



**The ATM Forum
Technical Committee**

**Conformance ATS for
PNNI Routing**

AF-TEST-0155.000

October, 2000

© 2000 by The ATM Forum. The ATM Forum hereby grants the limited right to reproduce this specification/document in whole, but not in part, for the purpose of including this specification/document, at no extra cost and on an "as is" basis, in documentation for products. This right shall not be, and is not, transferable. Any material so included shall contain all of the disclaimers and propriety and other notices contained herein or therein. All other rights reserved. Except as expressly stated in this notice, no part of this specification/document may be reproduced or transmitted in any form or by any means, or stored in any information storage and retrieval system, without the prior written permission of The ATM Forum.

The information in this publication is believed to be accurate as of its publication date. Such information is subject to change without notice and The ATM Forum is not responsible for any errors. The ATM Forum does not assume any responsibility to update or correct any information in this publication. Notwithstanding anything to the contrary, neither The ATM Forum nor the publisher make any representation or warranty, expressed or implied, concerning the completeness, accuracy, or applicability of any information contained in this publication. No liability of any kind shall be assumed by The ATM Forum or the publisher as a result of reliance upon any information contained in this publication.

The receipt or any use of this document or its contents does not in any way create by implication or otherwise:

- Any express or implied license or right to or under any ATM Forum member company's patent, copyright, trademark or trade secret rights which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
- Any warranty or representation that any ATM Forum member companies will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
- Any form of relationship between any ATM Forum member companies and the recipient or user of this document.

Implementation or use of specific ATM standards or recommendations and ATM Forum specifications will be voluntary, and no company shall agree or be obliged to implement them by virtue of participation in The ATM Forum.

The ATM Forum is a non-profit international organization accelerating industry cooperation on ATM technology. The ATM Forum does not, expressly or otherwise, endorse or promote any specific products or services.

NOTE: The user's attention is called to the possibility that implementation of the ATM interoperability specification contained herein may require use of an invention covered by patent rights held by ATM Forum Member companies or others. By publication of this ATM interoperability specification, no position is taken by The ATM Forum with respect to validity of any patent claims or of any patent rights related thereto or the ability to obtain the license to use such rights. ATM Forum Member companies agree to grant licenses under the relevant patents they own on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. For additional information contact:

The ATM Forum
Worldwide Headquarters
1000 Executive Parkway, Suite 220
St. Louis, MO 63141
Tel: +1 314 205 0200
Fax: +1 314 576 7960

Preface

The editor would like to thank the following people for their support and help with this document: Fred Kaudel and Gregan Crawford for their advise and practical help as Testing Working Group Chairs, David Cypher for the initial contributions, and Stephan Pietsch and Mang Li for their valuable contributions and fruitful discussions.

Theofanis Vassiliou-Gioles (Editor)

This specification uses three levels for indicating the degree of compliance necessary for specific functions, procedures, or coding. They are indicated by the use of key words as follows:

- **Requirement:** "Shall" indicates a required function, procedure, or coding necessary for compliance. The word "shall" used in text indicates a conditional requirement when the operation described is dependent on whether or not an objective or option is chosen.
- **Objective:** "Should" indicates an objective which is not required for compliance, but which is considered desirable.
- **Option:** "May" indicates an optional operation without implying a desirability of one operation over another. That is, it identifies an operation that is allowed while still maintaining compliance.

Table of Contents

| | | |
|----------|--|------------|
| 1 | INTRODUCTION..... | 5 |
| 2 | REFERENCES..... | 5 |
| 3 | GLOSSARY | 6 |
| 4 | ABSTRACT TEST METHOD..... | 6 |
| 4.1 | ABSTRACT TEST METHOD FOR HELLO SS_M AND HELLO SS_B TEST CASES..... | 6 |
| 4.2 | ABSTRACT TEST METHOD FOR DATABASE SYNCHRONIZATION TEST CASES | 6 |
| 4.3 | ABSTRACT TEST METHOD FOR FLOODING TEST CASES | 7 |
| 5 | TEST SUITE | 7 |
| 5.1 | HELLO SS_M AND HELLO SS_B CONFIGURATION..... | 8 |
| 5.2 | DATABASE SYNCHRONIZATION CONFIGURATION..... | 9 |
| 5.3 | FLOODING CONFIGURATION..... | 9 |
| 6 | EXPLANATION OF TEST SUITE..... | 10 |
| 6.1 | OVERVIEW PART | 10 |
| 6.2 | DECLARATION PART | 10 |
| 6.3 | CONSTRAINTS PART | 11 |
| 6.4 | DYNAMIC PART..... | 11 |
| 7 | TEST GROUPS OF THE TEST SUITE | 11 |
| 8 | ABSTRACT TEST SUITE FOR PNNI ROUTING..... | 11 |
| 9 | PROTOCOL IMPLEMENTATION EXTRA INFORMATION FOR TESTING (PIXIT) PROFORMA FOR THE ATS FOR PNNI ROUTING | 533 |

List of Figures

| | | |
|----------|--|----|
| FIGURE 1 | HELLO SS_M AND SS_B TEST CONFIGURATION..... | 6 |
| FIGURE 2 | DATABASE SYNCHRONIZATION TEST CONFIGURATION | 7 |
| FIGURE 3 | FLOODINGTEST CONFIGURATION..... | 7 |
| FIGURE 4 | PROTOCOL STACK FOR THE PNNI ROUTING PROTOCOL TESTS | 8 |
| FIGURE 5 | TEST CONFIGURATION FOR HELLO SS_M AND HELLO SS_B TESTS | 8 |
| FIGURE 6 | TEST CONFIGURATION FOR DATABASE SYNCHRONIZATIONTESTS | 9 |
| FIGURE 7 | TEST CONFIGURATION FOR FLOODING TESTS | 10 |

1 Introduction

This specification contains a draft version of conformance test suite for the routing protocol of the Private-Network-Network Interface (PNNI) protocol (v1.0) [af-pnni-0055.000]. This ATS covers Hello SS_M, Database synchronization and Flooding test cases for single peer group routing. In addition Hello SS_B test cases for multiple peer group routing are provided.

It has been developed according to the Protocol Implementation Conformance Statement (PICS) for PNNI v1.0 [af-pnni-0081.000].

This document specifies the ATS for PNNI routing protocol described in Tree and Tabular Combined Notation (TTCN) [ISO/IEC 9646-3]. This specification aligns with the principles of the ISO/IEC Conformance Testing Methodology and Framework (CTMF) [ISO/IEC 9646-1, ISO/IEC 9646-2].

2 References

- af-pnni-0055.000, ATM Forum Private Network-Network Interface Specification Version 1.0, March 1996.
- af-pnni-0081.000, ATM Forum PNNI v1.0 Errata and PICS, May 1997.
- ISO/IEC 9646-1, Information Technology – Open Systems Interconnection – Conformance Testing Methodology and Framework – Part 1: General Concepts), ISO/IEC, Geneva, 1991.
- ISO/IEC 9646-2, Information Technology – Open Systems Interconnection – Conformance Testing Methodology and Framework – Part 2: Abstract Test Suite Specification), ISO/IEC, Geneva, 1991.
- ISO/IEC 9646-3, Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 3: The Tree and Tabular Combined Notation (TTCN), ISO/IEC, Geneva, 1998.

3 Glossary

| | |
|-------|---|
| AAL | ATM Adaptation Layer |
| ATS | Abstract Test Suite |
| CP | Coordination Point |
| MTC | Main Test Component |
| PCO | Point of Control and Observation |
| PDU | Protocol Data Unit |
| PICS | Protocol Implementation Conformance Statement |
| PIXIT | Protocol Implementation eXtra Information for Testing |
| PNNI | Private Network Network Interface |
| PTC | Parallel Test Component |
| SUT | System Under Test |
| TTCN | Tree and Tabular Combined Notation |

4 Abstract Test Method

The Remote Single Layer Test Method is used for this test suite. The test suite uses up to four PCOs.

4.1 Abstract Test Method for Hello SS_M and Hello SS_B Test Cases

The test configuration used for testing PNNI Hello SS_M and SS_B routing is given in the figure below. The PCO belongs physically to RCC channel (VPI/VCI = 0/18) between System under Test (SUT) and Tester. Therefore, only one physical link is needed to perform the tests described in the ATS.

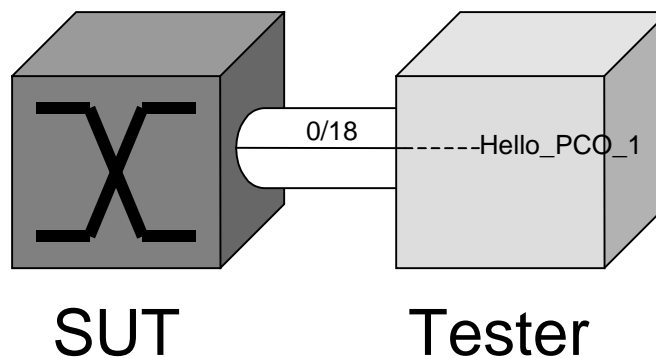


Figure 1: Hello SS_M and SS_B Test Configuration

4.2 Abstract Test Method for Database Synchronization Test Cases

The test configuration used for testing Database Synchronization in a PNNI single peer group is given in the figure below. The two PCOs belong physically to the channel 0/18 (VPI/VCI) between System under Test (SUT) and Tester. Therefore, only one physical link is needed to perform the tests described in the ATS.

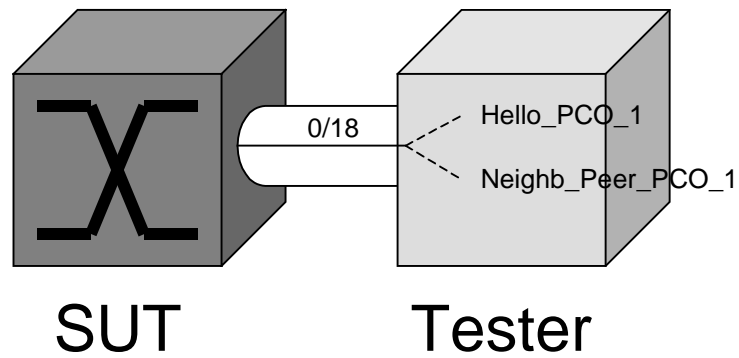


Figure 2: Database Synchronization Test Configuration

4.3 Abstract Test Method for Flooding Test Cases

The test configuration used for testing PNNI flooding is given in the figure below. The four PCOs belong physically to two RCC channels (VPI/VCI = 0/18) between System under Test (SUT) and Tester. Therefore, only two physical links are needed to perform the tests described in the ATS.

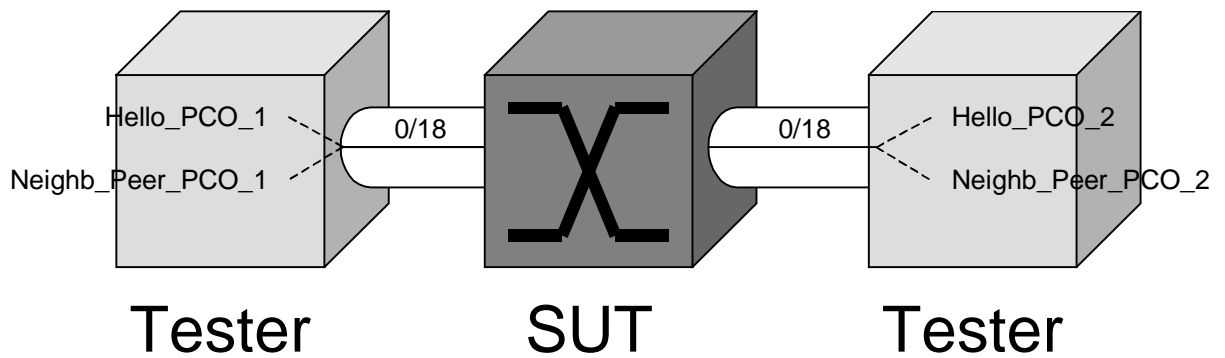


Figure 3: FloodingTest Configuration

5 Test Suite

The protocol stack for the PNNI routing tests is shown in the figure below. AAL5 is the underlying service provider for the test execution.

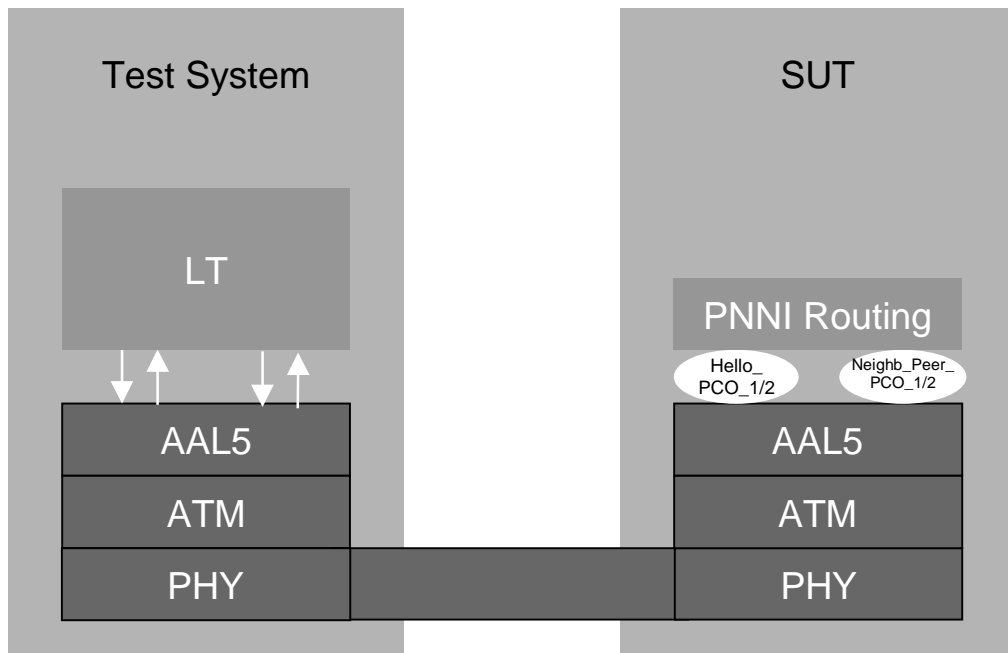


Figure 4: Protocol Stack for the PNNI Routing Protocol Tests

The test suite is written in Concurrent TTCN. A Main Test Component (MTC) controls one or more Parallel Test Components (PTCs) via Coordination Points (CPs).

5.1 Hello SS_M and Hello SS_B Configuration

The test configuration for the Hello SS_M and Hello SS_B tests is given in the figure below.

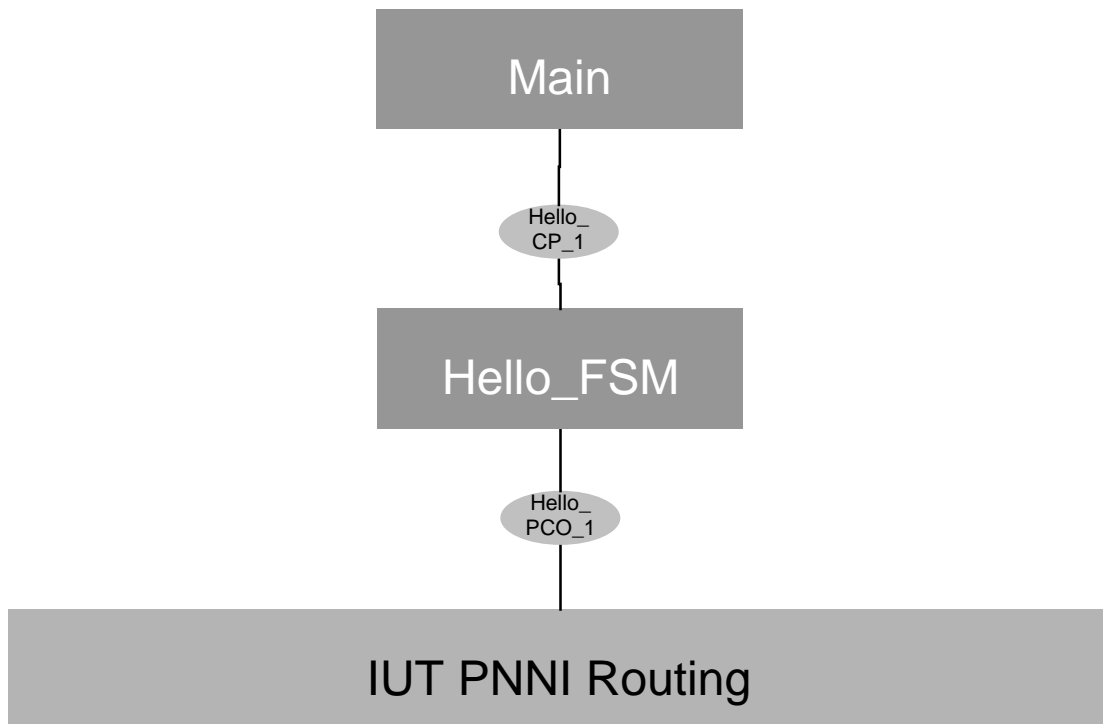


Figure 5: Test Configuration for Hello SS_M and Hello SS_B Tests

Hello_FSM is the parallel test component and emulates the behavior of the Hello finite state machine within the tester according to the PNNI routing protocol. This PTC is controlled by the main test component via the Hello_CP_1 coordination point. It has the capability to control and observe the protocol behaviors of the SUT through the PCO Hello_PCO_1 via the underlying service provider. The Hello_FSM PTC performs a lower tester role.

5.2 Database Synchronization Configuration

The test configuration for database synchronization tests is given in the figure below.

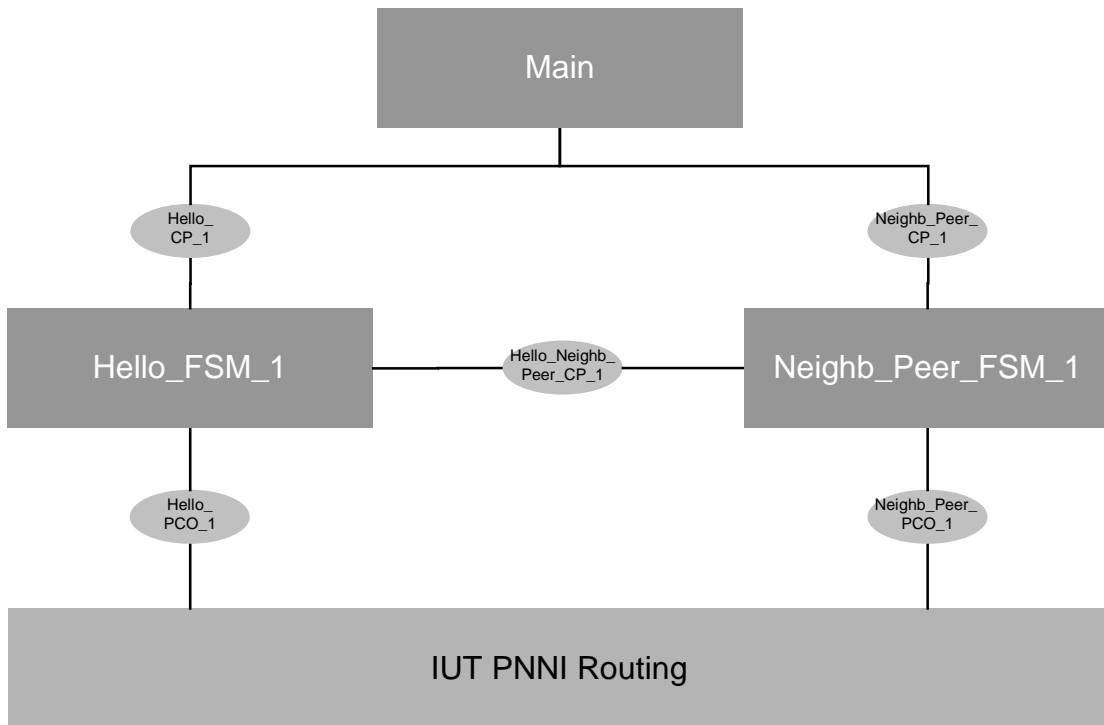


Figure 6: Test Configuration for Database Synchronization Tests

This test configuration uses in addition to Hello_FSM_1 a PTC Neighb_Peer_FSM_1 for emulating the behavior of the Neighbouring Peer finite state machine within the tester according to the PNNI routing protocol. This PTC is controlled by the MTC via the coordination point Neighb_Peer_CP_1 and coordinated with the PTC Hello_FSM_1 via the coordination point Hello_Neighb_Peer_CP_1. It has the capability to control and observe the protocol behaviors of the SUT through the PCO Neighb_Peer_PCO_1 via the underlying service provider. The Neighb_Peer_FSM_1 PTC is also in the lower tester role.

5.3 Flooding Configuration

The test configuration for the flooding tests is given in the figure below.

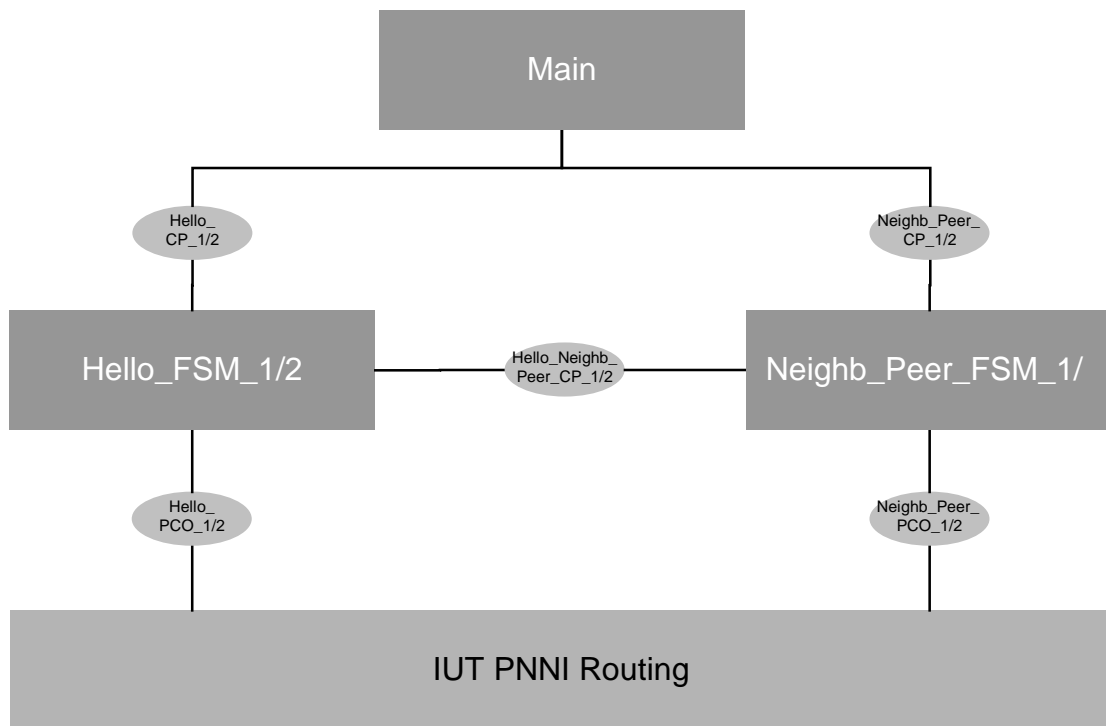


Figure 7: Test Configuration for Flooding Tests

Hello_FSM_1/2 are the parallel test components that emulate the behavior of the Hello finite state machines within the tester according to the PNNI routing protocol. This PTCs are controlled by the main test component via the Hello_CP_1/2 coordination points. They have the capability to control and observe the protocol behaviors of the SUT through the PCOs Hello_PCO_1/2 via the underlying service provider. The Hello_FSM_1/2 PTCs perform a lower tester role. The PTCs Neighb_Peer_FSM_1/2 emulate the behavior of the Neighboring Peer finite state machines within the tester according to the PNNI routing protocol. This PTCs are controlled by the MTC via the coordination points Neighb_Peer_CP_1/2 and coordinated with the PTCs Hello_FSM_1/2 via the coordination points Hello_Neighb_Peer_CP_1/2. They have the capability to control and observe the protocol behaviors of the SUT through the PCOs Neighb_Peer_PCO_1/2 via the underlying service provider. The Neighb_Peer_FSM_1/2 PTCs are also in the lower tester role.

6 Explanation of test suite

The ATS consists of the following four parts.

6.1 Overview part

The overview part of this ATS gives the information needed for general presentation and understanding of the test suite.

6.2 Declaration part

The declaration part of ATS gives the definitions and declarations of all components used in the test suite, that is, test suite constants, test suite variables, test case variables, test suite parameters, PCOs, CPs, timers, PDU types, test components and so forth.

6.3 Constraints part

The constraints part defines concrete values of the PDUs to be sent or received by the tester.

6.4 Dynamic part

The dynamic part of the ATS describes the detailed set of test cases. Each test case is composed of test steps for the preamble, test body, state check, and postamble.

7 Test Groups of the Test Suite

The test suite contains three main groups: one for the Hello protocol, one for the Database Synchronization protocol and one for Flooding. The two main groups Hello and DBSynchronization are further divided into subgroups.

The main test group Hello contains a subgroup for the SS_M test cases and one for the SS_B test cases. They are again divided into subgroups.

SS_M:

- general test cases (Hello/SS_M/GENERAL),
- test cases, where the Hello FSM is in state Down (Hello/SS_M/DOWN),
- test cases, where the Hello FSM is in state Attempt (Hello/SS_M/ATTEMPT),
- test cases, where the Hello FSM is in state OneWayInside (Hello/SS_M/ONE_WAY_INSIDE), and
- test cases, where the Hello FSM is in state TwoWayInside (Hello/SS_M/TWO_WAY_INSIDE).
- SS_B:general test cases (Hello/SS_B/GENERAL),
- test cases, where the Hello FSM is in state Attempt (Hello/SS_B/ATTEMPT),
- test cases, where the Hello FSM is in state OneWayOutside (Hello/SS_B/ONE_WAY_OUTSIDE), and
- test cases, where the Hello FSM is in state TwoWayOutside (Hello/SS_B/TWO_WAY_OUTSIDE).

The main test group DBSynchronization contains subgroups for

- test cases, where the Neighbouring Peer FSM is in state NPDown (DBSynchronization/NPDown),
- test cases, where the Neighbouring Peer FSM is in state Negotiating (DBSynchronization/Negotiating),
- test cases, where the Neighbouring Peer FSM is in state Exchanging (DBSynchronization/Exchanging),
- test cases, where the Neighbouring Peer FSM is in state Loading (DBSynchronization/Loading), and
- test cases, where the Neighbouring Peer FSM is in state Full (DBSynchronization/Full).

The main test group 'Flooding' contains no subgroup.

8 Abstract Test Suite for PNNI Routing

The Abstract test suite for the PNNI routing protocol is given next.

