Peer SAP

The layer manager issues a status request to determine the bind state and the update state of the peer SAP. The peer SAP can be either bound or unbound. The update state can have the following values:

- Idle
- Warmstart is going on
- Synchronization is going on
- Wait for confirmation from the standby for the warmstart completion
- Wait for confirmation from the standby for the synchronization completion

5 MEMORY AND PERFORMANCE CHARACTERISTICS

This section describes the code size, data size, and performance characteristics of the PSF - ISUP software.

5.1 Code Size

The code size is the number of bytes of memory needed for the executable code. Code size includes all function calls to system services, the layer manager, and the upper and lower layers, but does not include the actual code provided within these functions.

Code size depends on the options delivered, the compiler, linker, locator, memory model, and whether all delivered features (such as error checking, management capabilities, and protocol capabilities) are retained. The code size can be determined from the software link map.

5.1.1 Tightly Coupled Interface

A sample compile under the following conditions:

Type	Condition
Product	PSF - ISUP (FT / HA), version 1.2
Compiler	Microtec C 68k Compiler, version: 4.4
Product option	ITU - T (1988 and 1992)
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following code sizes:

File	Size (bytes)
`zi_bdy1.c`	4415
`zi_bdy2.c`	7871
`zi_bdy3.c`	4615
`zi_bdy4.c`	8237
`zi_ex_ms.c`	237
`zi_ptmi.c`	191
`zi_ptpi.c`	121
`zi_id.c`	55
Total	**25742**

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Compiler	Microtec C 68k Compiler, version: 4.4
Product option	ITU - T (1988 and 1992) and ANSI (1988 and 1992)
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following code sizes:

File	**Size (bytes)**
`zi_bdy1.c`	4415
`zi_bdy2.c`	7871
`zi_bdy3.c`	4627
`zi_bdy4.c`	8243
`zi_ex_ms.c`	237
`zi_ptmi.c`	191
`zi_ptpi.c`	121
`zi_id.c`	55
Total	**25760**

A sample compile under the following conditions:

Type	Condition
Product	PSF - ISUP (FT / HA), version 1.2
Compiler	Microtec C 68k Compiler, version: 4.4
Product option	ITU - T (1988 and 1992) and ETSI (1995)
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following code sizes:

File	Size (bytes)
`zi_bdy1.c`	4415
`zi_bdy2.c`	7871
`zi_bdy3.c`	4651
`zi_bdy4.c`	8277
`zi_ex_ms.c`	237
`zi_ptmi.c`	191
`zi_ptpi.c`	121
`zi_id.c`	55
Total	**25818**

5.1.2 Loosely Coupled Interface

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following code sizes:

File	**Size (bytes)**
`zi_bdy1.c`	5685
`zi_bdy2.c`	8793
`zi_bdy3.c`	4463
`zi_bdy4.c`	9557
`zi_ex_ms.c`	645
`zi_ptmi.c`	775
`zi_ptpi.c`	207
`zi_id.c`	55
Total	**30180**

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ANSI (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following code sizes:

File	**Size (bytes)**
`zi_bdy1.c`	5685
`zi_bdy2.c`	8793
`zi_bdy3.c`	4475
`zi_bdy4.c`	9563
`zi_ex_ms.c`	645
`zi_ptmi.c`	775
`zi_ptpi.c`	207
`zi_id.c`	55
Total	**30198**

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ETSI (1995)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following code sizes:

File	**Size (bytes)**
`zi_bdy1.c`	5685
`zi_bdy2.c`	8793
`zi_bdy3.c`	4499
`zi_bdy4.c`	9597
`zi_ex_ms.c`	645
`zi_ptmi.c`	775
`zi_ptpi.c`	207
`zi_id.c`	55
Total	**30256**

5.2 Static Data Size

The static data size is the number of bytes of memory needed for:

- Initialized variables and structures (such as state matrices or strings)
- Uninitialized variables and structures (such as anchors for control points and SAPs)

Static data is allocated at compile time and represents the global variables and structures used by the software. Static data does not include any structures allocated at run time.

The static data size depends on the options delivered, the compiler, linker, and memory model. The static data size can be determined from the software link map.

