



Interface Management API Implementation Agreement (Packet Over SONET Interfaces)

Revision 3.0

Editor: John Renwick, Agere Systems, jrenwick@agere.com

Copyright © 2003 The Network Processing Forum (NPF). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction other than the following, (1) the above copyright notice and this paragraph must be included on all such copies and derivative works, and (2) this document itself may not be modified in any way, such as by removing the copyright notice or references to the NPF, except as needed for the purpose of developing NPF Implementation Agreements.

By downloading, copying, or using this document in any manner, the user consents to the terms and conditions of this notice. Unless the terms and conditions of this notice are breached by the user, the limited permissions granted above are perpetual and will not be revoked by the NPF or its successors or assigns.

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "AS IS" BASIS WITHOUT ANY WARRANTY OF ANY KIND. THE INFORMATION, CONCLUSIONS AND OPINIONS CONTAINED IN THE DOCUMENT ARE THOSE OF THE AUTHORS, AND NOT THOSE OF NPF. THE NPF DOES NOT WARRANT THE INFORMATION IN THIS DOCUMENT IS ACCURATE OR CORRECT. THE NPF DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT LIMITED THE IMPLIED LIMITED WARRANTIES OF MERCHANTABILITY, TITLE OR FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS.

The words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in the remainder of this document are to be interpreted as described in the NPF Software API Conventions Implementation Agreement revision 1.0.

For additional information contact:
The Network Processing Forum, 39355 California Street,
Suite 307, Fremont, CA 94538
+1 510 608-5990 phone info@npforum.org

Table of Contents

1	Revision History	3
2	Introduction	4
2.1	ASSUMPTIONS AND EXTERNAL REQUIREMENTS	4
2.2	SCOPE	4
2.3	DEPENDENCIES	4
2.3.1	<i>Packet Over Sonet (POS) Interface</i>	4
3	Data Types	5
3.1	POS INTERFACE DATA TYPES	5
3.1.1	<i>Interface Type Code (NPF_IF_TYPE_POS)</i>	5
3.1.2	<i>Packet Over SONET (POS) Interface Attributes: NPF>IfPOS_t</i>	5
3.2	DATA STRUCTURES FOR COMPLETION CALLBACKS	5
3.2.1	<i>Completion Callback Type Codes: NPF>IfCallbackType_t</i>	5
3.2.2	<i>Asynchronous Response Details</i>	5
3.3	ERROR CODES	5
3.4	DATA STRUCTURES FOR EVENT NOTIFICATIONS	6
3.4.1	<i>Event Types: NPF>IfEvent_t</i>	6
4	Functions	7
4.1	COMPLETION CALLBACK	7
4.2	EVENT NOTIFICATION	7
4.3	INTERFACE MANAGEMENT API	7
5	References	8
6	API Capabilities	9
6.1	OPTIONAL SUPPORT OF SPECIFIC TYPES	9
6.2	API FUNCTIONS	9
6.3	API EVENTS	9
Appendix A	Header File: npf_if_POS.h	10
Appendix B	List of companies belonging to NPF DURING APPROVAL PROCESS	11

Table of Figures

1 Revision History

Revision	Date	Reason for Changes
3.0	11/22/2004	Extracted POS interface features from Interface Management API Implementation Agreement; made editorial changes, and technical changes as needed for independence of Interface Management header files.

2 Introduction

This document defines the Packet Over SONET (POS) interface type under the NPF Interface Management API.

2.1 Assumptions and External Requirements

1. This API assumes the existence of the Interface Management API Core Function Set, and shares all the same assumptions and external requirements of that API.

2.2 Scope

This document is concerned only with definitions and functions supporting POS interfaces under NPF Interface Management.

2.3 Dependencies

This API shares the same dependences as the Interface Management API Core Function Set.

2.3.1 Packet Over Sonet (POS) Interface

- This interface type represents a point-to-point SONET link carrying datagrams encapsulated in PPP protocol. It currently has no attributes; instead it inherits those of a SONET interface, of which it is a child.