I

Test Suite Overview

| Test Suite Structure | | | |
|--|---------------|--|---------|
| <p>Suite Name : AF_TEST_0155_000</p> <p>Standards Ref : Private Network-Network Interface Specification Version 1.0 (PNNI 1.0) af-pnni-0055.000 March 1996</p> <p>PICS Ref : PNNI v1.0 Errata and PICS af-pnni-0081.000 May 1997</p> <p>PIXIT Ref :</p> <p>Test Method(s) : Remote Single Layer Test Method</p> <p>Comments : Non-hierarchical part \$Id: PNNI_ROUT.mp,v 1.18 2000/03/24 13:35:42 pie Exp \$</p> | | | |
| Test Group Reference | Selection Ref | Test Group Objective | Page Nr |
| Hello/ | | Verify the IUT's conformance to the PNNI Hello Protocol. (PNNI 1.0 5.6) | 249 |
| Hello/SS_M/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M. (PNNI 1.0 5.6) | 249 |
| Hello/SS_M/GENERAL/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M - general features. (PNNI 1.0 5.6) | 249 |
| Hello/SS_M/DOWN/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M - Hello state Down. (PNNI 1.0 5.6) | 255 |
| Hello/SS_M/ATTEMPT/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M - Hello state Attempt. (PNNI 1.0 5.6) | 255 |
| Hello/SS_M/ONE_WAY_INSIDE/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M - Hello state 1-Way Inside. (PNNI 1.0 5.6) | 258 |
| Hello/SS_M/TWO_WAY_INSIDE/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the minimum function switching system subset SS_M - Hello state 2-Way Inside. (PNNI 1.0 5.6) | 260 |
| Hello/SS_B/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the border node capable switching system subset SS_B. (PNNI 1.0 5.6) | 263 |
| Hello/SS_B/GENERAL/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the border node capable switching system subset SS_B - general features. (PNNI 1.0 5.6) | 263 |

Continued on next page

Continued from previous page

| Test Suite Structure | | | |
|--------------------------------|---------------|--|---------|
| Test Group Reference | Selection Ref | Test Group Objective | Page Nr |
| Hello/SS_B/ATTEMPT/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the border node capable switching system subset SS_B - Hello state Attempt. (PNNI 1.0 5.6) | 267 |
| Hello/SS_B/ONE_WAY_OUTSIDE/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the border node capable switching system subset SS_B - Hello state 1-Way Outside. (PNNI 1.0 5.6) | 269 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | | Verify the IUT's conformance to the PNNI Hello Protocol for the border node capable switching system subset SS_B - Hello state 2-Way Outside. (PNNI 1.0 5.6) | 271 |
| DBSynchronization/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol. (PNNI 1.0 5.7) | 273 |
| DBSynchronization/NPDown/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol - Neighboring Peer state NPDown. (PNNI 1.0 5.7) | 273 |
| DBSynchronization/Negotiating/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol - Neighboring Peer state Negotiating. (PNNI 1.0 5.7) | 274 |
| DBSynchronization/Exchanging/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol - Neighboring Peer state Exchanging. (PNNI 1.0 5.7) | 278 |
| DBSynchronization>Loading/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol - Neighboring Peer state Loading. (PNNI 1.0 5.7) | 287 |
| DBSynchronization/Full/ | | Verify the IUT's conformance to the PNNI Database Synchronization Protocol - Neighboring Peer state Full. (PNNI 1.0 5.7) | 299 |
| Flooding/ | | Verify the IUT's conformance to the PNNI Topology Description and Distribution Protocol - Flooding. (PNNI 1.0 5.8.3) | 307 |
| Detailed Comments : | | | |

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_M/GENERAL/ | Hello_V001 | SELECT_SS_M | To verify that the node includes the newest and oldest version supported fields in all packets. | 249 |
| Hello/SS_M/GENERAL/ | Hello_V002 | SELECT_SS_M | To verify that all versions in the range advertised are supported by the advertiser. | 249 |
| Hello/SS_M/GENERAL/ | Hello_V003 | SELECT_SS_M | To verify that the Inactivity timer is set to the value, InactivityFactor times the HelloInterval from the most recent Hello received from the neighbor. | 249 |
| Hello/SS_M/GENERAL/ | Hello_V004 | SELECT_SS_M | To verify that the Hello timer is restarted after an event-triggered Hello is transmitted. | 250 |
| Hello/SS_M/GENERAL/ | Hello_V005 | SELECT_SS_M | To verify that if a Hello has a top level unknown TLV with the mandatory tag bit set, that the Hello packet is discarded. | 250 |
| Hello/SS_M/GENERAL/ | Hello_V006 | SELECT_SS_M | To verify that if the Hello interval in the Hello packet is set to zero, that the Hello packet is discarded. | 250 |
| Hello/SS_M/GENERAL/ | Hello_V007 | SELECT_SS_M | To verify that if the port ID in the Hello packet is set to zero, that the Hello packet is discarded. | 251 |
| Hello/SS_M/GENERAL/ | Hello_V008_1 | SELECT_SS_M | To verify that when in state One Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 251 |
| Hello/SS_M/GENERAL/ | Hello_V008_2 | SELECT_SS_M | To verify that when in state Two Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 251 |
| Hello/SS_M/GENERAL/ | Hello_V008_3 | SELECT_SS_M | To verify that when in state Attempt, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 252 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_M/GENERAL/ | Hello_V009_1 | SELECT_SS_M | To verify that a hello is sent upon state change from Attempt to One Way Inside subject to the HoldDown timer. | 252 |
| Hello/SS_M/GENERAL/ | Hello_V009_2 | SELECT_SS_M | To verify that a Hello is sent upon state change from Attempt to Two Way Inside subject to the HoldDown timer. | 252 |
| Hello/SS_M/GENERAL/ | Hello_V009_3 | SELECT_SS_M | To verify that a Hello is sent upon state change from One Way Inside to Attempt subject to the HoldDown timer. | 253 |
| Hello/SS_M/GENERAL/ | Hello_V009_4 | SELECT_SS_M | To verify that a Hello is sent upon state change from Two Way Inside to Attempt subject to the HoldDown timer. | 253 |
| Hello/SS_M/GENERAL/ | Hello_V009_5 | SELECT_SS_M | To verify that a Hello is not sent upon state change from One Way Inside to Two Way Inside. | 253 |
| Hello/SS_M/GENERAL/ | Hello_V009_6 | SELECT_SS_M | To verify that a Hello is sent upon state change from Down to Attempt subject to the HoldDown timer. | 254 |
| Hello/SS_M/GENERAL/ | Hello_V010_1 | SELECT_SS_M | To verify that when in state One Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 254 |
| Hello/SS_M/GENERAL/ | Hello_V010_2 | SELECT_SS_M | To verify that when in state Two Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 254 |
| Hello/SS_M/DOWN/ | Hello_V101 | SELECT_SS_M | To verify that while in the Down state and a Link Up event is generated, that a Hello is sent and the Attempt state is entered. | 255 |
| Hello/SS_M/ATTEMPT/ | Hello_V201_1 | SELECT_SS_M | To verify that while in the Attempt state and a 1-Way Inside Received event is generated, sends a Hello, and enters the 1-Way Inside state. | 255 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|--------------|----------------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_M/ATTEMPT/ | Hello_V201_2 | SELECT_SS_M | To verify that while in the Attempt state and a 1-Way Inside Received event is generated, that the IUT starts the Inactivity timer, sends a Hello and restarts the Hello Timer. | 255 |
| Hello/SS_M/ATTEMPT/ | Hello_V202_1 | SELECT_SS_M | To verify that while in the Attempt state and a 2-Way Inside Received event is generated that a Hello is sent, and 2-Way Inside state is entered. | 256 |
| Hello/SS_M/ATTEMPT/ | Hello_V202_2 | SELECT_SS_M | To verify that while in the Attempt state and a 2-Way Inside Received event is generated that the Inactivity timer is restarted, a hello is sent, the Hello Timer is restarted. | 256 |
| Hello/SS_M/ATTEMPT/ | Hello_V203 | SELECT_SS_M_NOT_SS_B | To verify that while in the Attempt state and a Two Way Outside Received event is generated, the IUT does nothing. | 256 |
| Hello/SS_M/ATTEMPT/ | Hello_V204 | SELECT_SS_M_NOT_SS_B | To verify that while in the Attempt state and a Common Hierarchy Received event are generated that the IUT does nothing. | 257 |
| Hello/SS_M/ATTEMPT/ | Hello_V205 | SELECT_SS_M | To verify that while in the Attempt state and a Hello Mismatch Received event is generated, the IUT does nothing. | 257 |
| Hello/SS_M/ATTEMPT/ | Hello_V206 | SELECT_SS_M | To verify that when in the Attempt state, that the Hellos have their remote node ID and remote port ID set to zero. | 257 |
| Hello/SS_M/ATTEMPT/ | Hello_V207 | SELECT_SS_M_NOT_SS_B | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT does nothing. | 258 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V301 | SELECT_SS_M | To verify that while in the 1-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted. | 258 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V302_1 | SELECT_SS_M | To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that 2-Way Inside state is entered. | 258 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V302_2 | SELECT_SS_M | To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that the Inactivity Timer is restarted. | 259 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V303_1 | SELECT_SS_M | To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, and the Attempt state is entered. | 259 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V303_2 | SELECT_SS_M | To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Hello Timer is restarted. | 259 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V304_1 | SELECT_SS_M | To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 260 |
| Hello/SS_M/ONE_WAY_INSIDE/ | Hello_V304_2 | SELECT_SS_M | To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 260 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V401_1 | SELECT_SS_M | To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, a Hello is sent, and the 1-Way Inside state is entered. | 260 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V401_2 | SELECT_SS_M | To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted, a Hello is sent, that Hello Timer is restarted. | 261 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V402 | SELECT_SS_M | To verify that while in the 2-Way Inside state and a 2-Way Inside Received event is generated, that the Inactivity Timer is restarted. | 261 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|--------------|------------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V403_1 | SELECT_SS_M | To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, a Hello is sent, and the Attempt state is entered. | 261 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V403_2 | SELECT_SS_M | To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, Hello Timer is restarted. | 262 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V404_1 | SELECT_SS_M | To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 262 |
| Hello/SS_M/TWO_WAY_INSIDE/ | Hello_V404_2 | SELECT_SS_M | To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 262 |
| Hello/SS_B/GENERAL/ | Hello_V501_1 | SELECT_SS_M_SS_B | To verify that when in state One Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 263 |
| Hello/SS_B/GENERAL/ | Hello_V501_2 | SELECT_SS_M_SS_B | To verify that when in state Two Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 263 |
| Hello/SS_B/GENERAL/ | Hello_V502_1 | SELECT_SS_M_SS_B | To verify that a Hello is sent upon state change from Attempt to One Way Outside subject to the HoldDown timer. | 263 |
| Hello/SS_B/GENERAL/ | Hello_V502_2 | SELECT_SS_M_SS_B | To verify that a Hello is sent upon state change from Attempt to Two Way Outside subject to the HoldDown timer. | 264 |
| Hello/SS_B/GENERAL/ | Hello_V502_3 | SELECT_SS_M_SS_B | To verify that a Hello is sent upon state change from One Way Outside to Attempt subject to the HoldDown timer. | 264 |
| Hello/SS_B/GENERAL/ | Hello_V502_4 | SELECT_SS_M_SS_B | To verify that a Hello is sent upon state change from Two Way Outside to Attempt subject to the HoldDown timer. | 264 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|------------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_B/GENERAL/ | Hello_V502_5 | SELECT_SS_M_SS_B | To verify that a Hello is not sent upon state change from One Way Outside to Two Way Outside state. | 265 |
| Hello/SS_B/GENERAL/ | Hello_V503 | SELECT_SS_M_SS_B | To verify that when multiple event triggered Hellos are deferred because of the HoldDown timer, that the IUT sends only one Hello which contains the most current information for all IGs when the HoldDown timer expires. | 265 |
| Hello/SS_B/GENERAL/ | Hello_V504_1 | SELECT_SS_M_SS_B | To verify that when in state One Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 265 |
| Hello/SS_B/GENERAL/ | Hello_V504_2 | SELECT_SS_M_SS_B | To verify that when in state Two Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 266 |
| Hello/SS_B/GENERAL/ | Hello_V505 | SELECT_SS_M_SS_B | To verify that the sequence number of the first instance of the nodal hierarchy list sent to any neighbor is greater than zero. | 266 |
| Hello/SS_B/GENERAL/ | Hello_V506 | SELECT_SS_M_SS_B | To verify that if no higher level is known, that an empty nodal hierarchy list is included in the Hello. | 266 |
| Hello/SS_B/GENERAL/ | Hello_V507_1 | SELECT_SS_M_SS_B | To verify that the ULIA information group is included in all Hellos while in the states: 1-Way Outside. | 267 |
| Hello/SS_B/GENERAL/ | Hello_V507_2 | SELECT_SS_M_SS_B | To verify that the ULIA information group is included in all Hellos while in the states: 2-Way Outside. | 267 |
| Hello/SS_B/ATTEMPT/ | Hello_V601_1 | SELECT_SS_M_SS_B | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | 267 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|---------------------------------|--------------|------------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_B/ATTEMPT/ | Hello_V601_2 | SELECT_SS_M_SS_B | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 268 |
| Hello/SS_B/ATTEMPT/ | Hello_V602_1 | SELECT_SS_M_SS_B | To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 2-Way Outside state. | 268 |
| Hello/SS_B/ATTEMPT/ | Hello_V602_2 | SELECT_SS_M_SS_B | To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 268 |
| Hello/SS_B/ONE_WAY_O UTSIDE/ | Hello_V701 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and a 1-Way outside Received event is generated, that the Inactivity Timer is restarted. | 269 |
| Hello/SS_B/ONE_WAY_O UTSIDE/ | Hello_V702_1 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the IUT enters the 2-Way Outside state. | 269 |
| Hello/SS_B/ONE_WAY_O UTSIDE/ | Hello_V702_2 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | 269 |
| Hello/SS_B/ONE_WAY_O UTSIDE/ | Hello_V703_1 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | 270 |
| Hello/SS_B/ONE_WAY_O UTSIDE/ | Hello_V703_2 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | 270 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-----------------------------|--------------|------------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_B/ONE_WAY_OUTSIDE/ | Hello_V704_1 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 270 |
| Hello/SS_B/ONE_WAY_OUTSIDE/ | Hello_V704_2 | SELECT_SS_M_SS_B | To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 271 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V801_1 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | 271 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V801_2 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 271 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V802 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | 272 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V803_1 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | 272 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V803_2 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | 272 |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V804_1 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 273 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|--------------------------------|---------------|------------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_V804_2 | SELECT_SS_M_SS_B | To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 273 |
| DBSynchronization/NP Down/ | DBSync_V001_M | SELECT_SS_M | To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | 273 |
| DBSynchronization/NP Down/ | DBSync_V001_S | SELECT_SS_M | To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | 274 |
| DBSynchronization/Negotiating/ | DBSync_V101_M | SELECT_SS_M | To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. | 274 |
| DBSynchronization/Negotiating/ | DBSync_V101_S | SELECT_SS_M | To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. | 275 |
| DBSynchronization/Negotiating/ | DBSync_V102_M | SELECT_SS_M | To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. | 275 |
| DBSynchronization/Negotiating/ | DBSync_V102_S | SELECT_SS_M | To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. | 276 |
| DBSynchronization/Negotiating/ | DBSync_V103_M | SELECT_SS_M | To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. | 276 |
| DBSynchronization/Negotiating/ | DBSync_V103_S | SELECT_SS_M | To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. | 277 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|--------------------------------|---------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Negotiating/ | DBSync_V104_M | SELECT_SS_M | To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Master in the database synchronization. | 277 |
| DBSynchronization/Negotiating/ | DBSync_V105_S | SELECT_SS_M | To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Slave in the database synchronization. | 278 |
| DBSynchronization/Exchanging/ | DBSync_V201_M | SELECT_SS_M | To verify that if Master, the DSRxmt Timer is restarted when the node receives a correct Database Summary packet. | 278 |
| DBSynchronization/Exchanging/ | DBSync_V202_M | SELECT_SS_M | To verify that while in Exchanging when the node is Master, that Database Summary packets are sent when the Slave acknowledges the previous Database Summary packet and it has DS packets to send. | 279 |
| DBSynchronization/Exchanging/ | DBSync_V203_M | SELECT_SS_M | To verify that while in Exchanging when the node is Master and this packet includes the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to zero. | 279 |
| DBSynchronization/Exchanging/ | DBSync_V204_M | SELECT_SS_M | To verify that while in Exchanging when the node is Master and all of the database summary has already been sent to the Slave, that the More (M) bit in the Database Summary packet is set to zero and the contents are empty. | 280 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-------------------------------|---------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Exchanging/ | DBSync_V205_M | SELECT_SS_M | To verify that while in Exchanging when the node is Master and this packet does not include the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to one. | 280 |
| DBSynchronization/Exchanging/ | DBSync_V206_S | SELECT_SS_M | To verify that while in Exchanging when the node is Slave, that Database Summary packets are sent only in response to Database Summary packets received. | 281 |
| DBSynchronization/Exchanging/ | DBSync_V207_S | SELECT_SS_M | To verify that while in Exchanging when the node is Slave and all of the database summary has already been previously sent to the Master, that the More (M) bit in the Database Summary packet is set to zero. | 281 |
| DBSynchronization/Exchanging/ | DBSync_V208_S | SELECT_SS_M | To verify that while in Exchanging when the node is Slave and this packet contains at least one item of the database summary to be sent to the Master, that the more (M) bit is set to one. | 282 |
| DBSynchronization/Exchanging/ | DBSync_V209_M | SELECT_SS_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is not empty, that the DS Rxmt Timer is stopped and (thus) no more DS packets are sent. | 282 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-------------------------------|---------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Exchanging/ | DBSync_V210_M | SELECT_SS_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the DS Rxmt Timer is stopped, (thus) no more DS packets are sent. | 283 |
| DBSynchronization/Exchanging/ | DBSync_V211_M | SELECT_SS_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has not sent its entire database that a new Database Summary packet is sent and the DS Rxmt Timer is restarted. | 283 |
| DBSynchronization/Exchanging/ | DBSync_V212_M | SELECT_SS_M | To verify that while in Exchanging and the node is Master and a duplicate Database Summary packet is received, that the processing of this packet is stopped. | 284 |
| DBSynchronization/Exchanging/ | DBSync_V213_S | SELECT_SS_M | To verify that while in Exchanging and this node is Slave and the packet's DS sequence number is one more than this node's DS sequence number, that the packet is accepted. | 284 |
| DBSynchronization/Exchanging/ | DBSync_V214_S | SELECT_SS_M | To verify that while in Exchanging and this node is Slave and a duplicate Database Summary packet is received, that the last Database Summary packet sent to the Master is retransmitted. | 285 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-------------------------------|---------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Exchanging/ | DBSync_V215_M | SELECT_SS_M | To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. | 285 |
| DBSynchronization/Exchanging/ | DBSync_V215_S | SELECT_SS_M | To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. | 286 |
| DBSynchronization/Exchanging/ | DBSync_V216_M | SELECT_SS_M | To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. | 286 |
| DBSynchronization/Exchanging/ | DBSync_V216_S | SELECT_SS_M | To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. | 287 |
| DBSynchronization/Loading/ | DBSync_V301_M | SELECT_SS_M | To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. | 287 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Loading/ | DBSync_V301_S | SELECT_SS_M | To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. | 288 |
| DBSynchronization/Loading/ | DBSync_V302_1_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 288 |
| DBSynchronization/Loading/ | DBSync_V302_1_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 289 |
| DBSynchronization/Loading/ | DBSync_V302_2_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 289 |
| DBSynchronization/Loading/ | DBSync_V302_2_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 290 |
| DBSynchronization/Loading/ | DBSync_V302_3_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 290 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Loading/ | DBSync_V302_3_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 291 |
| DBSynchronization/Loading/ | DBSync_V303_1_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 291 |
| DBSynchronization/Loading/ | DBSync_V303_1_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 292 |
| DBSynchronization/Loading/ | DBSync_V303_2_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 292 |
| DBSynchronization/Loading/ | DBSync_V303_2_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 293 |
| DBSynchronization/Loading/ | DBSync_V303_3_M | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 293 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Loading/ | DBSync_V303_3_S | SELECT_SS_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 294 |
| DBSynchronization/Loading/ | DBSync_V304_1_M | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 294 |
| DBSynchronization/Loading/ | DBSync_V304_1_S | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 295 |
| DBSynchronization/Loading/ | DBSync_V304_2_M | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 295 |
| DBSynchronization/Loading/ | DBSync_V304_2_S | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 296 |
| DBSynchronization/Loading/ | DBSync_V304_3_M | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. | 296 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Loading/ | DBSync_V304_3_S | SELECT_SS_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. | 297 |
| DBSynchronization/Loading/ | DBSync_V305_M | SELECT_SS_M | To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. | 297 |
| DBSynchronization/Loading/ | DBSync_V305_S | SELECT_SS_M | To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. | 298 |
| DBSynchronization/Loading/ | DBSync_V306_M | SELECT_SS_M | To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list | 298 |
| DBSynchronization/Loading/ | DBSync_V306_S | SELECT_SS_M | To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list | 299 |
| DBSynchronization/Full/ | DBSync_V401_M | SELECT_SS_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the link is advertised in a PTSE. | 299 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Fu 11/ | DBSync_V401_S | SELECT_SS_M | To verify that while in Exchanging and the node is Slave, if a packet is received that has the DS sequence number one more than this node's own DS sequence number, the More bit set to zero and the just transmitted Database Summary packet had the M bit is set to zero and the PTSE Request List is empty, that the link is advertised in a PTSE. | 300 |
| DBSynchronization/Fu 11/ | DBSync_V402_1_M | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 300 |
| DBSynchronization/Fu 11/ | DBSync_V402_1_S | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 301 |
| DBSynchronization/Fu 11/ | DBSync_V402_2_M | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 301 |
| DBSynchronization/Fu 11/ | DBSync_V402_2_S | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 302 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-----------------------------|-----------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Fu 11/ | DBSync_V403_1_M | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 302 |
| DBSynchronization/Fu 11/ | DBSync_V403_1_S | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 303 |
| DBSynchronization/Fu 11/ | DBSync_V403_2_M | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 303 |
| DBSynchronization/Fu 11/ | DBSync_V403_2_S | SELECT_SS_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 304 |
| DBSynchronization/Fu 11/ | DBSync_V404_1_M | SELECT_SS_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 304 |
| DBSynchronization/Fu 11/ | DBSync_V404_1_S | SELECT_SS_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 305 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|-----------------------------|-----------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| DBSynchronization/Fu 11/ | DBSync_V404_2_M | SELECT_SS_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 305 |
| DBSynchronization/Fu 11/ | DBSync_V404_2_S | SELECT_SS_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 306 |
| Flooding/ | Fldg_V001 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node. | 307 |
| Flooding/ | Fldg_V002 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories. | 308 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V003 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 309 |
| Flooding/ | Fldg_V004 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories. | 310 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V005 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 (Cell Loss Ratio for CLP=0) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 311 |
| Flooding/ | Fldg_V006 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 312 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V007 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. | 313 |
| Flooding/ | Fldg_V008 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. | 314 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V009 | SELECT_GCAC_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. | 315 |
| Flooding/ | Fldg_V010 | SELECT_GCAC_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category. | 316 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V011 | SELECT_GCAC_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. | 317 |
| Flooding/ | Fldg_V012 | SELECT_GCAC_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category. | 318 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|-----------------|---|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V013 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 319 |
| Flooding/ | Fldg_V014 | SELECT_IRA_RA_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 320 |
| Flooding/ | Fldg_V015 | | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 321 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|-----------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V016 | SELECT_ERA_RA_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 322 |
| Flooding/ | Fldg_V017 | SELECT_ERA_TN_T | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. | 323 |
| Flooding/ | Fldg_V018 | | To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. | 324 |
| Flooding/ | Fldg_V019 | | To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. | 325 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V020 | | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. | 326 |
| Flooding/ | Fldg_V021 | | To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. | 327 |
| Flooding/ | Fldg_V022 | | To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node. | 328 |
| Flooding/ | Fldg_V023 | | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. | 329 |
| Flooding/ | Fldg_V024 | | To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. | 330 |
| Flooding/ | Fldg_V025 | | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. | 331 |