5.2.1 Tightly Coupled Interface

A sample compile under the following conditions:

Type	Condition
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following static data sizes:

Type	Sizes (bytes)
Strings	196
Constants	36248
Initialized variables	8244
Uninitialized variables	22356
Total	**67044**

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ANSI (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following static data sizes:

Type	**Sizes (bytes)**
Strings	196
Constants	43832
Initialized variables	9736
Uninitialized variables	22360
Total	**76124**

A sample compile under the following conditions:

Type	Condition
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ETSI (1995)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Disabled
Layer coupling	Tightly coupled across all interfaces

Yielded the following static data sizes:

Type	Sizes (bytes)
Strings	196
Constants	36520
Initialized variables	8244
Uninitialized variables	22356
Total	**67316**

5.2.2 Loosely Coupled Interface

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following static data sizes:

Type	**Sizes (bytes)**
Strings	56008
Constants	36248
Initialized variables	8244
Uninitialized variables	22356
Total	**122856**

A sample compile under the following conditions:

Type	Condition
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ANSI (1988 and 1992)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following static data sizes:

Type	Sizes (bytes)
Strings	63408
Constants	43832
Initialized variables	9736
Uninitialized variables	22360
Total	**139336**

A sample compile under the following conditions:

Type	**Condition**
Product	PSF - ISUP (FT / HA), version 1.2
Product option	ITU - T (1988 and 1992) and ETSI (1995)
Compiler	Microtec C 68k Compiler, version: 4.4
Compiler options	**`-c -p68040 -H -A -v -Fsm -nKc -nKm -Gf -O -Ob -Oe -Os`**
Processor	Motorola 68040
Error checking	Enabled
Layer coupling	Loosely coupled across all interfaces

Yielded the following static data sizes:

Type	**Sizes (bytes)**
Strings	56580
Constants	36520
Initialized variables	8244
Uninitialized variables	22356
Total	**123700**

5.3 Dynamic Data Size

The dynamic data size is the number of bytes of memory needed for:

- Structures (for example, control points, SAPs) used to manage the interface and protocols
- Buffers used to store messages

Dynamic data is allocated at run time and represents memory managed by the operating system. Allocation of dynamic data depends on the compile-time configuration, run-time configuration, and net flow of traffic through the software.

The dynamic data size is dependent upon the maximum allowable configuration (A), the dynamically allocated structure size (B), the maximum number of messages to be stored (C), and the message buffer size (D). This is computed by the following formula: (A x B) + (C x D).

SS7 Glossary

A

Adjacent Signalling Points: Two signalling points directly connected through signalling links.

Authentication Center (AC): A database that stores security codes embedded into the memory of cellular phones. This code, along with the particular serial number of a given phone, prevents the use of unauthorized cellular devices within a particular network. See **Equipment Identity Register (EIR)**.

B

Base Station Subsystem (BSS): Within a cellular communications network, antenna sites (also known as cell sites) are made up of a **Base Transceiver Station (BTS)** and a **Base Station Controller (BSC).** The BSS is the pairing of the BTS and BSC. The BTS communicates with cellular phones within a given network, connecting the caller with the cell. The BSC is the interface between the BTS and any switching facilities that may be needed by the caller. See **Mobile Switching Center (MSC)**.

C

Combined Link Set: A collection of link sets that perform load sharing.

Common Channel Signalling: A signalling technique in which the signalling information is sent across the network separately from the voice and data that it is related to.

Consultative Committee International Telegraph and Telephone (CCITT): An international organization that developed communication standards such as Recommendation X.25. Replaced by the United Nation's ITU-T.

Cyclic Redundancy Check (CRC): A mathematical algorithm that derives a numerical value based on the bits in a block of data prior to transmission. If the receiving layer finds any discrepancies between the bit value of the received data packet and the accompanying frame check sequence field, then a transmission error is assumed.

D

Data Communications Equipment (DCE): Devices that handle routing and switching functions for a given network. See also **Data Terminal Equipment (DTE)**.

Destination Point Code (DPC): A node ID that identifies the destination point of a message in a signalling network. See **Originating Point Code (OPC)**.

Data Terminal Equipment (DTE): A device, such as a PC or main frame computer, that is attached to a network and is either the point of origin or the destination point of data.

E

End-to-End Signalling: Signalling that is transmitted directly between network endpoints.

Equipment Identity Register (EIR): A database that stores the serial numbers of each cellular telephone in use within a particular coverage area. In conjunction with the AC, the EIR prevents unauthorized use of cellular phones within a given geographic area.

F

Flow Control: A function that regulates the transmission of messages between adjacent protocol layers.

G

Global Title: An address which does not contain all the information necessary to route it to a specific point within the network. An example of a global title are digits dialed by the customer; in order for them to be correctly routed across an SS7 stack the SCCP translation function is needed.