3 Data Types

3.1 POS Interface Data Types

3.1.1 Interface Type Code (NPF_IF_TYPE_POS)

The type code for POS interfaces is 4 (this is a restatement of the existing definition in the Interface Management API Implementation Agreement, Core Function Set).

```
#define NPF_IF_TYPE_POS      4          /* POS Interface */
```

3.1.2 Packet Over SONET (POS) Interface Attributes: NPF>IfPOS_t

```
/*
 *   Packet over SONET (POS) Interface Attributes
 */
typedef struct NPF>IfPOS {
    /* Structure is currently empty */
};
```

3.2 Data Structures for Completion Callbacks

3.2.1 Completion Callback Type Codes: NPF>IfCallbackType_t

These codes are used as values of the NPF>IfCallbackType_t variable in callbacks, to indicate what function is returning information. No codes for POS are defined at present.

```
/*
 *   Completion Callback Types for POS interfaces
 */
```

3.2.2 Asynchronous Response Details

Asynchronous responses are returned by the mechanism defined in Interface Management API Core Function Set. The NPF>IfAsyncResponse_t structure defined there contains a (void *) pointer; in the case of POS API function calls, this pointer can point to a structure containing POS-specific or generic information. The following table indicates what is returned by each of the functions defined in this Implementation Agreement. (No POS-specific functions are currently defined.)

Function Name	Type Code	Structure Returned
---------------	-----------	--------------------

3.3 Error Codes

The following codes are used as values of NPF>IfErrorType_t.

```
/*
 *   Asynchronous error codes (returned in function callbacks)
```

```
*      used by POS interface functions  
*/
```

```
#define NPF_IF_E_POS_CODE(code) (0x10000+(NPF_IF_TYPE_POS<<8)+(code))
```

3.4 Data Structures for Event Notifications

3.4.1 Event Types: NPF_IfEvent_t

```
/*  
*      Event types  
*/
```

4 Functions

4.1 Completion Callback

No completion callback function is defined in this document. All interface type-specific functions use the callback definitions of the Core Interface Management API.

4.2 Event Notification

No event-related functions are defined in this document. All interface type-specific functions use the event function definitions of the Core Interface Management API.

4.3 Interface Management API

This section will define functions for querying and modifying the interface properties and attributes. No functions for Packet Over Sonet interfaces are currently defined.

5 References

- 1 NP Forum – Software API Conventions Implementation Agreement Revision 2.0.

6 API Capabilities

This section defines the capabilities of the Interface Management API.

It summarizes the defined APIs and Events and defines the mandatory and optional features.

6.1 *Optional support of specific types*

The support of any specific type of interface is optional in an implementation. An implementation MAY support exclusively one type of interface, and still claim compliance to the NP Forum Interface Management API.

6.2 *API Functions*

Function Name	Required?

6.3 *API Events*

Event Name	Required?

APPENDIX A HEADER FILE: NPF_IF_POS.H

```
/*
 * This header file defines typedefs, constants, and functions
 * that apply to the NPF Interface Management API, support for
 * Packet Over Sonet (POS) interface type.
 */
#ifndef __NPF_IF_POS_H__
#define __NPF_IF_POS_H__

#ifdef __cplusplus
extern "C" {
#endif

#define NPF_IF_TYPE_POS 4      /* Packet over SONET interface */

/*
 * Packet over SONET (POS) Interface Attributes
 */
struct NPF>IfPOS {
    /* Structure is currently empty */
};

#ifdef __cplusplus
}
#endif

#endif
```

APPENDIX B LIST OF COMPANIES BELONGING TO NPF DURING APPROVAL PROCESS

Agere Systems	IBM	Samsung Electronics
Alcatel	IDT	Sandburst Corporation
Altera	Intel	Silicon & Software Systems
AMCC	IP Infusion	Silicon Access
Analog Devices	Kawasaki LSI	Sony Electronics
Avici Systems	LSI Logic	STMicroelectronics
Azanda Network Devices	Modelware	Sun Microsystems
Cypress Semiconductor	Mosaid	Teja Technologies
Ericsson	Motorola	TranSwitch
Erlang Technologies	NEC	U4EA Group
EZ Chip	NetLogic	Xelerated
Flextronics	Nokia	Xilinx
Fujitsu Ltd.	Paion Co., Ltd.	Zettacom
FutureSoft	PMC Sierra	ZTE
HCL Technologies	RadiSys	
Hi/fn		