Continued on next page

Continued from previous page

| Test Case Index | | | | |
|----------------------|--------------|---------------|--|---------|
| Test Group Reference | Test Case Id | Selection Ref | Description | Page Nr |
| Flooding/ | Fldg_V026 | | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. | 332 |
| Detailed Comments : | | | | |

| Test Step Index | | | |
|----------------------------|------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V001 | To verify that the node includes the newest and oldest version supported fields in all packets. | 333 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V002 | To verify that all versions in the range advertised are supported by the advertiser. | 334 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V003 | To verify that the Inactivity timer is set to the value, InactivityFactor times the HelloInterval from the most recent Hello received from the neighbor. | 335 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V004 | To verify that the Hello timer is restarted after an event-triggered Hello is transmitted. | 336 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V005 | To verify that if a Hello has a top level unknown TLV with the mandatory tag bit set, that the Hello packet is discarded. | 337 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V006 | To verify that if the hello interval in the Hello packet is set to zero, that the Hello packet is discarded. | 337 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V007 | To verify that if the port ID in the Hello packet is set to zero, that the Hello packet is discarded. | 338 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V008_1 | To verify that when in state One Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 339 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V008_2 | To verify that when in state Two Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 340 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V008_3 | To verify that when in state Attempt, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 341 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_1 | To verify that a hello is sent upon state change from Attempt to One Way Inside subject to the HoldDown timer. | 342 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_2 | To verify that a Hello is sent upon state change from Attempt to Two Way Inside subject to the HoldDown timer. | 343 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_3 | To verify that a Hello is sent upon state change from One Way Inside to Attempt subject to the HoldDown timer. | 344 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_4 | To verify that a Hello is sent upon state change from Two Way Inside to Attempt subject to the HoldDown timer. | 345 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|----------------------------|------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_5 | To verify that a Hello is not sent upon state change from One Way Inside to Two Way Inside. | 345 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V009_6 | To verify that a Hello is sent upon state change from Down to Attempt subject to the HoldDown timer. | 346 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V010_1 | To verify that when in state One Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 346 |
| Bodies/Hello/SS_M/GENERAL/ | Hello_FSM_V010_2 | To verify that when in state Two Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 347 |
| Bodies/Hello/SS_M/DOWN/ | Hello_FSM_V101 | To verify that while in the Down state and a Link Up event is generated, that a Hello is sent and the Attempt state is entered. | 347 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V201_1 | To verify that while in the Attempt state and a 1-Way Inside Received event is generated, sends a Hello, and enters the 1-Way Inside state. | 348 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V201_2 | To verify that while in the Attempt state and a 1-Way Inside Received event is generated, that the IUT starts the Inactivity timer, sends a Hello and restarts the Hello Timer. | 349 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V202_1 | To verify that while in the Attempt state and a 2-Way Inside Received event is generated that a Hello is sent, and 2-Way Inside state is entered. | 350 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V202_2 | To verify that while in the Attempt state and a 2-Way Inside Received event is generated that the Inactivity timer is restarted, a hello is sent, the Hello Timer is restarted. | 351 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V203 | To verify that while in the Attempt state and a Two Way Outside Received event is generated, the IUT does nothing. | 352 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V204 | To verify that while in the Attempt state and a Common Hierarchy Received event are generated that the IUT does nothing. | 352 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V205 | To verify that while in the Attempt state and a Hello Mismatch Received event is generated, the IUT does nothing. | 353 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|-----------------------------------|------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V206 | To verify that when in the Attempt state, that the Hellos have their remote node ID and remote port ID set to zero. | 353 |
| Bodies/Hello/SS_M/ATTEMPT/ | Hello_FSM_V207 | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT does nothing. | 354 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V301 | To verify that while in the 1-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted. | 354 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V302_1 | To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that 2-Way Inside state is entered. | 355 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V302_2 | To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that the Inactivity Timer is restarted. | 355 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V303_1 | To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, and the Attempt state is entered. | 356 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V303_2 | To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Hello Timer is restarted. | 357 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V304_1 | To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 358 |
| Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | Hello_FSM_V304_2 | To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 359 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V401_1 | To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, a Hello is sent, and the 1-Way Inside state is entered. | 360 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V401_2 | To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted, a Hello is sent, that Hello Timer is restarted. | 361 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|-----------------------------------|------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V402 | To verify that while in the 2-Way Inside state and a 2-Way Inside Received event is generated, that the Inactivity Timer is restarted. | 362 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V403_1 | To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, a Hello is sent, and the Attempt state is entered. | 363 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V403_2 | To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, Hello Timer is restarted. | 364 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V404_1 | To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 365 |
| Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | Hello_FSM_V404_2 | To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 366 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V501_1 | To verify that when in state One Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 367 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V501_2 | To verify that when in state Two Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | 368 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V502_1 | To verify that a Hello is sent upon state change from Attempt to One Way Outside subject to the HoldDown timer. | 369 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V502_2 | To verify that a Hello is sent upon state change from Attempt to Two Way Outside subject to the HoldDown timer. | 370 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V502_3 | To verify that a Hello is sent upon state change from One Way Outside to Attempt subject to the HoldDown timer. | 371 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V502_4 | To verify that a Hello is sent upon state change from Two Way Outside to Attempt subject to the HoldDown timer. | 372 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V502_5 | To verify that a Hello is not sent upon state change from One Way Outside to Two Way Outside state. | 372 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|----------------------------|------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V503 | To verify that when multiple event triggered Hellos are deferred because of the HoldDown timer, that the IUT sends only one Hello which contains the most current information for all IGs when the HoldDown timer expires. | 373 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V504_1 | To verify that when in state One Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 374 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V504_2 | To verify that when in state Two Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | 374 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V505 | To verify that the sequence number of the first instance of the nodal hierarchy list sent to any neighbor is greater than zero. | 375 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V506 | To verify that if no higher level is known, that an empty nodal hierarchy list is included in the Hello. | 376 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V507_1 | To verify that the ULIA information group is included in all Hellos while in the states: 1-Way Outside. | 376 |
| Bodies/Hello/SS_B/GENERAL/ | Hello_FSM_V507_2 | To verify that the ULIA information group is included in all Hellos while in the states: 2-Way Outside. | 377 |
| Bodies/Hello/SS_B/ATTEMPT/ | Hello_FSM_V601_1 | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | 377 |
| Bodies/Hello/SS_B/ATTEMPT/ | Hello_FSM_V601_2 | To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 378 |
| Bodies/Hello/SS_B/ATTEMPT/ | Hello_FSM_V602_1 | To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 2-Way Outside state. | 379 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--|------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_B/ATTEMPT/ | Hello_FSM_V602_2 | To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 380 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V701 | To verify that while in the 1-Way Outside state and a 1-Way Outside Received event is generated, that the Inactivity Timer is restarted. | 381 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V702_1 | To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the IUT enters the 2-Way Outside state. | 382 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V702_2 | To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | 382 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V703_1 | To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | 383 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V703_2 | To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | 384 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V704_1 | To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 385 |
| Bodies/Hello/SS_B/ONE_WAY_0 UTSIDE/ | Hello_FSM_V704_2 | To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 386 |
| Bodies/Hello/SS_B/TWO_WAY_0 UTSIDE/ | Hello_FSM_V801_1 | To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | 387 |
| Bodies/Hello/SS_B/TWO_WAY_0 UTSIDE/ | Hello_FSM_V801_2 | To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | 388 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--|--------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_FSM_V802 | To verify that while in the 2-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | 389 |
| Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_FSM_V803_1 | To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | 390 |
| Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_FSM_V803_2 | To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | 391 |
| Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_FSM_V804_1 | To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | 392 |
| Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | Hello_FSM_V804_2 | To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | 393 |
| Bodies/DBSynchronization/NP Down/ | DBSync_FSM_V001_MS | To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | 394 |
| Bodies/DBSynchronization/Ne gotiating/ | DBSync_FSM_V101_MS | To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. | 394 |
| Bodies/DBSynchronization/Ne gotiating/ | DBSync_FSM_V102_MS | To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. | 395 |
| Bodies/DBSynchronization/Ne gotiating/ | DBSync_FSM_V103_MS | To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. | 396 |
| Bodies/DBSynchronization/Ne gotiating/ | DBSync_FSM_V104_M | To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Master in the database synchronization. | 397 |
| Bodies/DBSynchronization/Ne gotiating/ | DBSync_FSM_V105_S | To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Slave in the database synchronization. | 398 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--------------------------------------|-------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V201_M | To verify that if Master, the DSRxmt Timer is restarted when the node receives a correct Database Summary packet. | 399 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V202_M | To verify that while in Exchanging when the node is Master, that Database Summary packets are sent when the Slave acknowledges the previous Database Summary packet and it has DS packets to send. | 400 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V203_M | To verify that while in Exchanging when the node is Master and this packet includes the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to zero. | 401 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V204_M | To verify that while in Exchanging when the node is Master and all of the database summary has already been sent to the Slave, that the More (M) bit in the Database Summary packet is set to zero and the contents are empty. | 402 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V205_M | To verify that while in Exchanging when the node is Master and this packet does not include the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to one. | 403 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V206_S | To verify that while in Exchanging when the node is Slave, that Database Summary packets are sent only in response to Database Summary packets received. | 404 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V207_S | To verify that while in Exchanging when the node is Slave and all of the database summary has already been previously sent to the Master, that the More (M) bit in the Database Summary packet is set to zero. | 405 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V208_S | To verify that while in Exchanging when the node is Slave and this packet contains at least one item of the database summary to be sent to the Master, that the more (M) bit is set to one. | 406 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--------------------------------------|-------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V209_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is not empty, that the DS Rxmt Timer is stopped and (thus) no more DS packets are sent. | 407 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V210_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the DS Rxmt Timer is stopped, (thus) no more DS packets are sent. | 408 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V211_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has not sent its entire database that a new Database Summary packet is sent and the DS Rxmt Timer is restarted. | 409 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V212_M | To verify that while in Exchanging and the node is Master and a duplicate Database Summary packet is received, that the processing of this packet is stopped. | 410 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V213_S | To verify that while in Exchanging and this node is Slave and the packet's DS sequence number is one more than this node's DS sequence number, that the packet is accepted. | 411 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V214_S | To verify that while in Exchanging and this node is Slave and a duplicate Database Summary packet is received, that the last Database Summary packet sent to the Master is retransmitted. | 412 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--------------------------------------|---------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V215_M | To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. | 413 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V215_S | To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. | 414 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V216_M | To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. | 415 |
| Bodies/DBSynchronization/Exchanging/ | DBSync_FSM_V216_S | To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. | 416 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V301_M | To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. | 417 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V301_S | To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. | 417 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_1_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 418 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|-----------------------------------|---------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_1_S | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 419 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_2_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 420 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_2_S | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 421 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_3_M | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 421 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V302_3_S | To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 422 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_1_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 423 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_1_S | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 424 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_2_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 425 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|-----------------------------------|---------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_2_S | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 426 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_3_M | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 426 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V303_3_S | To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 427 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_1_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 428 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_1_S | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 429 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_2_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 430 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_2_S | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. | 430 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_3_M | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. | 431 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|-----------------------------------|---------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V304_3_S | To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. | 431 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V305_M | To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. | 432 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V305_S | To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. | 433 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V306_M | To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list | 434 |
| Bodies/DBSynchronization/Loading/ | DBSync_FSM_V306_S | To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list | 435 |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V401_M | To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the link is advertised in a PTSE. | 435 |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V401_S | To verify that while in Exchanging and the node is Slave, if a packet is received that has the DS sequence number one more than this node's own DS sequence number, the More bit set to zero and the just transmitted Database Summary packet had the M bit is set to zero and the PTSE Request List is empty, that the link is advertised in a PTSE. | 436 |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V402_1_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 437 |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V402_1_S | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 438 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--------------------------------|---------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V402_2_M | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 439 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V402_2_S | To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 439 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V403_1_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 440 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V403_1_S | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 441 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V403_2_M | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 442 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V403_2_S | To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 442 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V404_1_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 443 |
| Bodies/DBSynchronization/Fu11/ | DBSync_FSM_V404_1_S | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. | 444 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V404_2_M | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 445 |
| Bodies/DBSynchronization/Full/ | DBSync_FSM_V404_2_S | To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | 445 |
| Bodies/DBSynchronization/Hello_FSM_Emulation/ | Hello_FSM_Emull | Emulate the Hello FSM for Database Synchronization Tests | 446 |
| Bodies/Flooding/ | Fldg_FSM_V001_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node. | 447 |
| Bodies/Flooding/ | Fldg_FSM_V001_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node. | 448 |
| Bodies/Flooding/ | Fldg_FSM_V002_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories. | 449 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V002_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories. | 450 |
| Bodies/Flooding/ | Fldg_FSM_V003_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 451 |
| Bodies/Flooding/ | Fldg_FSM_V003_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 452 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V004_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories. | 453 |
| Bodies/Flooding/ | Fldg_FSM_V004_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories. | 454 |
| Bodies/Flooding/ | Fldg_FSM_V005_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 (Cell Loss Ratio for CLP=0) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 455 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V005_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 456 |
| Bodies/Flooding/ | Fldg_FSM_V006_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 457 |
| Bodies/Flooding/ | Fldg_FSM_V006_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. | 458 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V007_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. | 459 |
| Bodies/Flooding/ | Fldg_FSM_V007_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. | 460 |
| Bodies/Flooding/ | Fldg_FSM_V008_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. | 461 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V008_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. | 462 |
| Bodies/Flooding/ | Fldg_FSM_V009_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. | 463 |
| Bodies/Flooding/ | Fldg_FSM_V009_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. | 464 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V010_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category. | 465 |
| Bodies/Flooding/ | Fldg_FSM_V010_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category. | 466 |
| Bodies/Flooding/ | Fldg_FSM_V011_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. | 467 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V011_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. | 468 |
| Bodies/Flooding/ | Fldg_FSM_V012_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category. | 469 |
| Bodies/Flooding/ | Fldg_FSM_V012_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category. | 470 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V013_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included: <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 471 |
| Bodies/Flooding/ | Fldg_FSM_V013_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included: <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 472 |
| Bodies/Flooding/ | Fldg_FSM_V014_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 473 |
| Bodies/Flooding/ | Fldg_FSM_V014_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 474 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V015_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included: <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 475 |
| Bodies/Flooding/ | Fldg_FSM_V015_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included: <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. | 476 |
| Bodies/Flooding/ | Fldg_FSM_V016_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 477 |
| Bodies/Flooding/ | Fldg_FSM_V016_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present. | 478 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V017_First | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. | 479 |
| Bodies/Flooding/ | Fldg_FSM_V017_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. | 480 |
| Bodies/Flooding/ | Fldg_FSM_V018_First | To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. | 481 |
| Bodies/Flooding/ | Fldg_FSM_V018_Second | To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. | 482 |
| Bodies/Flooding/ | Fldg_FSM_V019_First | To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. | 484 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|---------------------------|----------------------|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V019_Second | To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. | 485 |
| Bodies/Flooding/ | Fldg_FSM_V020_First | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. | 486 |
| Bodies/Flooding/ | Fldg_FSM_V020_Second | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. | 487 |
| Bodies/Flooding/ | Fldg_FSM_V021_First | To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. | 488 |
| Bodies/Flooding/ | Fldg_FSM_V021_Second | To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. | 489 |
| Bodies/Flooding/ | Fldg_FSM_V022_First | To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node. | 490 |
| Bodies/Flooding/ | Fldg_FSM_V022_Second | To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node. | 491 |
| Bodies/Flooding/ | Fldg_FSM_V023_First | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. | 492 |
| Bodies/Flooding/ | Fldg_FSM_V023_Second | To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. | 493 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|----------------------------------|----------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/ | Fldg_FSM_V024_First | To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. | 494 |
| Bodies/Flooding/ | Fldg_FSM_V024_Second | To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. | 495 |
| Bodies/Flooding/ | Fldg_FSM_V025_First | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. | 497 |
| Bodies/Flooding/ | Fldg_FSM_V025_Second | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. | 498 |
| Bodies/Flooding/ | Fldg_FSM_V026_First | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. | 499 |
| Bodies/Flooding/ | Fldg_FSM_V026_Second | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. | 500 |
| Bodies/Flooding/Fldg_Unexpected/ | Fldg_Unexpected | To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. | 501 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|--------------------------------------|---|---|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Bodies/Flooding/Fldg_Unexpected/ | Fldg_Unexpected_Second | | 501 |
| Bodies/Flooding/Hello_FSM_Emulation/ | TwoLinks_Hello_FSM_Emulation/ | Emulate the Hello FSM | 502 |
| CheckState/ | CheckStateNeighbNegotiating | | 503 |
| CheckState/ | CheckStateHelloAttempt | | 504 |
| CheckState/ | CheckStateHelloOWI | Check the IUT state One Way Inside at the end of the test | 504 |
| CheckState/ | CheckStateHelloOWO | Check the IUT state One Way Outside at the end of the test | 504 |
| CheckState/ | CheckStateHelloTWI | Check the IUT state Two Way Inside at the end of the test | 505 |
| CheckState/ | CheckStateHelloTWO | Check the IUT states Two Way Outside at the end of the test | 505 |
| Postambles/ | PostambleDBSync_PTCs | Postamble for all DBSynchronization Test Cases | 506 |
| Postambles/ | PostambleHelloAttempt | To bring the IUT back to Attempt | 506 |
| Postambles/ | PostambleNeighb_Peer_Premature_PTSE_aging | Premature aging of the PTSE sent to the IUT | 507 |
| Postambles/PostTwoLinks/ | PostTwoLinks_Fldg | Postamble for all Flooding Test Cases | 507 |
| Postambles/PostTwoLinks/ | PostTwoLinks_HelloAttempt | To bring the IUT back to Attempt | 508 |
| Postambles/PostTwoLinks/ | PostTwoLinks_Neighb_Peer_Premature_PTSE_aging | Premature aging of the PTSE sent to the IUT | 508 |
| Preambles/PreTwoLinks/ | PreTwoLinks_PreambleInit | To perform some initialization. | 508 |
| Preambles/PreTwoLinks/ | PreTwoLinks_HelloDown | To start the IUT Hello-FSM | 509 |
| Preambles/PreTwoLinks/ | PreTwoLinks_HelloAttempt | To bring the IUT in the Hello state Attempt. | 509 |
| Preambles/PreTwoLinks/ | PreTwoLinks_NeighbNegotiating | To bring the IUT into the Neighbouring Peer state Negotiating | 509 |
| Preambles/PreTwoLinks/ | PreTwoLinks_NeighbExchanging_M | To bring the IUT into the Neighboring Peer state Exchanging as Master | 510 |
| Preambles/PreTwoLinks/ | PreTwoLinks_NeighbExchanging_S | To bring the IUT into the Neighbouring Peer state Exchanging as Slave | 510 |
| Preambles/PreTwoLinks/ | PreTwoLinks_NeighbFull_M | To bring the IUT into the Neighboring Peer state Full as Master | 511 |
| Preambles/PreTwoLinks/ | PreTwoLinks_NeighbFull_S | To bring the IUT into the Neighboring Peer state Full as Slave. | 512 |
| Preambles/PreTwoLinks/ | PreTwoLinks_Fldg_First | To bring the IUT into the state before flooding begins | 512 |
| Preambles/PreTwoLinks/ | PreTwoLinks_Fldg_Second | To bring the IUT into the state before flooding begins | 513 |
| Preambles/ | PreambleHelloAttempt | To bring the IUT in the Hello state Attempt. | 513 |
| Preambles/ | PreambleHelloDown | To start the IUT Hello-FSM | 514 |
| Preambles/ | PreambleHelloOWI | To bring the IUT in the Hello state One-Way-Inside. | 514 |
| Preambles/ | PreambleHelloOWO | To bring the IUT in the Hello state One-Way-Outside. | 515 |

Continued on next page

Continued from previous page

| Test Step Index | | | |
|----------------------------|-------------------------------------|--|---------|
| Test Step Group Reference | Test Step Id | Description | Page Nr |
| Preambles/ | PreambleHelloTWI | To bring the IUT in the Hello state Two-Way-Inside. | 515 |
| Preambles/ | PreambleHelloTWO | To bring the IUT in the Hello state Two-Way-Outside. | 516 |
| Preambles/ | PreambleInit | To perform some initialization. | 516 |
| Preambles/ | PreambleNeighbExchanging_M | To bring the IUT into the Neighboring Peer state Exchanging as Master | 517 |
| Preambles/ | PreambleNeighbExchanging_S | To bring the IUT into the Neighbouring Peer state Exchanging as Slave | 517 |
| Preambles/ | PreambleNeighbFull_LinkAdvertized_M | To bring the IUT into the Neighboring Peer state Full as Master and acknowledge the link advertizement | 518 |
| Preambles/ | PreambleNeighbFull_LinkAdvertized_S | To bring the IUT into the Neighboring Peer state Full as Slave and acknowledge the link advertizement | 518 |
| Preambles/ | PreambleNeighbFull_M | To bring the IUT into the Neighboring Peer state Full as Master | 519 |
| Preambles/ | PreambleNeighbFull_S | To bring the IUT into the Neighboring Peer state Full as Slave. | 519 |
| Preambles/ | PreambleNeighbLoading_M | To bring the IUT into the Neighboring Peer state Loading as Master | 520 |
| Preambles/ | PreambleNeighbLoading_S | To bring the IUT into the Neighboring Peer state Loading as Slave. | 521 |
| Preambles/ | PreambleNeighbNegotiating | To bring the IUT into the Neighbouring Peer state Negotiating | 521 |
| Preambles/ | PreambleHelloCO | To bring the IUT in the Hello state Common-Outside. | 522 |
| Detailed Comments : | | | |

| Default Index | | | |
|-------------------------|--|---|---------|
| Default Group Reference | Default Id | Description | Page Nr |
| DefTwoLinks/ | DefTwoLinks_Fldg_MTC_one | Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_Flooding test component configurations. | 523 |
| DefTwoLinks/ | DefTwoLinks_Fldg_MTC_two | Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_Flooding test component configurations. | 524 |
| DefTwoLinks/ | DefTwoLinks_Hello_Emul | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC and Neighb_Peer_FSM. | 525 |
| DefTwoLinks/ | DefTwoLinks_Hello_PTC_Preamble | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | 526 |
| DefTwoLinks/ | DefTwoLinks_Neighb_Peer_PTC | Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | 526 |
| DefTwoLinks/ | DefTwoLinks_Neighb_Peer_PTC_Preamble | Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | 527 |
| DefTwoLinks/ | DefTwoLinks_Neighb_Peer_PTC_with_Postamble | Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | 527 |
| | Default_DBSync_MTC | Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_DBSync test component configurations. | 528 |
| | Default_Hello_Emull | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC and Neighb_Peer_FSM. | 529 |
| | Default_Hello_MTC | Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs. | 530 |
| | Default_Hello_PTC | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | 530 |
| | Default_Hello_PTC_Preamble | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | 531 |
| | Default_Neighb_Peer_PTC | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | 531 |

Continued on next page

Continued from previous page

| Default Index | | | |
|----------------------------|--|--|---------|
| Default Group Reference | Default Id | Description | Page Nr |
| | Default_Neighb_Peer_PTC_Preamble | Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | 532 |
| | Default_Neighb_Peer_PTC_with_Postamble | Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | 532 |
| Detailed Comments : | | | |

II

Declarations Part

| ASN.1 Type Definition | |
|---|------------------------|
| Type Name | : AggregationTokenIG_T |
| Encoding Variation | : |
| Comments | : Aggregation Token IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 32 (AT) length OCTET STRING (SIZE(2)), -- Length token OCTET STRING (SIZE(4)) -- Aggregation Token }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.1 |

| ASN.1 Type Definition | |
|--|-----------------------------------|
| Type Name | : Cause_T |
| Encoding Variation | : |
| Comments | : Causes for Termination Requests |
| Type Definition | |
| <pre>ENUMERATED { MTC_FAILURE(1), -- Failure in the MTC (e.g. TIMEOUT) DBSync_DONE(2), -- PTC DBSync_FSM is DONE Hello_DONE(3) -- PTC Hello_FSM is DONE }</pre> | |
| Detailed Comments | : |

| ASN.1 Type Definition | |
|---|---------------------------------|
| Type Name | : DBSum_Packet_Flags_T |
| Encoding Variation | : |
| Comments | : Database Summary Packet Flags |
| Type Definition | |
| <pre>SEQUENCE { bit_16_init BIT STRING (SIZE(1)), -- Bit 16 (MSB) : 'Initialize' (I) bit bit_15_more BIT STRING (SIZE(1)), -- Bit 15: 'More' (M) bit: bit_14_master BIT STRING (SIZE(1)), -- Bit 14: 'Master' (MS) bit bit_13_1_res BIT STRING (SIZE(13)) -- Bits 13..1 (LSB): Reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Definition | |
|---|---|
| Type Name | : ExteriorReachableATMAddressesIG_SUB_T |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Type Definition | |
| <pre>SEQUENCE { prefix_len OCTET STRING (SIZE(1)), -- Prefix Length addr_prefix OCTET STRING, -- Reachable Address Prefix padding OCTET STRING (SIZE(0..3)), -- Padding tlv_groups SEQUENCE OF RAIG_SUB_T OPTIONAL, -- Optional TLV groups for resource availability information add_tlv_gr TransitNetworkIdIG_T OPTIONAL -- Transit Network IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Definition | |
|---|---------------------------------------|
| Type Name | : ExteriorReachableATMAddressesIG_T |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 256 length OCTET STRING (SIZE(2)), -- Length flags VP_Capabilities_Flag_T, -- VP Capability Flag reserved OCTET STRING (SIZE(2)), -- Reserved port OCTET STRING (SIZE(4)), -- Port ID scope OCTET STRING (SIZE(1)), -- Scope of advertisement ail OCTET STRING (SIZE(1)), -- Address Information Length (in octets) aic OCTET STRING (SIZE(2)), -- Address Information Count addr SEQUENCE OF ExteriorReachableATMAddressesIG_SUB_T -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Definition | |
|---|----------------------------|
| Type Name | : Hello_IGs_T |
| Encoding Variation | : |
| Comments | : Hello Information Groups |
| Type Definition | |
| <pre>SET { aggr_token AggregationTokenIG_T OPTIONAL, -- Aggregation Token nhl NodalHierarchyListIG_T OPTIONAL, -- Nodal Hierarchy List ulia UplinkInformationAttributeIG_T OPTIONAL, -- Uplink Information Attribute lgn_hl_ext LGN_HorizontalLinkExtensionIG_T OPTIONAL, -- LGN Horizontal Link Extension sys_cap SET OF SystemCapabilitiesIG_T OPTIONAL -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 |

| ASN.1 Type Definition | |
|--|-----------------------|
| Type Name | : HorizontalLinksIG_T |
| Encoding Variation | : |
| Comments | : Horizontal Links IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 288 length OCTET STRING (SIZE(2)), -- Length flags VP_Capabilities_Flag_T, -- VP Capability Flag r_node OCTET STRING (SIZE(22)), -- Remote Node ID r_port OCTET STRING (SIZE(4)), -- Remote Port ID l_port OCTET STRING (SIZE(4)), -- Local Port ID token OCTET STRING (SIZE(4)), -- Aggregation Token out_raig SEQUENCE OF ResourceAvailabilityIG_T -- Outgoing Resource Availability } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.9.1.5 | |

| ASN.1 Type Definition | |
|---|--------------------------|
| Type Name | : IG_TagBit_T |
| Encoding Variation | : |
| Comments | : IG Tag Bit Definitions |
| Type Definition | |
| <pre>SEQUENCE { mand BIT STRING (SIZE(1)), -- Mandatory Bit d_sum BIT STRING (SIZE(1)), -- Don't Summarize Bit trans BIT STRING (SIZE(1)), -- Transitive Bit reserved BIT STRING (SIZE(1)) -- Reserved Bit } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.2.6 | |

| ASN.1 Type Definition | |
|---|-------------|
| Type Name | : IG_Type_T |
| Encoding Variation | : |
| Comments | : IG Type |
| Type Definition | |
| <pre>SEQUENCE { tag IG_TagBit_T, -- IG Tag Bit Definitions type BIT STRING (SIZE(12)) -- IG Type } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.3 | |

| ASN.1 Type Definition | |
|--|---|
| Type Name | : InternalReachableATMAddressesIG_SUB_T |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Type Definition | |
| <pre>SEQUENCE { prefix_len OCTET STRING (SIZE(1)), -- Prefix Length addr_prefix OCTET STRING, -- Reachable Address Prefix padding OCTET STRING (SIZE(0..3)), -- Padding tlv_groups SET OF RAIG_SUB_T OPTIONAL -- Optional TLV groups for resource availability information } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Definition | |
|---|---------------------------------------|
| Type Name | : InternalReachableATMAddressesIG_T |
| Encoding Variation | : |
| Comments | : Internal Reachable ATM Addresses IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 224 length OCTET STRING (SIZE(2)), -- Length flags VP_Capabilities_Flag_T, -- VP Capability Flag reserved OCTET STRING (SIZE(2)), -- Reserved port OCTET STRING (SIZE(4)), -- Port ID scope OCTET STRING (SIZE(1)), -- Scope of advertisement ail OCTET STRING (SIZE(1)), -- Address Information Length (in octets) aic OCTET STRING (SIZE(2)), -- Address Information Count addr SEQUENCE OF InternalReachableATMAddressesIG_SUB_T -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Definition | |
|---|--|
| Type Name | : LGN_HorizontalLinkExtensionIG_SUB_T |
| Encoding Variation | : |
| Comments | : LGN Horizontal Link Extension IG Subtype |
| Type Definition | |
| <pre>SEQUENCE { token OCTET STRING (SIZE(4)), -- Aggregation Token l_port OCTET STRING (SIZE(4)), -- Local LGN Port r_port OCTET STRING (SIZE(4)) -- Remote LGN Port } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.3 |

| ASN.1 Type Definition | |
|---|------------------------------------|
| Type Name | : LGN_HorizontalLinkExtensionIG_T |
| Encoding Variation | : |
| Comments | : LGN Horizontal Link Extension IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 35 (LGN_HLE) length OCTET STRING (SIZE(2)), -- Length reserved OCTET STRING (SIZE(2)), -- Reserved count OCTET STRING (SIZE(2)), -- Horizontal Link Count hl_ext_sub SEQUENCE OF LGN_HorizontalLinkExtensionIG_SUB_T -- LGN Horizontal Link Extension IG Subtype } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.3 |