Group Special Mobile (GSM): The European cellular network. Due to the use of SS7, the GSM is a more reliable network than its North American counterparts, which are still primarily analog in nature.

H

Home Location Register (HLR): A database which stores information about all cellular subscribers within a service provider's home service area. See **Visitor Location Register (VLR)**.

I

Institute of Electrical and Electronics Engineers (IEEE): Professional organization that defines network standards in a number of communication fields.

Integrated Service Digital Network (ISDN): Introduced in 1984, ISDN uses the existing infrastructure of the Public Switched Telephone Network (PSTN) to provide digital communication services for user-to-network interfaces.

Intelligent Networks Application Part (INAP): Protocol which separates switching from services in the SS7 network. Databases can be directly accessed in INAP, rather than through a Service Control Point (SCP).

International Standards Organization (ISO): Based in Geneva, Switzerland, the ISO establishes voluntary telecommunication technology standards among its ninety member countries.

ISDN User Part (ISUP): The protocol used to set up, manage, and release circuits used for voice and data transmission in the PSTN. ISUP uses out of band signalling, in which separate paths carry signalling and voice transmissions.

International Telecommunications Union - Telecommunications Standardization Sector (ITU - TSS, or more commonly ITU-T): Formerly the CCITT, this international body defines and implements recommendations and standards pertaining to the development of global telecommunications.

L

Line Information Database (LIDB): A database containing information relating to customer services, such as whether a customer subscribes to call waiting, caller identification, conference calls, and call forwarding. Also used for verification purposes with calling card services.

Load Distribution Function (LDF): Distributes traffic loads among various instances of the portable layer software (SCCP, TCAP, MTP3) based upon configuration parameters.

Local Exchange (LE): The primary switching node that provides access to the PSTN. In a basic telephony network, when a customer lifts the receiver they are connected to a local exchange, also known as a **Service Switching Point (SSP)**. The SSP provides the numbers of both the calling and called parties to a router, the **Signal Transfer Point (STP)**, so that the call can be completed across the network.

M

Media Access Control (MAC): Data link layer protocols that control traffic and data flow in multi-access channels. MACs are important in keeping LANs congestion free.

Message Transfer Part (MTP): Levels 1 through 3 of the SS7 protocol stack. Provides the upper levels with node-to-node transmission, message sequencing, and error detection/correction.

Mobile Application Part (MAP): A layer in the SS7 stack that runs on top of TCAP to query HLRs and VLRs within wireless networks. There are two standards for the MAP protocol, IS-41 and GSM. IS-41 is the ANSI standard and is used primarily in North America, while GSM is the ITU standard and is used in Europe, the U.S., and Asia.

Mobile Switching Center (MSC): Entity which receives signal strength reports from cell sites (antennas) regarding the particular strength or weakness of a cellular phone connection. Based upon these reports, the MSC determines the cell site that will then handle the given call. The MSC directly communicates with the BSC through digital facilities in the 64 Kbps range.

Multiprocessor Operating System (MOS): Trillium portable C source code designed to operate as an operating system on any embedded system or as a guest operating system under DOS, UNIX, or Solaris.

N

Network Service Part: The combination of Message Transfer Part (MTP) and Signalling Connection Control Part (SCCP).

O

Operations Administration and Maintenance (OAM): Non-data cells that provide the network with basic data management and performance diagnosis functions in order to prevent a catastrophic network or system failure.

Origination Point Code (OPC): A node ID that identifies the originating point of a message in a signalling network. See **Destination Point Code (DPC)**.

Open Systems Interconnection (OSI): An architectural system, developed by the ISO, for the interconnection of multiple data communication systems. The seven standardized layers of this model, with their associated layer managers, are: application, presentation, session, transport, network, data link, and physical. Each layer builds upon the services provided by the layers beneath it.

P

Peer Entities: Communicating entities residing in the same layer but within different nodes.

Protocol Data Unit (PDU): A unit of data used to exchange information between peer protocols communicating across a network, typically in the form of a packet with headers and/or trailers.

Protocol Specific Function (PSF): As stand-alone software, PSF provides fault tolerance functionality to portable protocol layers. In conjunction with Load Distribution Function (LDF), PSF also provides high availability functionality to the portable layer.

Protocol Specific Interface Function (PSIF): PSIF provides a generic interface from the underlying protocol layer to Trillium's Interworking Call Control (ICC) software. PSIF - ISUP understands the interface as implemented in the underlying protocol layer and maps it to a uniform interface required by ICC.

Public Switched Telephone Network (PSTN): Basic telephony system, through which calls are established and torn down between the user and a local exchange.