| ASN.1 Type Definition | |
|---|-------------------------------|
| Type Name | : NetworkIdentificationData_T |
| Encoding Variation | : |
| Comments | : Network identification data |
| Type Definition | |
| <pre>SEQUENCE { bit_8_res BIT STRING (SIZE(1)), -- Reserved bit_7_5_type BIT STRING (SIZE(3)), -- Type of network identification bit_4_1_plan BIT STRING (SIZE(4)) -- Network identification plan } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.7 |

| ASN.1 Type Definition | |
|--|--------------------------------|
| Type Name | : NextHigherLevelBindingIG_T |
| Encoding Variation | : |
| Comments | : Next Higher Level Binding IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 192 (NHL IG) length OCTET STRING (SIZE(2)), -- Length parent_id OCTET STRING (SIZE(22)), -- Parent LGN ID (in parent peer group) parent_addr OCTET STRING (SIZE(20)), -- Parent LGN's ATM End System Address parent_pg OCTET STRING (SIZE(14)), -- Parent Peer Group ID parent_pgl OCTET STRING (SIZE(22)), -- Node ID of PGL of parent peer group -- (NULL if unknown) reserved OCTET STRING (SIZE(2)) } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Definition | |
|--|-----------------------------------|
| Type Name | : NodalHierarchyListIG_SUB_T |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG Subtype |
| Type Definition | |
| <pre>SEQUENCE { nhl_node OCTET STRING (SIZE(22)), -- Next Higher Level Logical Node ID nhl_addr OCTET STRING (SIZE(20)), -- Next Higher Level ATM End System Address nhl_pg OCTET STRING (SIZE(14)) -- Next Higher Level Peer Group ID }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.2 |

| ASN.1 Type Definition | |
|--|---------------------------|
| Type Name | : NodalHierarchyListIG_T |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 33 (NHL) length OCTET STRING (SIZE(2)), -- Length seq_num OCTET STRING (SIZE(4)), -- Sequence Number reserved OCTET STRING (SIZE(2)), -- Reserved level OCTET STRING (SIZE(2)), -- Level Count nhl_sub SEQUENCE OF NodalHierarchyListIG_SUB_T -- Nodal Hierarchy List IG Subtype }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.2 |

| ASN.1 Type Definition | |
|---|-----------------------|
| Type Name | : NodalIG_T |
| Encoding Variation | : |
| Comments | : Nodal IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 97 (Nodal IG) length OCTET STRING (SIZE(2)), -- Length addr OCTET STRING (SIZE(20)), -- ATM End System Address of the originating node lead_prio OCTET STRING (SIZE(1)), -- Leadership Priority -- (zero = unwilling/unable to become PGL) nodal_flags Nodal_Flags_T, -- Nodal Flags pref_pgl OCTET STRING (SIZE(22)), -- Preferred PGL node ID nhl_bind NextHigherLevelBindingIG_T OPTIONAL -- NHL IG sent by PGLs if there is a higher level }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Definition | |
|---|------------------------------------|
| Type Name | : NodalPTSEAcknowledgementIG_SUB_T |
| Encoding Variation | : |
| Comments | : PTSE Identifier |
| Type Definition | |
| <pre>SEQUENCE { ptse_ident OCTET STRING (SIZE(4)), -- PTSE Identifier ptse_seq_no OCTET STRING (SIZE(4)), -- PTSE Sequence Number ptse_crc OCTET STRING (SIZE(2)), -- PTSE Checksum ptse_ttl OCTET STRING (SIZE(2)) -- PTSE Remaining Lifetime } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Definition | |
|--|---------------------------------|
| Type Name | : NodalPTSEAcknowledgementIG_T |
| Encoding Variation | : |
| Comments | : Nodal PTSE Acknowledgement IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 384 length OCTET STRING (SIZE(2)), -- Length node OCTET STRING (SIZE(22)), -- Node ID ack_count OCTET STRING (SIZE(2)), -- PTSE Acknowledgement Count ptse_id SET OF NodalPTSEAcknowledgementIG_SUB_T -- PTSE Identifier } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Definition | |
|---|------------------------------|
| Type Name | : NodalPTSERequestListIG_T |
| Encoding Variation | : |
| Comments | : Nodal PTSE Request List IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 513 length OCTET STRING (SIZE(2)), -- Length o_node OCTET STRING (SIZE(22)), -- Originating Node ID req_count OCTET STRING (SIZE(2)), -- PTSE Request Count ptse_id SEQUENCE OF OCTET STRING (SIZE(4)) -- PTSE Identifier } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.12 |

| ASN.1 Type Definition | |
|---|--|
| Type Name | : NodalPTSESummariesIG_SUB_T |
| Encoding Variation | : |
| Comments | : Sequence of PNNI Topology State Elements (PTSE) header information |
| Type Definition | |
| <pre>SEQUENCE { ptse_type OCTET STRING (SIZE(2)), -- PTSEType reserved OCTET STRING (SIZE(2)), -- ptse_ident OCTET STRING (SIZE(4)), -- PTSE Identifier ptse_seq_no OCTET STRING (SIZE(4)), -- PTSE Sequence Number ptse_crc OCTET STRING (SIZE(2)), -- PTSE Checksum ptse_ttl OCTET STRING (SIZE(2)) -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Definition | |
|---|---|
| Type Name | : NodalPTSESummariesIG_T |
| Encoding Variation | : |
| Comments | : Sequence of PNNI Topology State Packets (PTSP) and PNNI Topology State Elements (PTSE) header information |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 512 (Nodal PTSE summaries) length OCTET STRING (SIZE(2)), -- o_node OCTET STRING (SIZE(22)), -- Originating Node ID o_pg OCTET STRING (SIZE(14)), -- Originating Node's Peer Group ID reserved OCTET STRING (SIZE(2)), -- ptse_sc OCTET STRING (SIZE(2)), -- PTSE Summary Count ptse_seq SEQUENCE OF NodalPTSESummariesIG_SUB_T -- Number of PTSE summaries for this originating node ID. -- PTSE header information of -- all PTSEs in a node's topology database }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Definition | |
|---|-----------------------------|
| Type Name | : NodalStateParametersIG_T |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 96 (Nodal State Parameters) length OCTET STRING (SIZE(2)), -- Length flags VP_Capabilities_Flag_T, -- VP Capability Flag reserved OCTET STRING (SIZE(2)), -- in_port OCTET STRING (SIZE(4)), -- Input Port ID out_port OCTET STRING (SIZE(4)), -- Input Port ID out_raig SET OF ResourceAvailabilityIG_T -- Outgoing Resource Availability IG }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Definition | |
|---|-----------------------|
| Type Name | : Nodal_Flags_T |
| Encoding Variation | : |
| Comments | : Nodal Flags |
| Type Definition | |
| <pre>SEQUENCE { bit_8_leader BIT STRING (SIZE(1)), -- 'I am Leader' bit bit_7_restr_trans BIT STRING (SIZE(1)), -- Restricted Transit bit bit_6_nodal_repr BIT STRING (SIZE(1)), -- Nodal Representation bit bit_5_restr_branch BIT STRING (SIZE(1)), -- Restricted Branching bit bit_4_non_trans BIT STRING (SIZE(1)), -- Non-Transit for PGL Election bit bit_3_l_res BIT STRING (SIZE(3)) -- Reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Definition | |
|--|---|
| Type Name | : OptionalGCACparametersIG_T |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 160 (GCAC) length OCTET STRING (SIZE(2)), -- Length crm OCTET STRING (SIZE(4)), -- Cell Rate Margin variance OCTET STRING (SIZE(4)) -- Variance Factor }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Definition | |
|---|-------------------------------|
| Type Name | : PTSEIG_T |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 64 (PTSE) length OCTET STRING (SIZE(2)), -- Length ptse_type OCTET STRING (SIZE(2)), -- Indicates which restricted IGs are allowed to -- appear inside of the PTSE reserved OCTET STRING (SIZE(2)), -- reserved ptse_id OCTET STRING (SIZE(4)), -- Identifies one of multiple different PTSEs -- from a node ptse_seq_no OCTET STRING (SIZE(4)), -- PTSE Sequence Number ptse_crc OCTET STRING (SIZE(2)), -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl OCTET STRING (SIZE(2)), -- PTSE remaining lifetime ig PTSE_IGs_T OPTIONAL -- PTSE Information Groups }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Definition | |
|---|---------------------------|
| Type Name | : PTSE_IGs_T |
| Encoding Variation | : |
| Comments | : PTSE Information Groups |
| Type Definition | |
| <pre> SET { nodal_ig SET OF NodalIG_T OPTIONAL, -- Nodal Information Group nodal_state_par SET OF NodalStateParametersIG_T OPTIONAL, -- Nodal State Parameter int_reach_addr SET OF InternalReachableATMAddressesIG_T OPTIONAL, -- Internal Reachable ATM Addresses ext_reach_addr SET OF ExteriorReachableATMAddressesIG_T OPTIONAL, -- Exterior Reachable ATM Addresses horizontal_links SET OF HorizontalLinksIG_T OPTIONAL, -- Horizontal Links uplinks SET OF UplinksIG_T OPTIONAL, -- Uplinks sys_cap SET OF SystemCapabilitiesIG_T OPTIONAL -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Definition | |
|--|----------------------|
| Type Name | : PnniHeader_T |
| Encoding Variation | : |
| Comments | : PNNI Packet Header |
| Type Definition | |
| <pre> SEQUENCE { type OCTET STRING (SIZE(2)), -- Packet Type length OCTET STRING (SIZE(2)), -- Packet Length version OCTET STRING (SIZE(1)), -- Protocol Version n_version OCTET STRING (SIZE(1)), -- Newest Version Supported o_version OCTET STRING (SIZE(1)), -- Oldest Version Supported reserved OCTET STRING (SIZE(1)) -- Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.4 |

| ASN.1 Type Definition | |
|---|-------------------|
| Type Name | : RAIG_Flags_T |
| Encoding Variation | : |
| Comments | : RAIG Flags |
| Type Definition | |
| <pre> SEQUENCE { bit_16_cbr BIT STRING (SIZE(1)), -- CBR bit_15_rt_vbr BIT STRING (SIZE(1)), -- rt-VBR bit_14_nrt_vbr BIT STRING (SIZE(1)), -- nrt-VBR bit_13_abr BIT STRING (SIZE(1)), -- ABR bit_12_ubr BIT STRING (SIZE(1)), -- UBR bit_11_2_res BIT STRING (SIZE(10)), -- Reserved bit_1_gcac_clp_attr BIT STRING (SIZE(1)) -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Definition | |
|---|---|
| Type Name | : RAIG_SUB_T |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Type Definition | |
| <pre> SET { out_raig ResourceAvailabilityIG_T OPTIONAL, -- Outgoing resource availability information group in_raig ResourceAvailabilityIG_T OPTIONAL -- Incoming resource availability information group } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Definition | |
|---|----------------------------|
| Type Name | : ResourceAvailabilityIG_T |
| Encoding Variation | : |
| Comments | : Resource Availability IG |
| Type Definition | |
| <pre> SEQUENCE { type IG_Type_T, -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length OCTET STRING (SIZE(2)), -- Length flags RAIG_Flags_T, -- RAIG Flags reserved OCTET STRING (SIZE(2)), -- Reserved weight OCTET STRING (SIZE(4)), -- Administrative Weight (default Value = 5040) mcr OCTET STRING (SIZE(4)), -- Maximum Cell Rate acr OCTET STRING (SIZE(4)), -- Available Cell Rate ctd OCTET STRING (SIZE(4)), -- Cell Transfer Delay cdv OCTET STRING (SIZE(4)), -- Cell Delay Variation clr_0 OCTET STRING (SIZE(2)), -- Cell Loss Ratio (CLP=0) clr_01 OCTET STRING (SIZE(2)), -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparametersIG_T OPTIONAL -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : NNI 1.0 5.14.5 |

| ASN.1 Type Definition | |
|---|--------------------------|
| Type Name | : SystemCapabilitiesIG_T |
| Encoding Variation | : |
| Comments | : System Capabilities IG |
| Type Definition | |
| <pre> SEQUENCE { type IG_Type_T, -- Type = 640 length OCTET STRING (SIZE(2)), -- Length sc_length OCTET STRING (SIZE(2)), -- Length of system capabilities contents ieee_oui OCTET STRING (SIZE(3)), -- IEEE OUI sc_info OCTET STRING, -- System capabilities information padding OCTET STRING (SIZE(0..3)) -- Padding } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.13 |

| ASN.1 Type Definition | |
|--|-------------------------|
| Type Name | : TransitNetworkIdIG_T |
| Encoding Variation | : |
| Comments | : Transit Network ID IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 304 length OCTET STRING (SIZE(2)), -- Length tns_length OCTET STRING (SIZE(2)), -- Length of TNS n_id_data NetworkIdentificationData_T, -- Network identification data n_id OCTET STRING, -- Network identification padding OCTET STRING (SIZE(0..3)) -- Padding } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.7 |

| ASN.1 Type Definition | |
|--|-----------------------------------|
| Type Name | : UplinkInformationAttributeIG_T |
| Encoding Variation | : |
| Comments | : Uplink Information Attribute IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 34 (ULIA) length OCTET STRING (SIZE(2)), -- Length seq_num OCTET STRING (SIZE(4)), -- Sequence Number out_ra SEQUENCE OF ResourceAvailablilityIG_T OPTIONAL -- Outgoing Resource Availability } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.6 |

| ASN.1 Type Definition | |
|---|-----------------------|
| Type Name | : UplinksIG_T |
| Encoding Variation | : |
| Comments | : Uplinks IG |
| Type Definition | |
| <pre>SEQUENCE { type IG_Type_T, -- Type = 289 length OCTET STRING (SIZE(2)), -- Length flags VP_Capabilities_Flag_T, -- VP Capability Flag reserved OCTET STRING (SIZE(2)), -- Reserved r_hl_node OCTET STRING (SIZE(22)), -- Remote Higher Level Node ID comm_pg OCTET STRING (SIZE(14)), -- Common Peer Group ID l_port OCTET STRING (SIZE(4)), -- Local Port ID token OCTET STRING (SIZE(4)), -- Aggregation Token up_addr OCTET STRING (SIZE(20)), -- ATM End System Address of Upnode out_raig SEQUENCE OF ResourceAvailablilityIG_T, -- Outgoing Resource Availability ulia UplinkInformationAttributeIG_T -- Uplink Information Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.6 |

| ASN.1 Type Definition | |
|---|--------------------------|
| Type Name | : VP_Capabilities_Flag_T |
| Encoding Variation | : |
| Comments | : VP Capability Flag |
| Type Definition | |
| <pre>SEQUENCE { bit_16_vp_capability BIT STRING (SIZE(1)), -- VP Capability Flag bit_15_1_res BIT STRING (SIZE(15)) -- reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| Test Suite Operation Definition | |
|--|--|
| Operation Name : | INT_TO_OCT(value, numberOfOctets: INTEGER) |
| Result Type : | OCTETSTRING |
| Comments : | |
| Description | |
| Encodes value into the specified numberOfOctets. All OCTETs in OCTETSTRING are set to 0xFF if the indicated value causes an overflow. Value shall always be a non-negative number. | |
| E.g. INT_TO_OCT(15, 1) returns '0F'0 INT_TO_OCT(-1, 2) returns 'FFFF'0 INT_TO_OCT(256,1) returns 'FF'0 | |
| Detailed Comments : | |

| Test Suite Operation Definition | |
|---|-------------------------------------|
| Operation Name : | OCT_TO_INT (octetField:OCTETSTRING) |
| Result Type : | INTEGER |
| Comments : | |
| Description | |
| This Test Suite Operation is used to convert an OCTETSTRING field into an integer value. All return values are assumed to be > 0. | |
| Example: OCT_TO_INT('FF'0) returns 255 OCT_TO_INT('00'0) returns 0 | |
| Detailed Comments : | |

| Test Suite Operation Definition | |
|--|-----------------------------------|
| Operation Name : | Version(TNV, TOV, NV, OV:INTEGER) |
| Result Type : | INTEGER |
| Comments : | |
| Description | |
| Returns the lowest of the highest locally supportable version and the highest version supportable by the neighbor. | |
| Detailed Comments : | |

| Test Suite Operation Definition | |
|--|-----------|
| Operation Name : | DSno_init |
| Result Type : | INTEGER |
| Comments : | |
| Description | |
| When the Negotiating state is first entered, the DS sequence number should be set to a value not previously seen by the neighboring peer but not too large to safely avoid sequence number wrapping. One possible scheme is to use the lower 24 bits of the machine's time of day counter. | |
| Detailed Comments : | |

| Test Suite Operation Definition | |
|---|---------------|
| Operation Name | : HelloLinkUp |
| Result Type | : BOOLEAN |
| Comments | : |
| Description | |
| Generate a Hello LinkUp event on the IUT. | |
| Detailed Comments : | |

| Test Suite Operation Definition | |
|---|---------------------------|
| Operation Name | : PTSE_crc(PTSP : PTSP_T) |
| Result Type | : OCTETSTRING |
| Comments | : PTSE check sum |
| Description | |
| /* The checksum field is the 16 bit one's complement of the one's complement sum of all 16 bit words in the PTSE except the Lifetime. In addition, the Originating Node ID and Originating Node's Peer Group ID fields from the PTSP header are included. For purposes of computing the checksum, the value of the checksum field is zero. */ | |
| Detailed Comments : PNNI 1.0 5.8.2.2.2 | |

| Test Suite Parameter Declarations | | | |
|-----------------------------------|-------------|----------------|---|
| Parameter Name | Type | PICS/PIXIT Ref | Comments |
| tsp_SS_M_SUPP | BOOLEAN | PICS SS_M | Does the IUT support the minimum function switching system subset? |
| tsp_SS_B_SUPP | BOOLEAN | PICS SS_B | Does the IUT support the border node capable switching system subset? |
| tsp_NID | OCTETSTRING | PIXIT A.1 | Node ID of the IUT. |
| tsp_AESA | OCTETSTRING | PIXIT A.2 | ATM End System Address of the IUT. |
| tsp_PGID | OCTETSTRING | PIXIT A.3 | Peer Group ID of the IUT. |
| tsp_NV | INTEGER | PIXIT SP.1 | Newest version supported by the IUT. |
| tsp_OV | INTEGER | PIXIT SP.2 | Oldest version supported by the IUT. |
| tsp_LGN | OCTETSTRING | PIXIT A.4 | Next higher level LGN ID of the IUT. |
| tsp_LPG | OCTETSTRING | PIXIT A.5 | Next higher level LGN Peer Group ID of the IUT. |
| tsp_TLNID | OCTETSTRING | PIXIT A.6 | Node ID of the Tester, smaller than the IUT's node ID but in the same PNNI level. |
| tsp_THNID | OCTETSTRING | PIXIT A.7 | Node ID of the Tester, higher than the IUT's node ID but in the same PNNI level. |
| tsp_TAESA | OCTETSTRING | PIXIT A.8 | ATM End System Address of the Tester. |
| tsp_TPGID | OCTETSTRING | PIXIT A.9 | Peer Group ID of the Tester different from the IUTs Peer Group ID. (used for outside messages) |
| tsp_TPID | OCTETSTRING | PIXIT A.10 | Port ID of the Tester. |
| tsp_TNV | INTEGER | PIXIT SP.3 | Newest version supported by the Tester. |
| tsp_TOV | INTEGER | PIXIT SP.4 | Oldest version supported by the Tester. |
| tsp_TLGN | OCTETSTRING | PIXIT A.11 | Next higher level LGN ID of the Tester. |
| tsp_TLPG | OCTETSTRING | PIXIT A.12 | Next higher level LGN Peer Group ID of the Tester. |
| tsp_AGT | OCTETSTRING | PIXIT SP.5 | Default value for the configured Aggregation Token. Default '00000000'0 |
| tsp_Resp_Time | INTEGER | PIXIT T.1 | Maximum time in seconds allowed for a response by the IUT. Must be longer than the Hold-down timer plus a 25% jitter. (default 2 sec) |

Continued on next page

Continued from previous page

| Test Suite Parameter Declarations | | | |
|-----------------------------------|---------|----------------|--|
| Parameter Name | Type | PICS/PIXIT Ref | Comments |
| tsp_NoResp_Time | INTEGER | PIXIT T.2 | Maximum time in seconds where no response is allowed by the IUT. Must be longer than the Hold-down timer plus a 25% jitter and shorter than the Hello Interval time minus a 25% jitter. (default 3 sec) |
| tsp_THI | INTEGER | PIXIT T.3 | Hello Interval time of the tester (default 15000 msec). |
| tsp_IF | INTEGER | PIXIT SP.6 | The number of HelloIntervals allowed to pass without receiving a Hello, before the Hello FSM declares that a link is down. (Inactivity factor) (default 5) |
| tsp_Hold_Time | INTEGER | PIXIT T.4 | Minimum interval between successive Hello transmissions (Hold-down timer) (default 1000 msec, minimum 100 msec). |
| tsp_PTSERetr | INTEGER | PIXIT T.5 | The interval at which unacknowledged PTSEs will be retransmitted. (PTSERetransmissionInterval) (default 5000 ms). |
| tsp_PTSERefr | INTEGER | PIXIT T.6 | This is the time in seconds between reoriginations of a self-originated PTSE in the absence of triggered updates. (PTSERefreshInterval) (default 1800s) |
| tsp_PeerDelayedAck | INTEGER | PIXIT T.7 | The minimum number of milliseconds between transmissions of delayed PTSE acknowledgment packets. (PeerDelayedAck Timer) (default 1000 msec) |
| tsp_DSRxmt | INTEGER | PIXIT T.8 | The amount of time, in milliseconds, a node waits before it sends the previous Database Summary packet again. (DSRxmtInterval) (default 5000 msec) |
| tsp_ReqRxmt | INTEGER | PIXIT T.9 | The amount of time, in milliseconds, before a node sends a new PTSE Request Packet requesting PTSEs of the last PTSE Request Packet that have not been received yet. (RequestRxmtInterval) (default 5000 msec) |

Continued on next page

Continued from previous page

| Test Suite Parameter Declarations | | | |
|-----------------------------------|---------|----------------|--|
| Parameter Name | Type | PICS/PIXIT Ref | Comments |
| tsp_GCAC_T_SUPP | BOOLEAN | PICS 3.15.10 | Does the IUT support the transmission of the optional field for GCAC (Generic Call Admission Control) related information in the Resource Availability IG? |
| tsp_IRA_RA_T_SUPP | BOOLEAN | PICS 3.15.14 | Does the IUT support the transmission of the optional resource availability information within the Internal Reachable ATM Address IG? |
| tsp_ERA_RA_T_SUPP | BOOLEAN | PICS 3.15.17 | Does the IUT support the transmission of the optional resource availability information within the Exterior Reachable ATM Address IG? |
| tsp_ERA_TN_T_SUPP | BOOLEAN | PICS 3.15.17 | Does the IUT support the transmission of the optional Transit Network ID within the Exterior Reachable ATM Address IG? |
| tsp_MinPTSEInterval | INTEGER | PIXIT T.10 | Minimal time interval between PTSE updates. That is, a node is limited to updating any particular PTSE no more than once every MinPTSEInterval seconds. (5.8.3.7) (default 1s) |
| Detailed Comments : | | | |

| Test Case Selection Expression Definitions | | |
|--|---|--|
| Expression Name | Selection Expression | Comments |
| SELECT_SS_M | tsp_SS_M_SUPP | Does the IUT support the minimum function switching system subset? |
| SELECT_SS_M_SS_B | tsp_SS_M_SUPP AND tsp_SS_B_SUPP | Does the IUT support the minimum function switching system subset and does the IUT support the border node capable switching system subset? |
| SELECT_SS_M_NOT_SS_B | tsp_SS_M_SUPP AND NOT(tsp_SS_B_SUPP) | Does the IUT support the minimum function switching system subset and does the IUT not support the border node capable switching system subset? |
| SELECT_GCAC_T | tsp_GCAC_T_SUPP | Does the IUT support the transmission of the optional field for GCAC (Generic Call Admission Control) related information in the Resource Availability IG? |
| SELECT_IRA_RA_T | tsp_IRA_RA_T_SUPP | Does the IUT support the transmission of the optional resource availability information within the Internal Reachable ATM Address IG? |
| SELECT_ERA_RA_T | tsp_ERA_RA_T_SUPP | Does the IUT support the transmission of the optional resource availability information within the Exterior Reachable ATM Address IG? |
| SELECT_ERA_TN_T | tsp_ERA_TN_T_SUPP | Does the IUT support the transmission of the optional Transit Network ID within the Exterior Reachable ATM Address IG? |
| Detailed Comments : | | |

| Test Suite Constant Declarations | | | |
|----------------------------------|-------------|---------------------|---|
| Constant Name | Type | Value | Comments |
| PT_HELLO | OCTETSTRING | '0001'O | Hello Packet Type |
| PT_PTSP | OCTETSTRING | '0002'O | PTSP Packet Type |
| PT_PTSE_ACK | OCTETSTRING | '0003'O | PTSE Acknowledgement Packet Type |
| PT_DB_SUMM | OCTETSTRING | '0004'O | Database Summary Packet Type |
| PT_PTSE_REQ | OCTETSTRING | '0005'O | PTSE Request Packet Type |
| IGT_AT | BITSTRING | '000000100000'B | Aggregation Token IG Type (32) |
| IGT_NHL | BITSTRING | '000000100001'B | Nodal Hierarchy List IG Type (33) |
| IGT_ULIA | BITSTRING | '000000100010'B | Uplink Information Attribute IG Type (34) |
| IGT_LGN_NHL | BITSTRING | '000000100011'B | LGN Horizontal Link Extension IG Type (35) |
| IGT_PTSE | BITSTRING | '000001000000'B | PTSE IG Type (64) |
| IGT_NSP | BITSTRING | '000001100000'B | Nodal State Parameters IG Type (96) |
| IGT_NODAL | BITSTRING | '000001100001'B | Nodal IG Type (97) |
| IGT_ORA | BITSTRING | '000010000000'B | Outgoing Resource Availability IG Type (128) |
| IGT_INRA | BITSTRING | '000010000001'B | Incoming Resource Availability IG Type (129) |
| IGT_GCAC | BITSTRING | '000010100000'B | Optional GCAC Parameters IG Type (160) |
| IGT_NHLB | BITSTRING | '000011000000'B | Next Higher Level Binding Information IG Type (192) |
| IGT_IRA | BITSTRING | '000011100000'B | Internal Reachable ATM Address IG Type (224) |
| IGT_ERA | BITSTRING | '000100000000'B | External Reachable ATM Address IG Type (256) |
| IGT_HL | BITSTRING | '000100100000'B | Horizontal Links IG Type (288) |
| IGT_UL | BITSTRING | '000100100001'B | Uplinks IG Type (289) |
| IGT_TNI | BITSTRING | '000100110000'B | Transit Network ID IG Type (304) |
| IGT_NPA | BITSTRING | '000110000000'B | Nodal PTSE Ack IG Type (384) |
| IGT_NPS | BITSTRING | '001000000000'B | Nodal PTSE Summaries IG Type (512) |
| IGT_RPH | BITSTRING | '001000000001'B | Requested PTSE Header IG Type (513) |
| IGT_SC | BITSTRING | '001010000000'B | System Capabilities IG Type (640) |
| SEQ_NUM | INTEGER | 1 | Sequence number used in IGs |
| PTSET_O | BITSTRING | '0000000000000001'B | PTSETType other(1) |
| PTSET_NSP | BITSTRING | '0000000001100000'B | PTSETType nodalStateParameters(96) |
| PTSET_NI | BITSTRING | '0000000001100001'B | PTSETType nodalInformation(97) |
| PTSET_IRA | BITSTRING | '0000000001110000'B | PTSETType internalReachableAddresses(224) |
| PTSET_ERA | BITSTRING | '0000000100000000'B | PTSETType exteriorReachableAddresses(256) |

Continued on next page

Continued from previous page

| Test Suite Constant Declarations | | | |
|----------------------------------|-------------|---------------------|--|
| Constant Name | Type | Value | Comments |
| PTSET_HL | BITSTRING | '0000000100100000'B | PTSETType horizontalLinks(288) |
| PTSET_UL | BITSTRING | '0000000100100001'B | PTSETType uplinks(289) |
| CBR_AW | OCTETSTRING | '000013B0'O | Administrative Weight of CBR/UBR announced by the tester. |
| CBR_MCR | OCTETSTRING | '000598C8'O | Maximum Cell Rate of CBR/UBR announced by the tester. |
| CBR_ACR | OCTETSTRING | '00055124'O | Available of CBR/UBR announced by the tester. |
| CBR_CTD | OCTETSTRING | '0000009A'O | Cell Transfer Delay of CBR/UBR announced by the tester. |
| CBR_CDV | OCTETSTRING | '0000008A'O | Cell Delay Variation of CBR/UBR announced by the tester. |
| CBR_CLR0 | OCTETSTRING | '000A'O | Cell Loss Ratio (CLP=0) of CBR/UBR announced by the tester. |
| CBR_CLR10 | OCTETSTRING | '000A'O | Cell Loss Ratio (CLP=0+1) of CBR/UBR announced by the tester. |
| RTVBR_AW | OCTETSTRING | '000013B0'O | Administrative Weight of Real Time VBR announced by the tester. |
| RTVBR_MCR | OCTETSTRING | '000598C8'O | Maximum Cell Rate of Real Time VBR announced by the tester. |
| RTVBR_ACR | OCTETSTRING | '000598C8'O | Available of Real Time VBR announced by the tester. |
| RTVBR_CTD | OCTETSTRING | '000002C3'O | Cell Transfer Delay of Real Time VBR announced by the tester. |
| RTVBR_CDV | OCTETSTRING | '000002B3'O | Cell Delay Variation of Real Time VBR announced by the tester. |
| RTVBR_CLR0 | OCTETSTRING | '0008'O | Cell Loss Ratio (CLP=0) of Real Time VBR announced by the tester. |
| RTVBR_CLR10 | OCTETSTRING | '0008'O | Cell Loss Ratio (CLP=0+1) of Real Time VBR announced by the tester. |
| NRTVBR_AW | OCTETSTRING | '000013B0'O | Administrative Weight of Non-Real Time VBR announced by the tester. |
| NRTVBR_MCR | OCTETSTRING | '000598C8'O | Maximum Cell Rate of Non-Real Time VBR announced by the tester. |
| NRTVBR_ACR | OCTETSTRING | '000598C8'O | Available of Non-Real Time VBR announced by the tester. |

Continued on next page

Continued from previous page

| Test Suite Constant Declarations | | | |
|----------------------------------|-------------|---|---|
| Constant Name | Type | Value | Comments |
| NRTVBR_CTD | OCTETSTRING | '000002C3'0 | Cell Transfer Delay of Non-Real Time VBR announced by the tester. |
| NRTVBR_CDV | OCTETSTRING | '00000000'0 | Cell Delay Variation of Non-Real Time VBR announced by the tester. |
| NRTVBR_CLR0 | OCTETSTRING | '0008'0 | Cell Loss Ratio (CLP=0) of Non-Real Time VBR announced by the tester. |
| NRTVBR_CLR10 | OCTETSTRING | '0008'0 | Cell Loss Ratio (CLP=0+1) of Non-Real Time VBR announced by the tester. |
| ABR_AW | OCTETSTRING | '000013B0'0 | Administrative Weight of ABR announced by the tester. |
| ABR_MCR | OCTETSTRING | '000598C8'0 | Maximum Cell Rate of ABR announced by the tester. |
| ABR_ACR | OCTETSTRING | '00000000'0 | Available of ABR announced by the tester. |
| ABR_CTD | OCTETSTRING | 'FFE17B7F'0 | Cell Transfer Delay of ABR announced by the tester. |
| ABR_CDV | OCTETSTRING | 'FFE17B7F'0 | Cell Delay Variation of ABR announced by the tester. |
| ABR_CLR0 | OCTETSTRING | 'FFFF'0 | Cell Loss Ratio (CLP=0) of ABR announced by the tester. |
| ABR_CLR10 | OCTETSTRING | 'FFFF'0 | Cell Loss Ratio (CLP=0+1) of ABR announced by the tester. |
| RTVBR_CRM | OCTETSTRING | '00000008'0 | Cell Rate Margin (CRM) of Real Time VBR announced by the tester. |
| RTVBR_VF | OCTETSTRING | '00000002'0 | Variance Factor (VF) of Real Time VBR announced by the tester. |
| NRTVBR_CRM | OCTETSTRING | '00000008'0 | Cell Rate Margin (CRM) of Non-Real Time VBR announced by the tester. |
| NRTVBR_VF | OCTETSTRING | '00000002'0 | Variance Factor (VF) of Non-Real Time VBR announced by the tester. |
| EXPIRED_AGE | INTEGER | 0 | PTSE ExpiredAge. |
| TAESA_2ND | OCTETSTRING | '39276F310001EF00000401700FFF1A3690001100'0 | ATM End System Address of the Tester that emulates the second node. |
| Detailed Comments : | | | |

| Test Suite Variable Declarations | | | |
|----------------------------------|-------------|-------------|--|
| Variable Name | Type | Value | Comments |
| tsv_R_PID | OCTETSTRING | '00000000'0 | Port ID of the IUT |
| tsv_R_HI | OCTETSTRING | | HelloInterval value of the IUT read from the most recent received Hello |
| tsv_DSno | INTEGER | | DS Sequence Number for the Negotiating State of the Neighbouring Peer State Machine. |
| Detailed Comments : | | | |

| Test Case Variable Declarations | | | |
|---------------------------------|-------------|---|---|
| Variable Name | Type | Value | Comments |
| tcv_LINK_UP | BOOLEAN | | Variable to store the state of the link. |
| tcv_TNID | OCTETSTRING | | Variable to store the current node ID of the tester. |
| tcv_CV | INTEGER | 1 | Variable to store the current version supported by the IUT. |
| tcv_TIME | INTEGER | 0 | Variable to store the time that has passed since a specified timer was started. |
| tcv_NHL_NO | INTEGER | 0 | Variable to store the Nodal Hierarchy List Sequence Number. |
| tcv_ULIA_NO | INTEGER | 0 | Variable to store the Uplink Information Attribute Sequence Number. |
| tcv_PTSE_TYPE | OCTETSTRING | | Variable to store a received PTSE type |
| tcv_PTSE_ID | OCTETSTRING | | Variable to store a received PTSE identifier |
| tcv_PTSE_SEQ | OCTETSTRING | | Variable to store a received PTSE sequence number |
| tcv_CRC | OCTETSTRING | '0000'0 | Variable to store the current PTSE checksum |
| tcv_PTSE_RLT | OCTETSTRING | | Variable to store a received PTSE remaining life time |
| tcv_CRC_r | OCTETSTRING | '0000'0 | Variable to store the received PTSE checksum |
| tcv_TAESA | OCTETSTRING | '00'0 | ATM End System Address of the tester. |
| Detailed Comments : | | | |

| PCO Type Declarations | | |
|----------------------------|------|----------|
| PCO Type | Role | Comments |
| R_SAP | LT | |
| Detailed Comments : | | |

| PCO Declarations | | | |
|----------------------------|----------|------|--|
| PCO Name | PCO Type | Role | Comments |
| Hello_PCO_1 | R_SAP | LT | Routing Service Access Point for the Hello Protocol |
| Neighb_Peer_PCO_1 | R_SAP | LT | Routing Service Access Point for the Database Synchronization Protocol |
| Hello_PCO_2 | R_SAP | LT | Routing Service Access Point for the Hello Protocol |
| Neighb_Peer_PCO_2 | R_SAP | LT | Routing Service Access Point for the Database Synchronization Protocol |
| Detailed Comments : | | | |

| Coordination Point Declarations | |
|--|---|
| CP Name | Comments |
| Hello_CP_1 | Coordination Point for the Hello Protocol |
| Neighb_Peer_CP_1 | Coordination Point for the Database Synchronization Protocol |
| Hello_Neighb_Peer_CP_1 | Coordination Point between the Hello Protocol and the Database Synchronization Protocol |
| Hello_CP_2 | Coordination Point for the Hello Protocol |
| Neighb_Peer_CP_2 | Coordination Point for the Database Synchronization Protocol |
| Hello_Neighb_Peer_CP_2 | Coordination Point between the Hello Protocol and the Database Synchronization Protocol |
| Detailed Comments : | |

| Timer Declarations | | | |
|--------------------|---|------|---|
| Timer Name | Duration | Unit | Comments |
| T_Resp | tsp_Resp_Time | s | Maximum time in seconds allowed for a response by the IUT. Must be longer than the Hold-down timer plus a 25% jitter. (default 2 sec) |
| T_NoResp | tsp_NoResp_Time | s | Maximum time in seconds where no response is allowed by the IUT. Must be longer than the Hold-down timer plus a 25% jitter and shorter than the Hello Interval time minus a 25% jitter. (default 3 sec) |
| T_Inact | $tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4$ | ms | Timer for checking the inactivity timeout. |
| T_Hello | $tsp_THI + tsp_THI/4$ | ms | Timer for checking the intervals in which Hellos are received. |
| T_Period | $tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4$ | ms | Interval timer for testing properties that should be true. (like, for all Hellos received in a certain state) |
| T_Hold | $tsp_Hold_Time - tsp_Hold_Time/4$ | ms | Interval timer, with period MinHelloInterval. Used for checking the minimum interval between successive Hello transmissions. (Hold-down timer) |
| T_PTSERetr | tsp_PTSERetr | ms | Interval timer: each unacknowledged PTSE is retransmitted every T_PTSERetr seconds (PTSERetransmissionInterval) (default 5000 ms) |
| T_PTSERefr | tsp_PTSERefr | s | Interval timer: time between reoriginations of a self-originated PTSE in the absence of triggered updates. (PTSERefreshInterval) (default 1800s) |
| T_PeerDelayedAck | $tsp_PeerDelayedAck + tsp_PeerDelayedAck/4$ | ms | Interval timer: The minimum number of milli-seconds between transmissions of delayed PTSE acknowledgment packets. (PeerDelayedAck Timer) (default 1000 msec) |
| T_DSRxmt | $tsp_DSRxmt + tsp_DSRxmt/4$ | ms | An interval timer that fires after DSRxmtInterval seconds. (DS Rxmt Timer) (default 5000 msec) |
| T_ReqRxmt | $tsp_ReqRxmt + tsp_ReqRxmt/4$ | ms | An interval timer that fires after RequestRxmtInterval seconds. (Request Rxmt Timer) (default 5000 msec) |

Continued on next page

Continued from previous page

| Timer Declarations | | | |
|---|----------|------|----------|
| Timer Name | Duration | Unit | Comments |
| T_FldgL | 120 | s | |
| T_FldgS | 3 | s | |
| Detailed Comments : Note: PNNI 1.0 5.1.1. All timers that trigger transmission of messages are jittered. Therefore, a fractional variance of 25% of the nominal value is added to this timers. | | | |

| Test Component Declarations | | | | |
|-----------------------------|----------------|---------|--------|---|
| Component Name | Component Role | Nr PCOs | Nr CPs | Comments |
| Main | MTC | 0 | 4 | Main Test Component |
| Hello_FSM | PTC | 1 | 1 | Parallel Test Component to test the IUT Hello-FSM |
| Hello_FSM_1 | PTC | 1 | 2 | Parallel Test Component to test the IUT Hello-FSM |
| Hello_FSM_1_2CPs | PTC | 1 | 2 | Parallel Test Component to test the IUT Hello-FSM |
| Neighb_Peer_FSM_1 | PTC | 1 | 2 | Parallel Test Component to test the IUT Neighbouring Peer FSM |
| Hello_FSM_2 | PTC | 1 | 2 | Parallel Test Component to test the IUT Hello-FSM |
| Neighb_Peer_FSM_2 | PTC | 1 | 2 | Parallel Test Component to test the IUT Neighbouring Peer FSM |
| Detailed Comments : | | | | |

| Test Components Configuration Declaration | | | |
|---|-------------|--------------------------|--|
| Configuration Name : tcc>Hello | | | |
| Comments : Test Components Configuration to test the IUT Hello-FSM | | | |
| Components Used | PCOs Used | CPs Used | Comments |
| Main Hello_FSM | Hello_PCO_1 | Hello_CP_1 Hello_CP_1 | Main Test Component Parallel Test Component to test the IUT Hello-FSM |
| Detailed Comments : | | | |

| Test Components Configuration Declaration | | | |
|--|--------------------------------------|---|--|
| Configuration Name : tcc>DBSync | | | |
| Comments : Test Components Configuration to test the IUT Database Synchronization FSM | | | |
| Components Used | PCOs Used | CPs Used | Comments |
| Main Hello_FSM_1 Neighb_Peer_FSM_1 | Hello_PCO_1 Neighb_Peer_PCO_1 | Hello_CP_1, Neighb_Peer_CP_1 Hello_CP_1, Hello_Neighb_Peer_CP_1 Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1 | Main Test Component Parallel Test Component to emulate the IUT Hello FSM Parallel Test Component to test the IUT Neighbouring Peer FSM |
| Detailed Comments : | | | |

| Test Components Configuration Declaration | | | |
|---|--|---|---|
| Configuration Name : tcc>Flooding | | | |
| Comments : Test Components Configuration to test the IUT wrt Flooding. | | | |
| Components Used | PCOs Used | CPs Used | Comments |
| Main Hello_FSM_1 Neighb_Peer_FSM_1 Hello_FSM_2 Neighb_Peer_FSM_2 | Hello_PCO_1 Neighb_Peer_PCO_1 Hello_PCO_2 Neighb_Peer_PCO_2 | Hello_CP_1, Neighb_Peer_CP_1, Hello_CP_2, Neighb_Peer_CP_2 Hello_CP_1, Hello_Neighb_Peer_CP_1 Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1 Hello_CP_2, Hello_Neighb_Peer_CP_2 Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2 | Main Test Component Parallel Test Component to emulate the IUT Hello FSM Parallel Test Component to test the IUT Neighbouring Peer FSM Parallel Test Component to emulate the IUT Hello FSM Parallel Test Component to test the IUT Neighbouring Peer FSM |
| Detailed Comments : | | | |

| ASN.1 PDU Type Definition | |
|---|--------------------------------|
| PDU Name | : DBSP_T |
| PCO Type | : R_SAP |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Database Summary Packet |
| Type Definition | |
| <pre> SEQUENCE { header PnniHeader_T, -- PNNI Header (Type = 4) flags DBSum_Packet_Flags_T, -- Database Summary Packet Flags reserved OCTET STRING (SIZE(2)), -- Reserved ds_seq_no OCTET STRING (SIZE(4)), -- DS sequence number ptsp_seq SEQUENCE OF NodalPTSESummariesIG_T OPTIONAL -- PTSP and PTSE header information of -- all PTSEs in a node's topology database } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Type Definition | |
|--|---------------------|
| PDU Name | : Hello_T |
| PCO Type | : R_SAP |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Type Definition | |
| <pre> SEQUENCE { header PnniHeader_T, -- PNNI Header (Type = 1) flags OCTET STRING (SIZE(2)), -- Flags -Reserved- o_node OCTET STRING (SIZE(22)), -- Originating Node ID addr OCTET STRING (SIZE(20)), -- ATM End System Address o_pg OCTET STRING (SIZE(14)), -- Originating Peer Group ID r_node OCTET STRING (SIZE(22)), -- Remote Node ID o_port OCTET STRING (SIZE(4)), -- Port ID r_port OCTET STRING (SIZE(4)), -- Remote Port ID hello_int OCTET STRING (SIZE(2)), -- Hello Interval reserved OCTET STRING (SIZE(2)), -- Reserved ig Hello_IGs_T OPTIONAL -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 |

| ASN.1 PDU Type Definition | |
|---|--------------------------------|
| PDU Name | : PTSE_Ack_T |
| PCO Type | : R_SAP |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Acknowledgement Packets |
| Type Definition | |
| <pre>SEQUENCE { header PnniHeader_T, -- PNNI Header (Type = 3) ptse_req_list SET OF NodalPTSEAcknowledgementIG_T -- Nodal PTSE Acknowledgement } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 PDU Type Definition | |
|---|------------------------|
| PDU Name | : PTSE_Req_T |
| PCO Type | : R_SAP |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Request Packets |
| Type Definition | |
| <pre>SEQUENCE { header PnniHeader_T, -- PNNI Header (Type = 5) ptse_req_list SEQUENCE OF NodalPTSERequestListIG_T -- Nodal PTSE Request List } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.12 |

| ASN.1 PDU Type Definition | |
|--|------------------------------|
| PDU Name | : PTSP_T |
| PCO Type | : R_SAP |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet |
| Type Definition | |
| <pre>SEQUENCE { header PnniHeader_T, -- PNNI Header (Type = 2) o_node OCTET STRING (SIZE(22)), -- Originating Node ID o_pg OCTET STRING (SIZE(14)), -- Originating Peer Group ID ptse_seq SET OF PTSEIG_T -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 CM Type Definition |
|--|
| CM Name : AddPort_T Comments : A Hello state machine for a link to the neighboring peer has reached 2-WayInside state. Database Synchrononization is initiated. |
| Type Definition |
| <pre>SEQUENCE { }</pre> |
| Detailed Comments : |

| ASN.1 CM Type Definition |
|--|
| CM Name : DropPort_T Comments : When a link falls out of the Hello state 2-WayInside, the event DropPort is triggered in the corresponding neighboring peer state machine. When the DropPort event for the last link between the neighboring peers occurs, the neighboring peer state machine will internally generate the DropPortLast event causing all state information for the neighboring peer to be cleared. |
| Type Definition |
| <pre>SEQUENCE { }</pre> |
| Detailed Comments : |

| ASN.1 CM Type Definition |
|--|
| CM Name : TerminateReq_T Comments : MTC -> PTC Request by the MTC that the PTC indicated by the MCP terminate its execution. |
| Type Definition |
| <pre>SEQUENCE { cause Cause_T -- Cause for Termination Request }</pre> |
| Detailed Comments : |

| ASN.1 CM Type Definition |
|---|
| CM Name : TestBodyStartReq_T Comments : MTC -> PTC Request by the MTC that the PTC start with execution of the test body. |
| Type Definition |
| <pre>SEQUENCE { }</pre> |
| Detailed Comments : |

| ASN.1 CM Type Definition | |
|---|---|
| CM Name : NeighbFullInd_T | |
| Comments : PTC -> MTC PTC informs MTC that the Neighbouring Peer state Full is entered. | |
| Type Definition | |
| SEQUENCE | { |
| | } |
| Detailed Comments : | |

III

Constraints Part

| ASN.1 Type Constraint Declaration | |
|--|----------------------------|
| Constraint Name | : AggregationTokenIG_I_1_s |
| ASN1 Type | : AggregationTokenIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Aggregation Token IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_2_s('000000011111'B), -- Type = 31 (unknown) length INT_TO_OCT(8, 2), -- Length token tsp_AGT -- Aggregation Token }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.1 Aggregation Token IG to be sent by the tester with madatory bit tag fom invalid Aggregation Token TLV set to one and unknown IG type | |

| ASN.1 Type Constraint Declaration | |
|--|----------------------------|
| Constraint Name | : AggregationTokenIG_V_1_r |
| ASN1 Type | : AggregationTokenIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Aggregation Token IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_AT), -- Type = 32 (AT) length INT_TO_OCT(8, 2), -- Length token tsp_AGT -- Aggregation Token }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.1 Aggregation Token IG to be sent by IUT. | |

| ASN.1 Type Constraint Declaration | |
|--|----------------------------|
| Constraint Name | : AggregationTokenIG_V_1_s |
| ASN1 Type | : AggregationTokenIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Aggregation Token IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_AT), -- Type = 32 (AT) length INT_TO_OCT(8, 2), -- Length token tsp_AGT -- Aggregation Token }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.1 Aggregation Token IG to be sent by IUT. | |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : DBSum_Packet_Flags_V_1_r(init, more, master : BITSTRING) |
| ASN1 Type | : DBSum_Packet_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet Flags |
| Constraint Value | |
| <pre> { bit_16_init init, -- Bit 16 (MSB) : 'Initialize' (I) bit bit_15_more more, -- Bit 15: 'More'(M) bit: bit_14_master master, -- Bit 14: 'Master' (MS) bit bit_13_1_res '000000000000'B -- Bits 13..1 (LSB): Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : DBSum_Packet_Flags_V_1_s(init, more, master : BITSTRING) |
| ASN1 Type | : DBSum_Packet_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet Flags |
| Constraint Value | |
| <pre> { bit_16_init init, -- Bit 16 (MSB) : 'Initialize' (I) bit bit_15_more more, -- Bit 15: 'More'(M) bit: bit_14_master master, -- Bit 14: 'Master' (MS) bit bit_13_1_res '000000000000'B -- Bits 13..1 (LSB): Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_1_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ERA), -- Type = 256 length INT_TO_OCT(20,2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_1_r} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_1_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ERA), -- Type = 256 length INT_TO_OCT(20,2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_1_s} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_2_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ERA), -- Type = 256 length INT_TO_OCT(340,2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_2_r} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_2_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ERA), -- Type = 256 length INT_TO_OCT(340,2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_2_s} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_3_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ERA), -- Type = 256 length INT_TO_OCT(32,2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_3_r} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : ExtReachATMAddrIG_Fldg_V_3_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Exterior Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ERA), -- Type = 256 length INT_TO_OCT(32,2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {ExtReachATMAddrIG_SUB_Fldg_V_3_s} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_1_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre> { prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups -, -- Optional TLV groups for resource availability information add_tlv_gr - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_1_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre> { prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups -, -- Optional TLV groups for resource availability information add_tlv_gr - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_2_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups {RAIG_SUB_Fldg_V_1_r, RAIG_SUB_Fldg_V_2_r, RAIG_SUB_Fldg_V_3_r, RAIG_SUB_Fldg_V_4_r, RAIG_SUB_Fldg_V_5_r}, add_tlv_gr - }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_2_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups {RAIG_SUB_Fldg_V_1_s, RAIG_SUB_Fldg_V_2_s, RAIG_SUB_Fldg_V_3_s, RAIG_SUB_Fldg_V_4_s, RAIG_SUB_Fldg_V_5_s}, add_tlv_gr - -- Optional TLV groups for resource availability information }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_3_r |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups -, -- Optional TLV groups for resource availability information add_tlv_gr TransitNetIdIG_Fldg_V_1_r }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ExtReachATMAddrIG_SUB_Fldg_V_3_s |
| ASN1 Type | : ExteriorReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '47'O, -- Reachable Address Prefix padding '0000'O, -- Padding tlv_groups -, -- Optional TLV groups for resource availability information add_tlv_gr TransitNetIdIG_Fldg_V_1_s }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.4 |

| ASN.1 Type Constraint Declaration | |
|--|----------------------------|
| Constraint Name | : Hello_IGs_I_1_s |
| ASN1 Type | : Hello_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Hello Information Groups |
| Constraint Value | |
| <pre>{ aggr_token AggregationTokenIG_I_1_s, -- Aggregation Token nhl NodalHierarchyListIG_V_2_s, -- Nodal Hierarchy List ulia UplinkInformationAttributeIG_V_1_s, -- Uplink Information Attribute lgn_hl_ext -, -- LGN Horizontal Link Extension sys_cap -, -- System Capabilities }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 |
| <p>Hello Information Groups to be sent by the tester with madatory bit tag fom invalid Aggregation Token TLV set to one.</p> | |