Q

QoS (Quality of Service): Quality of Service parameters center upon the working or contractual relationship between the service user and the service provider. These parameters, which are negotiated in advance, deal with the speed of the required service, the duration of the service, the rate of delivery, as well as desired or needed network characteristics, such as acceptable delay variations and errors in transmission.

S

Service Access Point (SAP): Information flow between the layers of a network is via Service Access Points (SAPs). The standardized interface of primitives and SAPs allows layers to be defined independently of each other. As long as the requirements of the layer interface are met, modifications may be made to the peer-to-peer protocol of one layer without affecting any upper or lower layer protocols.

Service Control Point (SCP): The interface between the STP and the database needed by the service user. The SCP is normally a computer connected to a mainframe computer that actually stores the needed information.

Service Switching Point (SSP): See **Local Exchange.**

Signal Transfer Point (STP): See **Local Exchange**

Signalling Connection and Control Protocol (SCCP): With TCAP, SCCP is part of layer 4 in the OSI protocol stack. In an SS7 network, SCCP provides end-to-end routing through the network of STPs. Unlike the layer below it (MTP 3), SCCP knows the entire route of the call. Connection Oriented SCCP is circuit-switched, while Connectionless SCCP is packet-switched.

Signalling Link: Abstract representation of the communications channel or connection between two logical nodes, including physical links and **Virtual Path Connections (VPCs)**.

Signalling System 7 (SS7): An ITU communication standard that first appeared in 1983, SS7 enables wireless and wireline call setup, call management, and call teardown over a digital signalling network. In addition to these basic functions, SS7 enables local number portability, the sharing of databases and connections within the PSTN, toll-free (800) and toll (900) wireline services, call forwarding, caller identification, and three-way calling.

Subsystem Number: Unique, fixed address given to a database, such as an HLR or VLR, used to route queries from SSPs through the network to the database itself.

T

Transaction Capabilities Application Part (TCAP): Layer 4 SS7 protocol that allows for remote database access (such as 800 or 900 numbers) from disparate networks using end-to-end switching.

Telephone User Part (TUP): The European equivalent to ISUP, except that TUP only supports analog circuits rather than digital circuits or data transmission.

V

Virtual Path Connection (VPC): A unidirectional concatenation of VPLs. Resources taken up in establishing individual VCCs are reduced or reserved by setting aside a given capacity within VPCs.

Visitor Location Register (VLR): Accesses the HLR through the SS7 network and stores information about subscribers who are outside of their home service area. See **Home Location Register (HLR)**.

Virtual Path Identifier (VPI): A distinct numerical marker, created within an 8-bit field in the ATM cell header, that identifies a particular VP for use by the cell.

Virtual Path Link (VPL): The unidirectional transmission of ATM cells within the life of a given VPI value—that is, from the assigning of the VPI at point A to its removal at point B.

References

Refer to the following documents for more information:

Fault-Tolerant/High-Availability (FT/HA) Core Functional Specification, Trillium Digital Systems, Inc. (p / n 1091133).

Fault-Tolerant/High-Availability (FT/HA) Core Service Definition, Trillium Digital Systems, Inc. (p / n 1092133).

ISDN User Part Functional Specification, Trillium Digital Systems, Inc. (p / n 1091029).

ISDN User Part Portation Guide, Trillium Digital Systems, Inc. (p / n 1093029).

ISDN User Part Service Definition, Trillium Digital Systems, Inc. (p / n 1092029).

ISDN User Part Software Test Sample, Trillium Digital Systems, Inc. (p / n 1094029).

ISDN User Part Training Course, Trillium Digital Systems, Inc. (p / n 1095029).

PSF - ISUP (FT/HA) Service Definition, Trillium Digital Systems, Inc. (p / n 1092146).

PSF - ISUP (FT/HA) Software Test Sample, Trillium Digital Systems, Inc. (p / n 1094146).

PSF - ISUP (FT/HA) Training Course, Trillium Digital Systems, Inc. (p / n 1095146).

PSIF-ISUP Functional Specification, Trillium Digital Systems, Inc. (p / n 1091141).

PSIF-ISUP Service Definition, Trillium Digital Systems, Inc. (p / n 1092141).

PSIF - ISUP Training Course, Trillium Digital Systems, Inc. (p / n 1095141).

SIT Interface Service Definition, Trillium Digital Systems, Inc. (p / n 1100010)

System Services Interface Service Definition, Trillium Digital Systems, Inc. (p / n 1111011).

Index

W

Z