| ASN.1 Type Constraint Declaration | |
|--|----------------------------|
| Constraint Name | : Hello_IGs_V_2_r |
| ASN1 Type | : Hello_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Hello Information Groups |
| Constraint Value | |
| <pre>{ aggr_token AggregationTokenIG_V_1_r, -- Aggregation Token nhl NodalHierarchyListIG_V_2_r, -- Nodal Hierarchy List ulia UplinkInformationAttributeIG_V_1_r, -- Uplink Information Attribute lgn_hl_ext -, -- LGN Horizontal Link Extension sys_cap *, -- System Capabilities }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 |
| <p>Hello Information Groups to be sent by the IUT with Hierarchy List (unknown).</p> | |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : Hello_IGs_V_2_s |
| ASN1 Type | : Hello_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Hello Information Groups |
| Constraint Value | |
| <pre>{ aggr_token AggregationTokenIG_V_1_s, -- Aggregation Token nhl NodalHierarchyListIG_V_3_s, -- Nodal Hierarchy List ulia UplinkInformationAttributeIG_V_1_s, -- Uplink Information Attribute lgn_hl_ext -, -- LGN Horizontal Link Extension sys_cap - -- System Capabilities }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Hello Information Groups (Common Hierarchy) to be sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : Hello_IGs_V_3_s |
| ASN1 Type | : Hello_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Hello Information Groups |
| Constraint Value | |
| <pre>{ aggr_token AggregationTokenIG_V_1_s, -- Aggregation Token nhl NodalHierarchyListIG_V_2_s, -- Nodal Hierarchy List ulia UplinkInformationAttributeIG_V_1_s, -- Uplink Information Attribute lgn_hl_ext -, -- LGN Horizontal Link Extension sys_cap - -- System Capabilities }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Hello Information Groups to be sent by the tester with empty Nodal Hierarchy List. |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------------------|
| Constraint Name | : HorizontalLinksIG_V_1_r |
| ASN1 Type | : HorizontalLinksIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Horizontal Links IG sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_HL), -- Type = 288 length ?, -- Length flags ?, -- VP Capability Flag r_node ?, -- Remote Node ID r_port ?, -- Remote Port ID l_port ?, -- Local Port ID token ?, -- Aggregation Token out_raig ? -- Outgoing Resource Availability } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.5 |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------|
| Constraint Name | : IG_TagBit_V_1_r |
| ASN1 Type | : IG_TagBit_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Tag Bit Definitions |
| Constraint Value | |
| <pre> { mand '0'B, -- Mandatory Bit d_sum '0'B, -- Don't Summarize Bit trans '0'B, -- Transitive Bit reserved '0'B -- Reserved Bit } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.2.6 |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------|
| Constraint Name | : IG_TagBit_V_1_s |
| ASN1 Type | : IG_TagBit_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Tag Bit Definitions |
| Constraint Value | |
| <pre> { mand '0'B, -- Mandatory Bit d_sum '0'B, -- Don't Summarize Bit trans '0'B, -- Transitive Bit reserved '0'B -- Reserved Bit } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.2.6 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : IG_TagBit_V_2_s |
| ASN1 Type | : IG_TagBit_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Tag Bit Definitions |
| Constraint Value | |
| <pre> { mand '1'B, -- Mandatory Bit d_sum '0'B, -- Don't Summarize Bit trans '0'B, -- Transitive Bit reserved '0'B -- Reserved Bit } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.2.6 IG Tag Bit Definitions with the mandatory tag bit set. |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : IG_Type_V_1_r(type_identifier:BITSTRING) |
| ASN1 Type | : IG_Type_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Type |
| Constraint Value | |
| <pre> { tag IG_TagBit_V_1_r, -- IG Tag Bit Definitions type type_identifier -- IG Type } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : IG_Type_V_1_s(type_identifier:BITSTRING) |
| ASN1 Type | : IG_Type_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Type |
| Constraint Value | |
| <pre> { tag IG_TagBit_V_1_s, -- IG Tag Bit Definitions type type_identifier -- IG Type } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : IG_Type_V_2_s(type_identifier:BITSTRING) |
| ASN1 Type | : IG_Type_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : IG Type |
| Constraint Value | |
| <pre>{ tag IG_TagBit_V_2_s, -- IG Tag Bit Definitions type type_identifier -- IG Type }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 IG Type with the mandatory tag bit set. |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------------------|
| Constraint Name | : IntReachATMAddrIG_Fldg_V_1_r |
| ASN1 Type | : InternalReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Internal Reachable ATM Addresses IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_IRA), -- Type = 224 length INT_TO_OCT(20,2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {IntReachATMAddrIG_SUB_V_1_r} -- Reachable Addresses }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------------------|
| Constraint Name | : IntReachATMAddrIG_Fldg_V_1_s |
| ASN1 Type | : InternalReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Internal Reachable ATM Addresses IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_IRA), -- Type = 224 length INT_TO_OCT(20,2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'0, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {IntReachATMAddrIG_SUB_V_1_s} -- Reachable Addresses }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------------------|
| Constraint Name | : IntReachATMAddrIG_Fldg_V_2_r |
| ASN1 Type | : InternalReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Internal Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_IRA), -- Type = 224 length INT_TO_OCT(340,2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved '?', -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {IntReachATMAddrIG_SUB_V_2_r} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------------------|
| Constraint Name | : IntReachATMAddrIG_Fldg_V_2_s |
| ASN1 Type | : InternalReachableATMAddressesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Internal Reachable ATM Addresses IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_IRA), -- Type = 224 length INT_TO_OCT(340,2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'0, -- Reserved port INT_TO_OCT(0,4), -- Port ID scope INT_TO_OCT(96,1), -- Scope of advertisement ail INT_TO_OCT(2,1), -- Address Information Length (in octets) aic INT_TO_OCT(1,2), -- Address Information Count addr {IntReachATMAddrIG_SUB_V_2_s} -- Reachable Addresses } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : IntReachATMAddrIG_SUB_V_1_r |
| ASN1 Type | : InternalReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre> { prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '39'0, -- Reachable Address Prefix padding '0000'0, -- Padding tlv_groups - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : IntReachATMAddrIG_SUB_V_1_s |
| ASN1 Type | : InternalReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '39'0, -- Reachable Address Prefix padding '0000'0, -- Padding tlv_groups - }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : IntReachATMAddrIG_SUB_V_2_r |
| ASN1 Type | : InternalReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '39'0, -- Reachable Address Prefix padding '0000'0, -- Padding tlv_groups {RAIG_SUB_Fldg_V_1_r, RAIG_SUB_Fldg_V_2_r, RAIG_SUB_Fldg_V_3_r, RAIG_SUB_Fldg_V_4_r, RAIG_SUB_Fldg_V_5_r} }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : IntReachATMAddrIG_SUB_V_2_s |
| ASN1 Type | : InternalReachableATMAddressesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Reachable ATM Addresses |
| Constraint Value | |
| <pre>{ prefix_len INT_TO_OCT(8,1), -- Prefix Length addr_prefix '39'0, -- Reachable Address Prefix padding '0000'0, -- Padding tlv_groups {RAIG_SUB_Fldg_V_1_s, RAIG_SUB_Fldg_V_2_s, RAIG_SUB_Fldg_V_3_s, RAIG_SUB_Fldg_V_4_s, RAIG_SUB_Fldg_V_5_s} }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3 |

| ASN.1 Type Constraint Declaration | |
|---|-------------------------------|
| Constraint Name | : NetworkIdData_Fldg_V_1_r |
| ASN1 Type | : NetworkIdentificationData_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Network identification data |
| Constraint Value | |
| <pre>{ bit_8_res '? 'B, -- Reserved bit_7_5_type '000 'B, -- Type of network identification (user-specified) bit_4_1_plan '0011 'B -- Network identification plan (data network identification code (X.121) }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.7, Table 4-21/Q.2931 | |

| ASN.1 Type Constraint Declaration | |
|---|-------------------------------|
| Constraint Name | : NetworkIdData_Fldg_V_1_s |
| ASN1 Type | : NetworkIdentificationData_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Network identification data |
| Constraint Value | |
| <pre>{ bit_8_res '0 'B, -- Reserved bit_7_5_type '000 'B, -- Type of network identification (user-specified) bit_4_1_plan '0011 'B -- Network identification plan (data network identification code (X.121) }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.7, Table 4-21/Q.2931 | |

| ASN.1 Type Constraint Declaration | |
|---|-----------------------------------|
| Constraint Name | : NodalHierarchyListIG_SUB_V_2_s |
| ASN1 Type | : NodalHierarchyListIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG Subtype |
| Constraint Value | |
| <pre>{ nhl_node tsp_LGN, -- Next Higher Level Logical Node ID nhl_addr tsp_AESA, -- Next Higher Level ATM End System Address nhl_pg tsp_LPG -- Next Higher Level Peer Group ID }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.2 | |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalHierarchyListIG_V_2_r |
| ASN1 Type | : NodalHierarchyListIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NHL), -- Type = 33 (NHL) length ?, -- Length seq_num ?, -- Sequence Number reserved '0000'O, -- Reserved level ?, -- Level Count nhl_sub * -- Nodal Hierarchy List IG Subtype } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.2 | |
| Nodal Hierarchy List IG to be sent by the IUT. (unknown) | |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalHierarchyListIG_V_2_s |
| ASN1 Type | : NodalHierarchyListIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NHL), -- Type = 33 (NHL) length INT_TO_OCT(12, 2), -- Length seq_num '00000001'O, -- Sequence Number reserved '0000'O, -- Reserved level '0000'O, -- Level Count nhl_sub - -- Nodal Hierarchy List IG Subtype } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.8.2 | |
| Nodal Hierarchy List (with empty list) IG to be sent by the tester | |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalHierarchyListIG_V_3_s |
| ASN1 Type | : NodalHierarchyListIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Hierarchy List IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NHL), -- Type = 33 (NHL) length INT_TO_OCT(68, 2), -- Length seq_num '00000001'O, -- Sequence Number reserved '0000'O, -- Reserved level '0001'O, -- Level Count nhl_sub {NodalHierarchyListIG_SUB_V_2_s} -- Nodal Hierarchy List IG Subtype } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8.2 Nodal Hierarchy List IG to be sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|--|-----------------------|
| Constraint Name | : NodalIG_Fldg_V_1_r |
| ASN1 Type | : NodalIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NODAL), -- Type = 97 (Nodal IG) length INT_TO_OCT(48,2), -- Length addr TAESA_2ND, -- tcv_TAESA, ATM End System Address of the originating node lead_prio '01'O, -- Leadership Priority nodal_flags Nodal_Flags_Fldg_V_1_r, -- Nodal Flags pref_pgl '0000000000000000000000000011111111111111111111111'O, -- Preferred PGL node ID nhl_bind - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Constraint Declaration | |
|--|-----------------------|
| Constraint Name | : NodalIG_Fldg_V_1_s |
| ASN1 Type | : NodalIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NODAL), -- Type = 97 (Nodal IG) length INT_TO_OCT(48,2), -- Length addr tcv_TAESA, -- ATM End System Address of the originating node lead_prio '01'O, -- Leadership Priority nodal_flags Nodal_Flags_Fldg_V_1_s, -- Nodal Flags pref_pgl '0000000000000000000000001111111111111111111111111111'O, -- Preferred PGL node ID nhl_bind - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSEAckIG_Fldg_V_1_r(id,seq:INTEGER; crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Acknowledgement IG for 1 PTSE |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NPA), -- Type = 384 length INT_TO_OCT(40,2), -- node tcv_TNID, -- Node ID ack_count INT_TO_OCT(1,2), -- PTSE Acknowledgement Count ptse_id {NodalPTSEAckIG_SUB_Fldg_V_1_r(id,seq,crc)} -- PTSE Identifier } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSEAckIG_Fldg_V_1_s(seq:INTEGER; nid, lt, crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Acknowledgement IG for 1 PTSE |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NPA), -- Type = 384 length INT_TO_OCT(40,2), -- node nid, -- Node ID ack_count INT_TO_OCT(1,2), -- PTSE Acknowledgement Count ptse_id {NodalPTSEAckIG_SUB_Fldg_V_1_s(seq,lt,crc)} -- PTSE Identifier } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : NodalPTSEAckIG_Fldg_V_2_s(nid, pid, seq, lt, crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Acknowledgement IG for 1 PTSE |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NPA), -- Type = 384 length INT_TO_OCT(40,2), -- node nid, -- Node ID ack_count INT_TO_OCT(1,2), -- PTSE Acknowledgement Count ptse_id {NodalPTSEAckIG_SUB_Fldg_V_2_s(pid,seq,lt,crc)} } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSEAckIG_SUB_Fldg_V_1_r(id,seq:INTEGER; crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Identifier |
| Constraint Value | |
| <pre> { ptse_ident INT_TO_OCT(id,4), -- PTSE Identifier ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc crc, -- PTSE Checksum ptse_ttl ? -- PTSE Remaining Lifetime } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : NodalPTSEAckIG_SUB_Fldg_V_1_s(seq:INTEGER; lt, crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Identifier |
| Constraint Value | |
| <pre> { ptse_ident INT_TO_OCT(1,4), -- PTSE Identifier ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc crc, -- PTSE Checksum ptse_ttl lt -- PTSE Remaining Lifetime } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSEAckIG_SUB_Fldg_V_2_s(pid, seq, lt, crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Identifier |
| Constraint Value | |
| <pre>{ ptse_ident pid, -- PTSE Identifier ptse_seq_no seq, -- PTSE Sequence Number ptse_crc crc, -- PTSE Checksum ptse_ttl lt -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSEAckIG_SUB_V_1_r(id,seq,crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Identifier |
| Constraint Value | |
| <pre>{ ptse_ident id, -- PTSE Identifier ptse_seq_no seq, -- PTSE Sequence Number ptse_crc crc, -- PTSE Checksum ptse_ttl ? -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSEAckIG_V_1_r(id,seq,crc:OCTETSTRING) |
| ASN1 Type | : NodalPTSEAcknowledgementIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal PTSE Acknowledgement IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_NPA), -- Type = 384 length INT_TO_OCT(40,2), -- node tcv_TNID, -- Node ID ack_count INT_TO_OCT(1,2), -- PTSE Acknowledgement Count ptse_id {NodalPTSEAckIG_SUB_V_1_r(id,seq,crc)} -- PTSE Identifier }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------------|
| Constraint Name | : NodalPTSERequestListIG_V_1_r |
| ASN1 Type | : NodalPTSERequestListIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal PTSE Request List IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_RPH), -- Type = 513 length '0020'O, -- Length o_node tcv_TNID, -- Originating Node ID req_count ?, -- PTSE Request Count ptse_id {'00000001'O} -- PTSE Identifier } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.12 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSESummariesIG_I_1_s |
| ASN1 Type | : NodalPTSESummariesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Sequence of PNNI Topology State Packets (PTSP) and PNNI Topology State Elements (PTSE) header information of the tester, stating that it was originated by the IUT (by setting the originating node ID to the node ID of the IUT). |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NPS), length INT_TO_OCT(60,2), o_node tsp_NID, -- node ID of the IUT o_pg tsp_PGID, reserved '0000'O, ptse_sc '0001'O, ptse_seq {NodalPTSESummariesIG_SUB_V_3_s} } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 Topology database information sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSESummariesIG_SUB_V_1_s |
| ASN1 Type | : NodalPTSESummariesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element (PTSE) header information for the Nodal Information Group IG (97) sent by the tester |
| Constraint Value | |
| <pre>{ ptse_type '0061'O, -- PTSEType reserved '0000'O, -- ptse_ident '00000001'O, -- PTSE Identifier ptse_seq_no '00000001'O, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE Checksum ptse_ttl INT_TO_OCT(3600, 2) -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSESummariesIG_SUB_V_2_s |
| ASN1 Type | : NodalPTSESummariesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element (PTSE) header information sent by the tester |
| Constraint Value | |
| <pre>{ ptse_type tcv_PTSE_TYPE, -- PTSEType reserved '0000'O, -- ptse_ident tcv_PTSE_ID, -- PTSE Identifier ptse_seq_no tcv_PTSE_SEQ, -- PTSE Sequence Number ptse_crc '83A3'O, -- PTSE Checksum ptse_ttl INT_TO_OCT(3600, 2) -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : NodalPTSESummariesIG_SUB_V_3_s |
| ASN1 Type | : NodalPTSESummariesIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element (PTSE) header information for the Nodal State Parameters IG (96) sent by the tester |
| Constraint Value | |
| <pre>{ ptse_type '0060'O, -- PTSEType reserved '0000'O, -- ptse_ident '0000000F'O, -- PTSE Identifier ptse_seq_no '00000001'O, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE Checksum ptse_ttl INT_TO_OCT(3600, 2) -- PTSE Remaining Lifetime }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSESummariesIG_V_1_r |
| ASN1 Type | : NodalPTSESummariesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Initial empty sequence for database summary with higher level node ID of the Tester. |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_NPS), length ?, o_node ?, o_pg ?, reserved ?, ptse_sc ?, ptse_seq ? }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 Topology database information received from the IUT. |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : NodalPTSESummariesIG_V_1_s |
| ASN1 Type | : NodalPTSESummariesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Sequence of PNNI Topology State Packets (PTSP) and PNNI Topology State Elements (PTSE) header information of the tester (Slave). |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_NPS), length INT_TO_OCT(60,2), o_node tcv_TNID, o_pg tsp_PGID, reserved '0000'O, ptse_sc '0001'O, ptse_seq {NodalPTSESummariesIG_SUB_V_1_s} }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 Topology database information sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : NodalPTSESummariesIG_V_2_s |
| ASN1 Type | : NodalPTSESummariesIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Sequence of PNNI Topology State Packets (PTSP) and PNNI Topology State Elements (PTSE) header information of the tester (Slave). |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NPS), length INT_TO_OCT(60,2), o_node tsp_NID, o_pg tsp_PGID, reserved '0000'O, ptse_sc '0001'O, ptse_seq {NodalPTSESummariesIG_SUB_V_2_s} } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 Topology database information sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|---|-------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_10_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_8_crm_r} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|-------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_11_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_8_vf_r} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_1_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(80, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_cdv_r, ResourceAvaIG_Fldg_V_2_cdv_r} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_1_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(80, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_s, ResourceAvaIG_Fldg_V_2_s} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_2_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(112, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_ctd_r, ResourceAvaIG_Fldg_V_2_ctd_r, ResourceAvaIG_Fldg_V_3_ctd_r} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_2_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(112, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_s, ResourceAvaIG_Fldg_V_2_s, ResourceAvaIG_Fldg_V_3_s} -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_3_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(144, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_2_aw_r, ResourceAvaIG_Fldg_V_3_aw_r, ResourceAvaIG_Fldg_V_4_aw_r, ResourceAvaIG_Fldg_V_5_aw_r } -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_3_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(144, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_2_s, ResourceAvaIG_Fldg_V_3_s, ResourceAvaIG_Fldg_V_4_s, ResourceAvaIG_Fldg_V_5_s } -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_4_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(112, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_clr0_r, ResourceAvaIG_Fldg_V_2_clr0_r, ResourceAvaIG_Fldg_V_3_clr0_r} -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_4_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(80, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_5_s, ResourceAvaIG_Fldg_V_6_s} -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_5_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(112, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_1_clr01_r, ResourceAvaIG_Fldg_V_2_clr01_r, ResourceAvaIG_Fldg_V_3_clr01_r} -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_5_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(144, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_2_s, ResourceAvaIG_Fldg_V_3_s, ResourceAvaIG_Fldg_V_1_s, ResourceAvaIG_Fldg_V_5_s } -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_6_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(80, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_5_mcr_r, ResourceAvaIG_Fldg_V_6_mcr_r} -- Outgoing Resource Availability IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_6_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_7_s } -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_7_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(144, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_2_acr_r, ResourceAvaIG_Fldg_V_3_acr_r, ResourceAvaIG_Fldg_V_1_acr_r, ResourceAvaIG_Fldg_V_5_acr_r } -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_7_s |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_s, -- VP Capability Flag reserved '0000'O, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_8_s } -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|---|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_8_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_7_crm_r} -- Outgoing Resource Availablility IG } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|------------------------------|
| Constraint Name | : NodalStateParIG_Fldg_V_9_r |
| ASN1 Type | : NodalStateParametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal State Parameters IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_NSP), -- Type = 96 (Nodal State Parameters) length INT_TO_OCT(60, 2), -- Length flags VP_Cap_Flag_Fldg_V_1_r, -- VP Capability Flag reserved ?, in_port INT_TO_OCT(6543,4), -- Input Port ID out_port INT_TO_OCT(6542,4), -- Output Port ID out_raig {ResourceAvaIG_Fldg_V_7_vf_r} -- Outgoing Resource Availability IG }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------|
| Constraint Name | : Nodal_Flags_Fldg_V_1_r |
| ASN1 Type | : Nodal_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Flags |
| Constraint Value | |
| <pre>{ bit_8_leader '0'B, -- 'I am Leader' bit bit_7_restr_trans '0'B, -- Restricted Transit bit bit_6_nodal_repr '0'B, -- Nodal Representation bit bit_5_restr_branch '0'B, -- Restricted Branching bit bit_4_non_trans '0'B, -- Non-Transit for PGL Election bit bit_3_l_res '???'B -- Reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------|
| Constraint Name | : Nodal_Flags_Fldg_V_1_s |
| ASN1 Type | : Nodal_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Nodal Flags |
| Constraint Value | |
| <pre>{ bit_8_leader '0'B, -- 'I am Leader' bit bit_7_restr_trans '0'B, -- Restricted Transit bit bit_6_nodal_repr '0'B, -- Nodal Representation bit bit_5_restr_branch '0'B, -- Restricted Branching bit bit_4_non_trans '0'B, -- Non-Transit for PGL Election bit bit_3_l_res '000'B -- Reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.2 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : OptionalGCACparIG_Fldg_V_1_crm_r |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for rt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm RTVBR_CRM, -- Cell Rate Margin variance ? -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : OptionalGCACparIG_Fldg_V_1_s |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for rt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm RTVBR_CRM, -- Cell Rate Margin variance RTVBR_VF -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : OptionalGCACparIG_Fldg_V_1_vf_r |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for rt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm ?, -- Cell Rate Margin variance RTVBR_VF -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : OptionalGCACparIG_Fldg_V_2_crm_r |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for nrt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm NRTVBR_CRM, -- Cell Rate Margin variance ? -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : OptionalGCACparIG_Fldg_V_2_s |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for nrt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm NRTVBR_CRM, -- Cell Rate Margin variance NRTVBR_VF -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : OptionalGCACparIG_Fldg_V_2_vf_r |
| ASN1 Type | : OptionalGCACparametersIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional Generic Connection Admission Control Parameters IG (for nrt-VBR) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_GCAC), -- Type = 160 (GCAC) length INT_TO_OCT(12,2), -- Length crm ?, -- Cell Rate Margin variance NRTVBR_VF -- Variance Factor } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.5 | |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : PTSEIG_Fldg_V_0_r(type, id, seq:INTEGER) |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT (with TTL = 0) |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(20,2), -- Length ptse_type INT_TO_OCT(type,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(id,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(EXPIRED_AGE,2), -- PTSE remaining lifetime ig - -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_0_s(type, id, seq:INTEGER; crc:OCTETSTRING) |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(20,2), -- Length ptse_type INT_TO_OCT(type,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(id,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc crc, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(EXPIRED_AGE,2), -- PTSE remaining lifetime ig - -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_10_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(7,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_10_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_10_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(360, 2), -- Length ptse_type INT_TO_OCT(224,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(10,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_10_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_11_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(8,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_11_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_11_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(40, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(11,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_11_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_12_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(8,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_12_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_12_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(360, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(12,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_12_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_13_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(40, 2), -- Length ptse_type INT_TO_OCT(224,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(9,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_13_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_13_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(52, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved '0000'O, -- reserved ptse_id INT_TO_OCT(13,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_13_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_14_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(360, 2), -- Length ptse_type INT_TO_OCT(224,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(10,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_14_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_14_s(seq, lt:INTEGER) |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(68, 2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(1,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(lt,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_1_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_15_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(40, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(11,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_15_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_16_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(360, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(12,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_16_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_17_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(52, 2), -- Length ptse_type INT_TO_OCT(256,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(13,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_17_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_18_r(id:OCTETSTRING;seq:INTEGER) |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element received from the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(68, 2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(1,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(seq,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_1_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_1_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element received from the IUT |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(68, 2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(1,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_1_r -- PTSE Information Groups }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_1_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(68, 2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(1,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_1_s -- PTSE Information Groups }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_2_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(100, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(2,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_2_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_2_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(100, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(2,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_2_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_3_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(132, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(3,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_3_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_3_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(132, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(3,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_3_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_4_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(164, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(4,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_4_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_4_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(164, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(4,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_4_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_5_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(132, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(3,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_5_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_5_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(100, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(5,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_5_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_6_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(132, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(3,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_6_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_6_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(164, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(6,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_6_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_7_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(100, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(5,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_7_r -- PTSE Information Groups }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_7_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(7,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_7_s -- PTSE Information Groups }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_8_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(164, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(6,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_8_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_Fldg_V_8_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(8,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_8_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_9_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(80, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id INT_TO_OCT(7,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_9_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_Fldg_V_9_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(40, 2), -- Length ptse_type INT_TO_OCT(224,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id INT_TO_OCT(9,4), -- Identifies one of multiple different PTSEs from a node ptse_seq_no INT_TO_OCT(1,4), -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_9_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_V_1_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length ?, -- Length ptse_type ?, -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id ?, -- Identifies one of multiple different PTSEs from a node ptse_seq_no ?, -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig PTSE_IGs_V_1_r -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_V_1_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(68, 2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved '0000'0, -- reserved ptse_id '00000001'0, -- Identifies one of multiple different PTSEs from a node ptse_seq_no '00000001'0, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_1_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_V_2_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length ?, -- Length ptse_type tcv_PTSE_TYPE, -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id tcv_PTSE_ID, -- Identifies one of multiple different PTSEs from a node ptse_seq_no ?, -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig ?, -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSEIG_V_2_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(20,2), -- Length ptse_type INT_TO_OCT(97,2), -- Indicates which restricted IGs are allowed reserved '0000'O, -- reserved ptse_id '00000001'O, -- Identifies one of multiple different PTSEs from a node ptse_seq_no '00000001'O, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl '0000'O, -- PTSE remaining lifetime ig - -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : PTSEIG_V_3_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT with PTSE without content and the Remaining Lifetime set to ExpiredAge |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length '0014'O, -- Length ptse_type ?, -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id '0000000F'O, -- Identifies one of multiple different PTSEs from a node ptse_seq_no '00000002'O, -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl '0000'O, -- PTSE remaining lifetime ig - -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_V_3_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(100, 2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'O, -- reserved ptse_id '00000002'O, -- Identifies one of multiple different PTSEs from a node ptse_seq_no '00000001'O, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl INT_TO_OCT(600,2), -- PTSE remaining lifetime ig PTSE_IGs_Fldg_V_2_s -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_V_4_r |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_PTSE), -- Type = 64 (PTSE) length ?, -- Length ptse_type ?, -- Indicates which restricted IGs are allowed reserved ?, -- reserved ptse_id ?, -- Identifies one of multiple different PTSEs from a node ptse_seq_no ?, -- PTSE Sequence Number ptse_crc ?, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl ?, -- PTSE remaining lifetime ig *, -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSEIG_V_4_s |
| ASN1 Type | : PTSEIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Element sent by the IUT |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_PTSE), -- Type = 64 (PTSE) length INT_TO_OCT(20,2), -- Length ptse_type INT_TO_OCT(96,2), -- Indicates which restricted IGs are allowed reserved '0000'O, -- reserved ptse_id '00000002'O, -- Identifies one of multiple different PTSEs from a node ptse_seq_no '00000001'O, -- PTSE Sequence Number ptse_crc tcv_CRC, -- PTSE checksum (see PNNI 1.0 5.8.2.2.2) ptse_ttl '0000'O, -- PTSE remaining lifetime ig -, -- PTSE Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_10_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_9_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_10_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr {IntReachATMAddrIG_Fldg_V_2_s}, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_11_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_10_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_11_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_1_s}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_12_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_11_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_12_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_2_s}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_13_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr {IntReachATMAddrIG_Fldg_V_1_r}, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_13_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_3_s}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_14_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr {IntReachATMAddrIG_Fldg_V_2_r}, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_15_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_1_r}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_16_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_2_r}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_17_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr {ExtReachATMAddrIG_Fldg_V_3_r}, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_1_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig {NodalIG_Fldg_V_1_r}, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_1_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig {NodalIG_Fldg_V_1_s}, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_2_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_1_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_2_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_1_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_3_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_2_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_3_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_2_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_4_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_3_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_4_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_3_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : PTSE_IGs_Fldg_V_5_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_4_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_5_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_4_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_6_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_5_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_6_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_5_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_7_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_6_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_7_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_6_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_8_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_7_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_8_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_7_s}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_9_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par {NodalStateParIG_Fldg_V_8_r}, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_Fldg_V_9_s |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr {IntReachATMAddrIG_Fldg_V_1_s}, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links -, -- Horizontal Links uplinks -, -- Uplinks sys_cap - -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PTSE_IGs_V_1_r |
| ASN1 Type | : PTSE_IGs_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PTSE Information Groups sent by the IUT |
| Constraint Value | |
| <pre> { nodal_ig -, -- Nodal Information Group nodal_state_par -, -- Nodal State Parameter int_reach_addr -, -- Internal Reachable ATM Addresses ext_reach_addr -, -- Exterior Reachable ATM Addresses horizontal_links {HorizontalLinksIG_V_1_r}, -- Horizontal Links uplinks -, -- Uplinks sys_cap -, -- System Capabilities } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.3 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PnniHeader_V_1_r(packet_type:OCTETSTRING; packet_length, version:INTEGER) |
| ASN1 Type | : PnniHeader_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Packet Header |
| Constraint Value | |
| <pre> { type packet_type, -- Packet Type length INT_TO_OCT(packet_length, 2), -- Packet Length version INT_TO_OCT(version, 1), -- Protocol Version n_version INT_TO_OCT(tsp_NV, 1), -- Newest Version Supported o_version INT_TO_OCT(tsp_OV, 1), -- Oldest Version Supported reserved '00'0 -- Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : PnniHeader_V_1_s(packet_type:OCTETSTRING; packet_length, version:INTEGER) |
| ASN1 Type | : PnniHeader_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Packet Header |
| Constraint Value | |
| <pre> { type packet_type, -- Packet Type length INT_TO_OCT(packet_length, 2), -- Packet Length version INT_TO_OCT(version, 1), -- Protocol Version n_version INT_TO_OCT(tsp_NV, 1), -- Newest Version Supported o_version INT_TO_OCT(tsp_OV, 1), -- Oldest Version Supported reserved '00'0 -- Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.4 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : PnniHeader_V_2_r(packet_type:OCTETSTRING; version:INTEGER) |
| ASN1 Type | : PnniHeader_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : PNNI Packet Header |
| Constraint Value | |
| <pre> { type packet_type, -- Packet Type length ?, -- Packet Length version INT_TO_OCT(version, 1), -- Protocol Version n_version INT_TO_OCT(tsp_NV, 1), -- Newest Version Supported o_version INT_TO_OCT(tsp_OV, 1), -- Oldest Version Supported reserved '00'0 -- Reserved } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.4 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_1_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with CBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '1'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res '?', -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_1_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with CBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '1'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|-------------------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_2_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with Real Time VBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '1'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res ?, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|-------------------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_2_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with Real Time VBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '1'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : RAIG_Flags_Fldg_V_3_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with Non-Real Time VBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '1'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res ?, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : RAIG_Flags_Fldg_V_3_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with Non-Real Time VBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '1'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|-----------------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_4_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with CBR and UBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '1'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '1'B, -- UBR bit_11_2_res '?', -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|-----------------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_4_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with CBR and UBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '1'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '1'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_5_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with ABR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '1'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res ?, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_5_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with ABR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '1'B, -- ABR bit_12_ubr '0'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_6_r |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with UBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '1'B, -- UBR bit_11_2_res ?, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---------------------------|
| Constraint Name | : RAIG_Flags_Fldg_V_6_s |
| ASN1 Type | : RAIG_Flags_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : RAIG Flags with UBR set |
| Constraint Value | |
| <pre> { bit_16_cbr '0'B, -- CBR bit_15_rt_vbr '0'B, -- rt-VBR bit_14_nrt_vbr '0'B, -- nrt-VBR bit_13_abr '0'B, -- ABR bit_12_ubr '1'B, -- UBR bit_11_2_res '0000000000'B, -- Reserved bit_1_gcac_clp_attr '0'B -- GCAC CLP Attribute } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_1_r |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre> { out_raig ResourceAvaIG_Fldg_V_1_r, in_raig ResourceAvaIG_Fldg_V_11_r } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_1_s |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre> { out_raig ResourceAvaIG_Fldg_V_1_s, in_raig ResourceAvaIG_Fldg_V_11_s } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_2_r |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_2_r, in_raig ResourceAvaIG_Fldg_V_12_r }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_2_s |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_2_s, in_raig ResourceAvaIG_Fldg_V_12_s }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_3_r |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_3_r, in_raig ResourceAvaIG_Fldg_V_13_r }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_3_s |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_3_s, in_raig ResourceAvaIG_Fldg_V_13_s }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_4_r |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_5_r, in_raig ResourceAvaIG_Fldg_V_15_r }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_4_s |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_5_s, in_raig ResourceAvaIG_Fldg_V_15_s }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_5_r |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_6_r, in_raig ResourceAvaIG_Fldg_V_16_r }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : RAIG_SUB_Fldg_V_5_s |
| ASN1 Type | : RAIG_SUB_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Optional TLV groups for resource availability information |
| Constraint Value | |
| <pre>{ out_raig ResourceAvaIG_Fldg_V_6_s, in_raig ResourceAvaIG_Fldg_V_16_s }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.3/4 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_11_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_s, -- RAIG Flags reserved ?, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_11_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_s, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_12_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_INRA), -- Type = 129 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr RTVBR_MCR, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_12_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_INRA), -- Type = 129 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_s, -- RAIG Flags reserved '0000'0, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr RTVBR_MCR, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_13_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved '?', -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr NRTVBR_MCR, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv NRTVBR_CDV, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_13_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_s, -- RAIG Flags reserved '0000'0, -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr NRTVBR_MCR, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv NRTVBR_CDV, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_15_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_r, -- RAIG Flags reserved ?, -- Reserved weight ABR_AW, -- Administrative Weight (default Value = 5040) mcr ABR_MCR, -- Maximum Cell Rate acr ABR_ACR, -- Available Cell Rate ctd ABR_CTD, -- Cell Transfer Delay cdv ABR_CDV, -- Cell Delay Variation clr_0 ABR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ABR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_15_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_s, -- RAIG Flags reserved '0000'0, -- Reserved weight ABR_AW, -- Administrative Weight (default Value = 5040) mcr ABR_MCR, -- Maximum Cell Rate acr ABR_ACR, -- Available Cell Rate ctd ABR_CTD, -- Cell Transfer Delay cdv ABR_CDV, -- Cell Delay Variation clr_0 ABR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ABR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_16_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_6_r, -- RAIG Flags reserved ?, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_16_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_INRA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_6_s, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_acr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory. |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_cdv_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory. |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_clr01_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory. |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_clr0_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory. |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_ctd_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory. |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_r, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_1_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_1_s, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_acr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_aw_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_cdv_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_clr01_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_clr0_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_ctd_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr RTVBR_MCR, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_2_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_2_s, -- RAIG Flags reserved '0000'0, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr RTVBR_MCR, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_acr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_aw_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_clr01_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_clr0_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_ctd_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr NRTVBR_MCR, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv NRTVBR_CDV, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_3_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_3_s, -- RAIG Flags reserved '0000'0, -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr NRTVBR_MCR, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv NRTVBR_CDV, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_4_aw_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR and UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_4_r, -- RAIG Flags reserved ?, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_4_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with CBR and UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_4_s, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_5_acr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ABR_ACR, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_5_aw_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_r, -- RAIG Flags reserved ?, -- Reserved weight ABR_AW, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_5_mcr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ABR_MCR, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_5_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_r, -- RAIG Flags reserved ?, -- Reserved weight ABR_AW, -- Administrative Weight (default Value = 5040) mcr ABR_MCR, -- Maximum Cell Rate acr ABR_ACR, -- Available Cell Rate ctd ABR_CTD, -- Cell Transfer Delay cdv ABR_CDV, -- Cell Delay Variation clr_0 ABR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ABR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_5_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_5_s, -- RAIG Flags reserved '0000'0, -- Reserved weight ABR_AW, -- Administrative Weight (default Value = 5040) mcr ABR_MCR, -- Maximum Cell Rate acr ABR_ACR, -- Available Cell Rate ctd ABR_CTD, -- Cell Transfer Delay cdv ABR_CDV, -- Cell Delay Variation clr_0 ABR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 ABR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_6_mcr_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with ABR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_6_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_6_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_6_r, -- RAIG Flags reserved ?, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_6_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with UBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(32, 2), -- Length flags RAIG_Flags_Fldg_V_6_s, -- RAIG Flags reserved '0000'0, -- Reserved weight CBR_AW, -- Administrative Weight (default Value = 5040) mcr CBR_MCR, -- Maximum Cell Rate acr CBR_ACR, -- Available Cell Rate ctd CBR_CTD, -- Cell Transfer Delay cdv CBR_CDV, -- Cell Delay Variation clr_0 CBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 CBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac - -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_7_crm_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_1_crm_r -- Optional GCAC Parameters } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|--|
| Constraint Name | : ResourceAvaIG_Fldg_V_7_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with rt-VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_2_s, -- RAIG Flags reserved '0000'0, -- Reserved weight RTVBR_AW, -- Administrative Weight (default Value = 5040) mcr RTVBR_MCR, -- Maximum Cell Rate acr RTVBR_ACR, -- Available Cell Rate ctd RTVBR_CTD, -- Cell Transfer Delay cdv RTVBR_CDV, -- Cell Delay Variation clr_0 RTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 RTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_1_s -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_7_vf_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_2_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_1_vf_r -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_8_crm_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_2_crm_r -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_8_s |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with nrt-VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_3_s, -- RAIG Flags reserved '0000'0, -- Reserved weight NRTVBR_AW, -- Administrative Weight (default Value = 5040) mcr NRTVBR_MCR, -- Maximum Cell Rate acr NRTVBR_ACR, -- Available Cell Rate ctd NRTVBR_CTD, -- Cell Transfer Delay cdv NRTVBR_CDV, -- Cell Delay Variation clr_0 NRTVBR_CLR0, -- Cell Loss Ratio (CLP=0) clr_01 NRTVBR_CLR10, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_2_s -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|---|---|
| Constraint Name | : ResourceAvaIG_Fldg_V_8_vf_r |
| ASN1 Type | : ResourceAvailablilityIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Resource Availablility IG with Non-Real Time VBR service catagory |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ORA), -- Type = 128 (Outgoing RA) Type = 129 (Incoming RA) length INT_TO_OCT(44, 2), -- Length flags RAIG_Flags_Fldg_V_3_r, -- RAIG Flags reserved ?, -- Reserved weight ?, -- Administrative Weight (default Value = 5040) mcr ?, -- Maximum Cell Rate acr ?, -- Available Cell Rate ctd ?, -- Cell Transfer Delay cdv ?, -- Cell Delay Variation clr_0 ?, -- Cell Loss Ratio (CLP=0) clr_01 ?, -- Cell Loss Ratio (CLP=0+1) gcac OptionalGCACparIG_Fldg_V_2_vf_r -- Optional GCAC Parameters }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.5 |

| ASN.1 Type Constraint Declaration | |
|--|-----------------------------|
| Constraint Name | : TransitNetIdIG_Fldg_V_1_r |
| ASN1 Type | : TransitNetworkIdIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Transit Network ID IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_TNI), -- Type = 304 length INT_TO_OCT(12,2), -- Length tns_length INT_TO_OCT(3,2), -- Length of TNS n_id_data NetworkIdData_Fldg_V_1_r, -- Network identification data n_id '1234'O, -- Network identification padding '000000'O -- Padding }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.7 |

| ASN.1 Type Constraint Declaration | |
|--|-----------------------------|
| Constraint Name | : TransitNetIdIG_Fldg_V_1_s |
| ASN1 Type | : TransitNetworkIdIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Transit Network ID IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_TNI), -- Type = 304 length INT_TO_OCT(12,2), -- Length tns_length INT_TO_OCT(3,2), -- Length of TNS n_id_data NetworkIdData_Fldg_V_1_s, -- Network identification data n_id '1234'O, -- Network identification padding '000000'O -- Padding } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.7 |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : UplinkInformationAttributeIG_V_1_r |
| ASN1 Type | : UplinkInformationAttributeIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Uplink Information Attribute IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_r(IGT_ULIA), -- Type = 34 (ULIA) length ?, -- Length seq_num ?, -- Sequence Number out_ra * -- Outgoing Resource Availability } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.6 Uplink Information Attribute IG to be sent by the IUT. |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : UplinkInformationAttributeIG_V_1_s |
| ASN1 Type | : UplinkInformationAttributeIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Uplink Information Attribute IG |
| Constraint Value | |
| <pre> { type IG_Type_V_1_s(IGT_ULIA), -- Type = 34 (ULIA) length INT_TO_OCT(8, 2), -- Length seq_num INT_TO_OCT(SEQ_NUM, 4), -- Sequence Number out_ra - -- Outgoing Resource Availability } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.6 Uplink Information Attribute IG to be sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|--|---|
| Constraint Name | : UplinkInformationAttributeIG_V_2_r |
| ASN1 Type | : UplinkInformationAttributeIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Uplink Information Attribute IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_r(IGT_ULIA), -- Type = 34 (ULIA) length ?, -- Length seq_num INT_TO_OCT(SEQ_NUM+1, 4), -- Sequence Number out_ra * -- Outgoing Resource Availability }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.6 Uplink Information Attribute IG (with new ULIA sequence number) to be sent by the IUT. |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : UplinkInformationAttributeIG_V_2_s |
| ASN1 Type | : UplinkInformationAttributeIG_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : Uplink Information Attribute IG |
| Constraint Value | |
| <pre>{ type IG_Type_V_1_s(IGT_ULIA), -- Type = 34 (ULIA) length INT_TO_OCT(8, 2), -- Length seq_num INT_TO_OCT(SEQ_NUM+1, 4), -- Sequence Number out_ra - -- Outgoing Resource Availability }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.6 Uplink Information Attribute IG (with new ULIA sequence number) to be sent by the tester. |

| ASN.1 Type Constraint Declaration | |
|--|--------------------------|
| Constraint Name | : VP_Cap_Flag_Fldg_V_1_r |
| ASN1 Type | : VP_Capabilities_Flag_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : VP Capability Flag |
| Constraint Value | |
| <pre>{ bit_16_vp_capability '0'B, -- VP Capability Flag bit_15_l_res ? -- reserved }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 Type Constraint Declaration | |
|--|--|
| Constraint Name | : VP_Cap_Flag_Fldg_V_1_s |
| ASN1 Type | : VP_Capabilities_Flag_T |
| Derivation Path | : |
| Encoding Variation | : |
| Comments | : VP Capability Flag |
| Constraint Value | |
| { | |
| bit_16_vp_capability | '0'B, -- VP Capability Flag |
| bit_15_1_res | '0000000000000000'B -- reserved |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9.1.1 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : DBSP_I_1_s(init, more, master : BITSTRING; d: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet with Nodal PTSE summaries send by the Tester as an answer on a received Database Summary Packet, stating that it was originated by the IUT (by setting the originating node ID to the node ID of the IUT). |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_DB_SUMM, 76, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s(init, more, master), reserved '0000'O, ds_seq_no INT_TO_OCT(d,4), ptsp_seq {NodalPTSESummariesIG_I_1_s} }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : DBSP_V_1_r |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Empty Database Summary Packet |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_2_r(PT_DB_SUMM, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_r('1'B, '1'B, '1'B), reserved ?, ds_seq_no ?, ptsp_seq - }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : DBSP_V_1_s(dsno: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Empty Master Database Summary Packet with higher level LGN ID of the Tester. |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_s(PT_DB_SUMM, 16, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s('1'B, '1'B, '1'B), reserved '0000'O, ds_seq_no INT_TO_OCT(dsno,4), ptsp_seq - }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.11 | |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : DBSP_V_2_r(init, more, master : BITSTRING; dsno: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Answer without PTSE header information of the IUT on a received Database Summary Packet. |
| Constraint Value | |
| <pre>{ header PnniHeader_V_2_r(PT_DB_SUMM, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_r(init, more, master), reserved ?, ds_seq_no INT_TO_OCT(dsno,4), ptsp_seq - }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.11 | |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : DBSP_V_2_s(dsno: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Slave Database Summary Answer on a received empty Database Summary Packet, additional PTSEs have to be summarized. |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_DB_SUMM, 16, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s('0'B, '1'B, '0'B), -- additional PTSEs have to be summarized reserved '0000'0, ds_seq_no INT_TO_OCT(dsno,4), ptsp_seq - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : DBSP_V_3_r(init, more, master : BITSTRING; dsno: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Answer with PTSE header information of the IUT on a received Database Summary Packet. |
| Constraint Value | |
| <pre> { header PnniHeader_V_2_r(PT_DB_SUMM, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_r(init, more, master), reserved ?, ds_seq_no INT_TO_OCT(dsno,4), ptsp_seq ? } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : DBSP_V_3_s(init, more, master : BITSTRING; d: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet with empty Nodal PTSE summaries send by the Tester as an answer on a received Database Summary Packet |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_DB_SUMM, 16, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s(init, more, master), reserved '0000'O, ds_seq_no INT_TO_OCT(d,4), ptsp_seq - }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : DBSP_V_4_r(init, more, master : BITSTRING; dsno: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Answer with PTSE header information of the IUT on a received Database Summary Packet. |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_2_r(PT_DB_SUMM, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_r(init, more, master), reserved ?, ds_seq_no INT_TO_OCT(dsno,4), ptsp_seq * }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : DBSP_V_4_s(init, more, master : BITSTRING; d: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet with Nodal PTSE summaries send by the Tester as an answer on a received Database Summary Packet |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_DB_SUMM, 76, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s(init, more, master), reserved '0000'O, ds_seq_no INT_TO_OCT(d,4), ptsp_seq {NodalPTSESummariesIG_V_1_s} }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : DBSP_V_5_s(init, more, master : BITSTRING; d: INTEGER) |
| PDU Type | : DBSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : Database Summary Packet with Nodal PTSE summaries send by the Tester as an answer on a received Database Summary Packet |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_DB_SUMM, 76, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags DBSum_Packet_Flags_V_1_s(init, more, master), reserved '0000'O, ds_seq_no INT_TO_OCT(d,4), ptsp_seq {NodalPTSESummariesIG_V_2_s} }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.11 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : Hello_I_1_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| { | <pre> header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Error in Originating Node ID addr tsp_TAESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node 'FF'0, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port tsv_R_PID, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups </pre> |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| Two-Way-Inside Hello with a mismatch (error in remote node ID field) to be sent by the tester. Should be interpreted by the IUT as a Hello Mismatch Received. | |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : Hello_I_2_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| { | <pre> header PnniHeader_V_1_s(PT_HELLO, 184, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Originating Node ID addr tsp_TAESA, -- ATM End System Address o_pg tsp_TPGID, -- Originating Peer Group ID r_node tsp_NID, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port tsv_R_PID, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig Hello_IGs_I_1_s -- Hello Information Groups </pre> |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| Two-Way-Inside Hello to be sent by the tester with madatory bit tag from invalid Aggregation Token TLV set to one. | |

| ASN.1 PDU Constraint Declaration | |
|--|---------------------|
| Constraint Name | : Hello_I_3_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tcv_TNID, addr tsp_TAESA, o_pg tsp_PGID, r_node '00'O, o_port tsp_TPID, r_port '00000000'O, hello_int '0000'O, reserved '0000'O, ig - }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| One-Way-Inside Hello to be sent by the tester with HelloInterval is set to zero. | |

| ASN.1 PDU Constraint Declaration | |
|---|---------------------|
| Constraint Name | : Hello_I_4_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tcv_TNID, addr tsp_TAESA, o_pg tsp_PGID, r_node '00'O, o_port '00000000'O, r_port '00000000'O, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig - }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| One-Way-Inside Hello to be sent by the tester with port ID is set to zero. | |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : Hello_I_5_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node 'FF'O, addr tsp_TAESA, o_pg tsp_TPGID, r_node '00'O, o_port tsp_TPID, r_port '00000000'O, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 One-Way-Outside Hello with a mismatch (error in the node id field) to be sent by the tester. Should be interpreted by the IUT as a Hello Mismatch Received. |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : Hello_V_1_r |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_HELLO, 100, tsp_NV), flags '0000'O, o_node tsp_NID, addr tsp_AESA, o_pg tsp_PGID, r_node '00'O, o_port ?, r_port '00000000'O, hello_int ?, reserved '0000'O, ig - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 The first Hello to be sent by the IUT, where the Port ID field has a value but unknown by the tester. |

| ASN.1 PDU Constraint Declaration | |
|--|--------------------------------|
| Constraint Name | : Hello_V_1_s(version:INTEGER) |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_s(PT_HELLO, 100, version), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Originating Node ID addr tsp_TAESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node '00'O, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port '00000000'O, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| One-Way-Inside Hello to be sent by the tester with Version field as a parameter. | |

| ASN.1 PDU Constraint Declaration | |
|--|--------------------------------|
| Constraint Name | : Hello_V_2_r(version:INTEGER) |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_r(PT_HELLO, 100, version), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tsp_NID, -- Originating Node ID addr tsp_AESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node tcv_TNID, -- Remote Node ID o_port tsv_R_PID, -- Port ID r_port tsp_TPID, -- Remote Port ID hello_int ?, -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups }</pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| Two-Way-Inside Hello to be sent by the IUT with version field as an parameter. | |

| ASN.1 PDU Constraint Declaration | |
|--|---------------------|
| Constraint Name | : Hello_V_2_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tcv_TNID, addr tsp_TAESA, o_pg tsp_PGID, r_node '00'O, o_port tsp_TPID, r_port '00000000'O, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig - } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| One-Way-Inside Hello to be sent by the tester. | |

| ASN.1 PDU Constraint Declaration | |
|---|---------------------|
| Constraint Name | : Hello_V_3_r |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tsp_NID, addr tsp_AESA, o_pg tsp_PGID, r_node tcv_TNID, o_port tsv_R_PID, r_port tsp_TPID, hello_int ?, reserved '0000'O, ig - } </pre> | |
| Detailed Comments : PNNI 1.0 5.14.8 | |
| Two-Way-Inside Hello to be sent by the IUT. | |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : Hello_V_3_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tcv_TNID, addr tsp_TAESA, o_pg tsp_PGID, r_node tsp_NID, o_port tsp_TPID, r_port tsv_R_PID, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Two-Way-Inside Hello to be sent by the tester. |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : Hello_V_4_r |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_HELLO, 100, tsp_NV), flags '0000'O, o_node tsp_NID, addr tsp_AESA, o_pg tsp_PGID, r_node '000'O, o_port tsv_R_PID, r_port '00000000'O, hello_int ?, reserved '0000'O, ig - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Hello Message sent whenever the IUT goes back to the Attempt state. The Version field is equal to tsp_NV, since the Version field in the Hello data structure is equal to zero. |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : Hello_V_4_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Originating Node ID addr tsp_TAESA, -- ATM End System Address o_pg tsp_TPGID, -- Originating Peer Group ID r_node '00'O, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port '00000000'O, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 One-Way-Outside Hello to be sent by the tester. |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : Hello_V_5_r |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tsp_NID, -- Originating Node ID addr tsp_AESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node '00'O, -- Remote Node ID o_port tsv_R_PID, -- Port ID r_port '00000000'O, -- Remote Port ID hello_int ?, -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Hello 1_Way-Inside/Outside by the IUT. |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : Hello_V_6_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 184, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Originating Node ID addr tsp_TAESA, -- ATM End System Address o_pg tsp_TPGID, -- Originating Peer Group ID r_node tsp_NID, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port tsv_R_PID, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig Hello_IGs_V_2_s -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Hello Common Hierarchy to be sent by the tester. |

| ASN.1 PDU Constraint Declaration | |
|---|---|
| Constraint Name | : Hello_V_7_r |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_2_r(PT_HELLO, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tsp_NID, -- Originating Node ID addr tsp_AESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node tcv_TNID, -- Remote Node ID o_port tsv_R_PID, -- Port ID r_port tsp_TPID, -- Remote Port ID hello_int ?, -- Hello Interval reserved '0000'O, -- Reserved ig Hello_IGs_V_2_r -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Two-Way-Outside Hello or Common Outside Hello to be sent by the IUT. It is the first received Outside Hello with Hierarchy list (unknown), Aggregation Token and ULIA. |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : Hello_V_7_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 128, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), flags '0000'O, o_node tcv_TNID, addr tsp_TAESA, o_pg tsp_TPGID, r_node tsp_NID, o_port tsp_TPID, r_port tspv_R_PID, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig Hello_IGs_V_3_s } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Two-Way-Inside Hello to be sent by the tester with empty Nodal Hierarchy List. |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : Hello_V_8_s(version:INTEGER) |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, version), flags '0000'O, o_node tcv_TNID, addr tcv_TAESA, o_pg tsp_PGID, r_node '00'O, o_port tsp_TPID, r_port '00000000'O, hello_int INT_TO_OCT(tsp_THI/1000, 2), reserved '0000'O, ig - } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 One-Way-Inside Hello to be sent by the tester with Version field as a parameter. |

| ASN.1 PDU Constraint Declaration | |
|---|---|
| Constraint Name | : Hello_V_9_s |
| PDU Type | : Hello_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Hello Packet |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_HELLO, 100, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 1) flags '0000'O, -- Flags -Reserved- o_node tcv_TNID, -- Originating Node ID addr tcv_TAESA, -- ATM End System Address o_pg tsp_PGID, -- Originating Peer Group ID r_node tsp_NID, -- Remote Node ID o_port tsp_TPID, -- Port ID r_port tsv_R_PID, -- Remote Port ID hello_int INT_TO_OCT(tsp_THI/1000, 2), -- Hello Interval reserved '0000'O, -- Reserved ig - -- Hello Information Groups } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.8 Two-Way-Inside Hello to be sent by the tester. |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSE_Ack_V_1_r(id,seq,crc:OCTETSTRING) |
| PDU Type | : PTSE_Ack_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Acknowledgement Packets |
| Constraint Value | |
| <pre> { header PnniHeader_V_2_r(PT_PTSE_ACK, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 3) ptse_req_list {NodalPTSEAckIG_V_1_r(id,seq,crc)} -- Nodal PTSE Acknowledgement } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSE_Req_V_1_r |
| PDU Type | : PTSE_Req_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Request Packets sent by the IUT |
| Constraint Value | |
| <pre>{ header PnniHeader_V_2_r(PT_PTSE_REQ, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 5) ptse_req_list ? -- Nodal PTSE Request List }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.12 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSE_Req_V_2_r |
| PDU Type | : PTSE_Req_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Request Packets sent by the IUT |
| Constraint Value | |
| <pre>{ header PnniHeader_V_2_r(PT_PTSE_REQ, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 5) ptse_req_list {NodalPTSERequestListIG_V_1_r} -- Nodal PTSE Request List }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.12 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSP_Fldg_V_1_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet received from the IUT |
| Constraint Value | |
| <pre>{ header PnniHeader_V_1_r(PT_PTSP, 112, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_1_r} -- multiple PTSEs, all from the same originating node }</pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_1_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 112, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_1_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_2_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet received from the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 144, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_2_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_2_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 144, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_2_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_3_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 176, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_3_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_3_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 176, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_3_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_4_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 208, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_4_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_4_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 208, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_4_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_5_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 176, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_5_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_5_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 144, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_5_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_6_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 176, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_6_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_6_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 208, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_6_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_7_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 144, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_7_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_7_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_7_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_8_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 208, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_8_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_8_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_8_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_9_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_9_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_Fldg_V_9_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 84, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_9_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_Fldg_V_10_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_r(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | id, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_Fldg_V_10_r} -- multiple PTSEs, all from the same originating |
| node | |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|---|
| Constraint Name | : PTSP_Fldg_V_10_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 404, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_10_s} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSP_Fldg_V_11_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_11_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : PTSP_Fldg_V_11_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 84, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_11_s} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSP_Fldg_V_12_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 124, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_12_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|---|
| Constraint Name | : PTSP_Fldg_V_12_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 404, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_12_s} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSP_Fldg_V_13_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 84, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_13_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : PTSP_Fldg_V_13_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 96, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_13_s} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSP_Fldg_V_14_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 404, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_14_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|---|
| Constraint Name | : PTSP_Fldg_V_14_s(seq,lt:INTEGER) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 112, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_14_s(seq,lt)} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSP_Fldg_V_15_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 84, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_15_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSP_Fldg_V_16_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 404, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_16_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|---|
| Constraint Name | : PTSP_Fldg_V_17_r(id:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 96, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_17_r} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSP_Fldg_V_18_r(id:OCTETSTRING; seq:INTEGER) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet received from the IUT |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_r(PT_PTSP, 112, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node id, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_18_r(id, seq)} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|---|--|
| Constraint Name | : PTSP_Fldg_V_0_r(oid:OCTETSTRING;type, id, seq:INTEGER) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester (with TTL = 0) |
| Constraint Value | |
| <pre> { header PnniHeader_V_2_r(PT_PTSP, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node oid, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_0_r(type,id,seq)} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSP_Fldg_V_0_s(type, id, seq:INTEGER; crc:OCTETSTRING) |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester (with TTL = 0) |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSP, 64, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_Fldg_V_0_s(type,id,seq,crc)} -- multiple PTSEs, all from the same originating node } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSE_Ack_Fldg_V_1_r(id,seq:INTEGER; crc:OCTETSTRING) |
| PDU Type | : PTSE_Ack_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Acknowledgement Packets |
| Constraint Value | |
| <pre> { header PnniHeader_V_2_r(PT_PTSE_ACK, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 3) ptse_req_list {NodalPTSEAckIG_Fldg_V_1_r(id,seq,crc)} -- Nodal PTSE Acknowledgement } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 PDU Constraint Declaration | |
|--|--|
| Constraint Name | : PTSE_Ack_Fldg_V_1_s(seq:INTEGER; nid, lt, crc:OCTETSTRING) |
| PDU Type | : PTSE_Ack_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Acknowledgement Packets |
| Constraint Value | |
| <pre> { header PnniHeader_V_1_s(PT_PTSE_ACK, 48, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 3) ptse_req_list {NodalPTSEAckIG_Fldg_V_1_s(seq,nid,lt,crc)} -- Nodal PTSE Acknowledgement } </pre> | |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSE_Ack_Fldg_V_2_s(nid, pid, seq, lt, crc:OCTETSTRING) |
| PDU Type | : PTSE_Ack_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PTSE Acknowledgement Packets |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_PTSE_ACK, 48, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 3) ptse_req_list {NodalPTSEAckIG_Fldg_V_2_s(nid, pid, seq,lt,crc)} -- Nodal PTSE Acknowledgement }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.10 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_V_1_r |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_2_r(PT_PTSP, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tsp_NID, -- Originating Node ID o_pg tsp_PGID, -- Originating Peer Group ID ptse_seq {PTSEIG_V_1_r} -- multiple PTSEs, all from the same originating node }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_V_1_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| | <pre>{ header PnniHeader_V_1_s(PT_PTSP, 112, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) o_node tcv_TNID, -- Originating Node ID o_pg tsp_TPGID, -- Originating Peer Group ID ptse_seq {PTSEIG_V_1_s} -- multiple PTSEs, all from the same originating node }</pre> |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_V_2_r |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_2_r(PT_PTSP, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tsp_NID, -- Originating Node ID |
| o_pg | tsp_PGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_2_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_V_2_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 64, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_2_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_V_3_r |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT with PTSE without content and the Remaining Lifetime set to ExpiredAge |
| Constraint Value | |
| { | |
| header | PnniHeader_V_2_r(PT_PTSP, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tsp_NID, -- Originating Node ID |
| o_pg | tsp_PGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_3_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|---|
| Constraint Name | : PTSP_V_3_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 144, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_3_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_V_4_r |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the IUT |
| Constraint Value | |
| { | |
| header | PnniHeader_V_2_r(PT_PTSP, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tsp_NID, -- Originating Node ID |
| o_pg | tsp_PGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_4_r} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 PDU Constraint Declaration | |
|----------------------------------|--|
| Constraint Name | : PTSP_V_4_s |
| PDU Type | : PTSP_T |
| Derivation Path | : |
| Encoding Rule Name | : |
| Encoding Variation | : |
| Comments | : PNNI Topology State Packet sent by the Tester |
| Constraint Value | |
| { | |
| header | PnniHeader_V_1_s(PT_PTSP, 64, Version(tsp_TNV, tsp_TOV, tsp_NV, tsp_OV)), -- PNNI Header (Type = 2) |
| o_node | tcv_TNID, -- Originating Node ID |
| o_pg | tsp_TPGID, -- Originating Peer Group ID |
| ptse_seq | {PTSEIG_V_4_s} -- multiple PTSEs, all from the same originating node |
| } | |
| Detailed Comments | : PNNI 1.0 5.14.9 |

| ASN.1 CM Constraint Declaration | |
|---------------------------------|---|
| Constraint Name | : AddPort_V_1 |
| CM Type | : AddPort_T |
| Derivation Path | : |
| Comments | : Hello_FSM -> Neighb_Peer_FSM A Hello state machine for a link to the neighboring peer has reached 2-WayInside state. Database Synchronizaton is initiated. |
| Constraint Value | |
| { | |
| } | |
| Detailed Comments : | |

| ASN.1 CM Constraint Declaration | |
|---------------------------------|---|
| Constraint Name | : DropPort_V_1 |
| CM Type | : DropPort_T |
| Derivation Path | : |
| Comments | : Hello_FSM -> Neighb_Peer_FSM When a link falls out of the Hello state 2-WayInside, the event DropPort is triggered in the corresponding neighboring peer state machine. When the DropPort event for the last link between the neighboring peers occurs, the neighboring peer state machine will internally generate the DropPortLast event causing all state information for the neighboring peer to be cleared. |
| Constraint Value | |
| { | |
| } | |
| Detailed Comments : | |

| ASN.1 CM Constraint Declaration | |
|---------------------------------|---|
| Constraint Name | : NeighbFullInd_V_1 |
| CM Type | : NeighbFullInd_T |
| Derivation Path | : |
| Comments | : PTC -> MTC PTC informs MTC that the Neighbouring Peer state Full is entered. |
| Constraint Value | |
| { | |
| } | |
| Detailed Comments : | |

| ASN.1 CM Constraint Declaration | |
|--|---|
| Constraint Name | : TerminateReq_V_1(Cause:Cause_T) |
| CM Type | : TerminateReq_T |
| Derivation Path | : |
| Comments | : MTC -> PTC Request by the MTC that the PTC indicated by the MCP terminate its execution. |
| Constraint Value | |
| <pre>{ cause Cause -- Cause for Termination Request }</pre> | |
| Detailed Comments : | |

| ASN.1 CM Constraint Declaration | |
|--|--|
| Constraint Name | : TestBodyStartReq_V_1 |
| CM Type | : TestBodyStartReq_T |
| Derivation Path | : |
| Comments | : MTC -> PTC Request by the MTC that the PTC start with execution of the test body. |
| Constraint Value | |
| <pre>{ }</pre> | |
| Detailed Comments : | |

IV

Dynamic Part

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V001 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that the node includes the newest and oldest version supported fields in all packets. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V001) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.1 PICS 3.10.2 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V002 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that all versions in the range advertised are supported by the advertiser. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V002) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.1 PICS 3.10.3 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V003 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that the Inactivity timer is set to the value, InactivityFactor times the HelloInterval from the most recent Hello received from the neighbor. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.5 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V003) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.1 PICS 3.10.8 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V004 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that the Hello timer is restarted after an event-triggered Hello is transmitted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.8 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V004) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.68 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V005 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that if a Hello has a top level unknown TLV with the mandatory tag bit set, that the Hello packet is discarded. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.9 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V005) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.3 PICS 3.10.82 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V006 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that if the Hello interval in the Hello packet is set to zero, that the Hello packet is discarded. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.10 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V006) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.3 PICS 3.10.83 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V007 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that if the port ID in the Hello packet is set to zero, that the Hello packet is discarded. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.11 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V007) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.3 PICS 3.10.83 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V008_1 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that when in state One Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.12.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V008_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.62 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V008_2 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that when in state Two Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.12.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V008_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.62 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V008_3 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that when in state Attempt, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.12.4 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V008_3) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.62 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_1 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a hello is sent upon state change from Attempt to One Way Inside subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_2 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Attempt to Two Way Inside subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_3 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from One Way Inside to Attempt subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.4 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_3) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_4 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Two Way Inside to Attempt subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.5 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_4) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_5 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is not sent upon state change from One Way Inside to Two Way Inside. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.7 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_5) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V009_6 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Down to Attempt subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.9 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V009_6) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V010_1 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that when in state One Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.16.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V010_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.70 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V010_2 | | | | | |
| Group : Hello/SS_M/GENERAL/ | | | | | |
| Purpose : To verify that when in state Two Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.16.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V010_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.70 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V101 | | | | | |
| Group : Hello/SS_M/DOWN/ | | | | | |
| Purpose : To verify that while in the Down state and a Link Up event is generated, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hd.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V101) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp1 PICS 3.10.10 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V201_1 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 1-Way Inside Received event is generated, sends a Hello, and enters the 1-Way Inside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V201_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp2 PICS 3.10.13 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V201_2 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 1-Way Inside Received event is generated, that the IUT starts the Inactivity timer, sends a Hello and restarts the Hello Timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.1.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V201_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp2 PICS 3.10.13 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V202_1 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 2-Way Inside Received event is generated that a Hello is sent, and 2-Way Inside state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.2.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V202_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp3 PICS 3.10.16 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V202_2 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 2-Way Inside Received event is generated that the Inactivity timer is restarted, a hello is sent, the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.2.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V202_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp3 PICS 3.10.16 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V203 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a Two Way Outside Received event is generated, the IUT does nothing. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V203) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Note 1 PICS 3.10.26 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V204 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a Common Hierarchy Received event are generated that the IUT does nothing. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.4 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V204) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Note 1 PICS 3.10.30 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V205 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a Hello Mismatch Received event is generated, the IUT does nothing. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.5 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V205) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp0 PICS 3.10.35 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V206 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that when in the Attempt state, that the Hellos have their remote node ID and remote port ID set to zero. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ha.8 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V206) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.69 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V207 | | | | | |
| Group : Hello/SS_M/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT does nothing. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hao.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V207) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Note 1 PICS 3.10.21 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V301 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V301) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.14 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V302_1 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that 2-Way Inside state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.2.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V302_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp4 PICS 3.10.17 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V302_2 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.2.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V302_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp4 PICS 3.10.17 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V303_1 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.3.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V303_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.36 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V303_2 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.3.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V303_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.36 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V304_1 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.5.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V304_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.48 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V304_2 | | | | | |
| Group : Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi1.5.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V304_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.48 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V401_1 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, a Hello is sent, and the 1-Way Inside state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V401_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp10 PICS 3.10.15 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V401_2 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted, a Hello is sent, that Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.1.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V401_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp10 PICS 3.10.15 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V402 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and a 2-Way Inside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V402) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.18 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V403_1 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, a Hello is sent, and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.3.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V403_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp16 PICS 3.10.37 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V403_2 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.3.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V403_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp16 PICS 3.10.37 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V404_1 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.5.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V404_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp16 PICS 3.10.49 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V404_2 | | | | | |
| Group : Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hi2.5.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V404_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp16 PICS 3.10.49 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V501_1 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that when in state One Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.12.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V501_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.62 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V501_2 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that when in state Two Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.12.5 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V501_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.62 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V502_1 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Attempt to One Way Outside subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V502_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V502_2 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Attempt to Two Way Outside subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.6 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V502_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V502_3 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from One Way Outside to Attempt subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.6 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V502_3) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V502_4 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is sent upon state change from Two Way Outside to Attempt subject to the HoldDown timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V502_4) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V502_5 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that a Hello is not sent upon state change from One Way Outside to Two Way Outside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.13.8.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V502_5) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.63 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V503 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that when multiple event triggered Hellos are deferred because of the HoldDown timer, that the IUT sends only one Hello which contains the most current information for all IGs when the HoldDown timer expires. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.14 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V503) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.67 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V504_1 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that when in state One Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.16.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V504_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.70 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V504_2 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that when in state Two Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 H.16.3 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V504_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.70 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V505 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that the sequence number of the first instance of the nodal hierarchy list sent to any neighbor is greater than zero. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hb.7 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V505) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.74 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V506 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that if no higher level is known, that an empty nodal hierarchy list is included in the Hello. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hb.9 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V506) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2 PICS 3.10.76 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V507_1 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that the ULIA information group is included in all Hellos while in the states: 1-Way Outside. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hb.10.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V507_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2, 5.6.2.2.1 PICS 3.10.77 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V507_2 | | | | | |
| Group : Hello/SS_B/GENERAL/ | | | | | |
| Purpose : To verify that the ULIA information group is included in all Hellos while in the states: 2-Way Outside. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hb.10.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V507_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.2, 5.6.2.2.1 PICS 3.10.77 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V601_1 | | | | | |
| Group : Hello/SS_B/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hao.1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V601_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp5 PICS 3.10.20 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V601_2 | | | | | |
| Group : Hello/SS_B/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hao.1.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V601_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp5 PICS 3.10.20 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V602_1 | | | | | |
| Group : Hello/SS_B/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 2-Way Outside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hao.3.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V602_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp5 PICS 3.10.25 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V602_2 | | | | | |
| Group : Hello/SS_B/ATTEMPT/ | | | | | |
| Purpose : To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Hao.3.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V602_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp5 PICS 3.10.25 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V701 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and a 1-Way outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V701) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.22 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V702_1 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the IUT enters the 2-Way Outside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.2.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V702_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.27 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V702_2 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.2.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V702_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.27 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V703_1 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.5.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V703_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.38 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V703_2 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.5.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V703_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.38 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V704_1 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.6.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V704_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.50 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V704_2 | | | | | |
| Group : Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho1.6.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V704_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.50 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V801_1 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.1.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V801_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp13 PICS 3.10.23 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V801_2 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.1.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V801_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp13 PICS 3.10.23 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V802 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V802) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp12 PICS 3.10.28 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V803_1 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.4.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V803_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.39 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V803_2 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.4.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V803_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.39 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V804_1 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.6.1 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V804_1) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.51 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-----------------|---------|----------|
| Test Case Name : Hello_V804_2 | | | | | |
| Group : Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Purpose : To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. | | | | | |
| Configuration : tcc_Hello | | | | | |
| Default : Default_Hello_MTC | | | | | |
| Comments : ATM Forum/97-0691 Ho2.6.2 | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM:Hello_FSM_V804_2) | | | |
| 2 | | ?DONE(Hello_FSM) | | R | |
| Detailed Comments : PNNI 1.0 5.6.2.1.4 Table 5-10 Hp8 PICS 3.10.51 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V001_M | | | | | |
| Group : DBSynchronization/NPDown/ | | | | | |
| Purpose : To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V001_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7 PICS 3.14.8 atm98-0466: DS.10 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V001_S | | | | | |
| Group : DBSynchronization/NPDown/ | | | | | |
| Purpose : To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V001_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7 PICS 3.14.8 atm98-0466: DS.10 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V101_M | | | | | |
| Group : DBSynchronization/Negotiating/ | | | | | |
| Purpose : To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V101_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.52 Applies to Master and Slave role of the IUT atm98-0466: DS.54 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V101_S Group : DBSynchronization/Negotiating/ Purpose : To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V101_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.52 Applies to Master and Slave role of the IUT atm98-0466: DS.54 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V102_M Group : DBSynchronization/Negotiating/ Purpose : To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V102_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.54 Applies to Master and Slave role of the IUT atm98-0466: DS.56 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V102_S Group : DBSynchronization/Negotiating/ Purpose : To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V102_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.54 Applies to Master and Slave role of the IUT atm98-0466: DS.56 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V103_M Group : DBSynchronization/Negotiating/ Purpose : To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V103_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.53 Applies to Master and Slave role of the IUT atm98-0466: DS.55 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V103_S Group : DBSynchronization/Negotiating/ Purpose : To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V103_MS) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.53 Applies to Master and Slave role of the IUT atm98-0466: DS.55 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V104_M Group : DBSynchronization/Negotiating/ Purpose : To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Master in the database synchronization. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V104_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.4 Table 5-12 Ds2 PICS 3.14.30 atm98-0466: DS.32a Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V105_S | | | | | |
| Group : DBSynchronization/Negotiating/ | | | | | |
| Purpose : To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Slave in the database synchronization. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V105_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.4 Table 5-12 Ds2 PICS 3.14.30 atm98-0466: DS.32b Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V201_M | | | | | |
| Group : DBSynchronization/Exchanging/ | | | | | |
| Purpose : To verify that if Master, the DSRxmt Timer is restarted when the node receives a correct Database Summary packet. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V201_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.1 PICS 3.14.17 atm98-0466: DS.19 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V202_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Master, that Database Summary packets are sent when the Slave acknowledges the previous Database Summary packet and it has DS packets to send. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V202_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.55 atm98-0466: DS.57 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V203_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Master and this packet includes the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to zero. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V203_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.56 atm98-0466: DS.58 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V204_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Master and all of the database summary has already been sent to the Slave, that the More (M) bit in the Database Summary packet is set to zero and the contents are empty. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V204_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.57 atm98-0466: DS.59 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V205_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Master and this packet does not include the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to one. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V205_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.58 atm98-0466: DS.60 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V206_S Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Slave, that Database Summary packets are sent only in response to Database Summary packets received. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V206_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.59 atm98-0466: DS.61 Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V207_S Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Slave and all of the database summary has already been previously sent to the Master, that the More (M) bit in the Database Summary packet is set to zero. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V207_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.60 atm98-0466: DS.62 Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V208_S Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging when the node is Slave and this packet contains at least one item of the database summary to be sent to the Master, that the more (M) bit is set to one. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V208_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.61 atm98-0466: DS.63 Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V209_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is not empty, that the DS Rxmt Timer is stopped and (thus) no more DS packets are sent. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V209_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.86 atm98-0466: DS.88 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V210_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the DS Rxmt Timer is stopped, (thus) no more DS packets are sent. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V210_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.87 atm98-0466: DS.89a Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V211_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has not sent its entire database that a new Database Summary packet is sent and the DS Rxmt Timer is restarted. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V211_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.88 atm98-0466: DS.90 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V212_M Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and the node is Master and a duplicate Database Summary packet is received, that the processing of this packet is stopped. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V212_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.89 atm98-0466: DS.91 Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V213_S Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and this node is Slave and the packet's DS sequence number is one more than this node's DS sequence number, that the packet is accepted. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V213_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.90 atm98-0466: DS.92 Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V214_S Group : DBSynchronization/Exchanging/ Purpose : To verify that while in Exchanging and this node is Slave and a duplicate Database Summary packet is received, that the last Database Summary packet sent to the Master is retransmitted. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V214_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.5 PICS 3.14.95 atm98-0466: DS.97a Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V215_M Group : DBSynchronization/Exchanging/ Purpose : To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V215_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.97 atm98-0466: DS.99 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V215_S Group : DBSynchronization/Exchanging/ Purpose : To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V215_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.97 atm98-0466: DS.99 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V216_M Group : DBSynchronization/Exchanging/ Purpose : To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V216_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.98 atm98-0466: DS.100 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V216_S | | | | | |
| Group : DBSynchronization/Exchanging/ | | | | | |
| Purpose : To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V216_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.98 atm98-0466: DS.100 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V301_M | | | | | |
| Group : DBSynchronization>Loading/ | | | | | |
| Purpose : To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. | | | | | |
| Configuration : tcc_DBSync | | | | | |
| Default : Default_DBSync_MTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V301_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.100 atm98-0466: DS.102b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V301_S Group : DBSynchronization/Loading/ Purpose : To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V301_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.100 atm98-0466: DS.102b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_1_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_1_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_2_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_2_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_3_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_3_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V302_3_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V302_3_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_1_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_1_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_2_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_2_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_3_M Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_3_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V303_3_S Group : DBSynchronization/Loading/ Purpose : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V303_3_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_1_M Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_1_S Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_2_M Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_2_S Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105b Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_3_M Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_3_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V304_3_S Group : DBSynchronization/Loading/ Purpose : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V304_3_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V305_M Group : DBSynchronization/Loading/ Purpose : To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V305_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.7 PICS 3.14.108 atm98-0466: DS.110 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V305_S Group : DBSynchronization/Loading/ Purpose : To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V305_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.7 PICS 3.14.108 atm98-0466: DS.110 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V306_M Group : DBSynchronization/Loading/ Purpose : To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V306_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.7 PICS 3.14.109 atm98-0466: DS.111 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V306_S Group : DBSynchronization/Loading/ Purpose : To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V306_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.7 PICS 3.14.109 atm98-0466: DS.111 Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V401_M Group : DBSynchronization/Full/ Purpose : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the link is advertised in a PTSE. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V401_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.87 atm98-0466: DS.89b Applies to Master role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V401_S Group : DBSynchronization/Full/ Purpose : To verify that while in Exchanging and the node is Slave, if a packet is received that has the DS sequence number one more than this node's own DS sequence number, the More bit set to zero and the just transmitted Database Summary packet had the M bit is set to zero and the PTSE Request List is empty, that the link is advertised in a PTSE. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V401_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.94 atm98-0466: DS.96 Applies to Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V402_1_M Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V402_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V402_1_S Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V402_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V402_2_M Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V402_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V402_2_S Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V402_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V403_1_M Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V403_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V403_1_S Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V403_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V403_2_M Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V403_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V403_2_S Group : DBSynchronization/Full/ Purpose : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V403_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V404_1_M Group : DBSynchronization/Full/ Purpose : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V404_1_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V404_1_S Group : DBSynchronization/Full/ Purpose : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V404_1_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108a Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V404_2_M Group : DBSynchronization/Full/ Purpose : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_TLNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V404_2_M) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Case Name : DBSync_V404_2_S Group : DBSynchronization/Full/ Purpose : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Configuration : tcc_DBSync Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:Hello_FSM_Emull (tsp_THNID)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:DBSync_FSM_V404_2_S) | | | (2) |
| 3 | | +PostambleDBSync_PTCs | | | |
| Detailed Comments : (1) Hello FSM Emulation (2) The test body PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108d Applies to Master and Slave role of the IUT | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V001 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node: - ATM End System address - leadership priority - nodal information flags - preferred peer group leader node ID. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V001_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V001_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.2, Table 5-35 PICS 3.15.1 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V002 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V002_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V002_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.2, 5.8.5.2.5.6 PICS 3.15.3 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V003</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V003_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V003_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.1.3.3, 5.8.5.2.5.5 PICS 3.15.4</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V004 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V004_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V004_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.4, 5.8.5.2.5.1 PICS 3.15.5 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V005</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 (Cell Loss Ratio for CLP=0) is present for CBR, Real Time VBR and Non-Real Time VBR service categories.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V005_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V005_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.2 PICS 3.15.6</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V006</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V006_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V006_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.3 PICS 3.15.7</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V007 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V007_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V007_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.7, 5.8.5.2.5.7 PICS 3.15.8 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V008 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V008_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V008_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.8 PICS 3.15.9 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V009 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V009_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V009_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V010</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V010_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V010_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V011 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V011_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V011_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|---|-------------------|---------|----------|
| Test Case Name : Fldg_V012 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V012_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V012_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V013</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included:</p> <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V013_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V013_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.3.1 PICS 3.15.14</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|--|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V014</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V014_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V014_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.3.1 PICS 3.15.14</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V015</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included:</p> <ul style="list-style-type: none"> - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V015_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V015_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.3.2 PICS 3.15.17</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|---|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V016</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA, Hello_CP_1, Hello_Neighb_Peer_CP_1, Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V016_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND, Hello_CP_2, Hello_Neighb_Peer_CP_2, Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V016_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.1.3.2 PICS 3.15.17</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V017 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V017_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V017_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V018 Group : Flooding/ Purpose : To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V018_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V018_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.2.2.4 PICS 3.15.22 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V019 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V019_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V019_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.2.2.4 PICS 3.15.22 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V020 Group : Flooding/ Purpose : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V020_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V020_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.3.2 PICS 3.15.31 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V021 Group : Flooding/ Purpose : To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V021_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V021_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.3.2 PICS 3.15.33 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|--|-------|--|----------------------|---------|----------|
| <p>Test Case Name : Fldg_V022</p> <p>Group : Flooding/</p> <p>Purpose : To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node.</p> <p>Configuration : tcc_Flooding</p> <p>Default : DefTwoLinks_Fldg_MTC_one</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V022_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V022_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | Neighb_Peer_CP_2?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 8 | | Neighb_Peer_CP_1!TestBodyStartReq_T | TestBodyStartReq_V_1 | | (6) |
| 9 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link (6) Send a request that the Neighbouring Peer FSM for the first link starts the test body execution.</p> <p>PNNI 1.0 5.8.3.3 PICS 3.15.39</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V023 Group : Flooding/ Purpose : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V023_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V023_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.3.4 PICS 3.15.41 | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V024 Group : Flooding/ Purpose : To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V024_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V024_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.3.3, 3b | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| <p>Test Case Name : Fldg_V025 Group : Flooding/ Purpose : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V025_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V025_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| <p>Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link</p> <p>PNNI 1.0 5.8.3.3, 4a</p> | | | | | |

| Test Case Dynamic Behaviour | | | | | |
|---|-------|--|-------------------|---------|----------|
| Test Case Name : Fldg_V026 Group : Flooding/ Purpose : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. Configuration : tcc_Flooding Default : DefTwoLinks_Fldg_MTC_one Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | CREATE(Hello_FSM_1:TwoLinks_Hello_FSM_Emul(tsp_TLNID,tsp_TAESA,Hello_CP_1,Hello_Neighb_Peer_CP_1,Hello_PCO_1)) | | | (1) |
| 2 | | CREATE(Neighb_Peer_FSM_1:Fldg_FSM_V026_First) | | | (2) |
| 3 | | Neighb_Peer_CP_1?NeighbFullInd_T | NeighbFullInd_V_1 | | (3) |
| 4 | | CREATE(Hello_FSM_2:TwoLinks_Hello_FSM_Emul(tsp_THNID,TAESA_2ND,Hello_CP_2,Hello_Neighb_Peer_CP_2,Hello_PCO_2)) | | | (4) |
| 5 | | CREATE(Neighb_Peer_FSM_2:Fldg_FSM_V026_Second) | | | (5) |
| 6 | | ACTIVATE(DefTwoLinks_Fldg_MTC_two) | | | |
| 7 | | +PostTwoLinks_Fldg | | | |
| Detailed Comments : (1) PreNegotiation and Hello FSM Emulation for the first link (2) The test body wrt Neighbouring Peer FSM for the first link (3) Received an indication that the first link has been established (entered in the Neighbouring Peer state Full) (4) PreNegotiation and Hello FSM Emulation for the second link (5) The test body wrt Neighbouring Peer FSM for the second link PNNI 1.0 5.8.3.3, 4b | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V001 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that the node includes the newest and oldest version supported fields in all packets. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| 4 | | [(tsp_NV>0) AND (tsp_OV>0)] | | R | (3) |
| 5 | | [NOT((tsp_NV>0) AND (tsp_OV>0))] | | F | |
| Detailed Comments : (1) Starts the IUTs Hello-FSM. (2) Receive the first Hello. (3) Check that the received Newest and Oldest versions (tsp_NV, tsp_OV) are not zero; then the test is considered successful since the first received packet is the only time when these two fields are really used. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---------------------|---------|----------|
| Test Step Name : Hello_FSM_V002 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that all versions in the range advertised are supported by the advertiser. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int, tcv_CV := tsp_OV) CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| 4 | L1 | Hello_PCO_1!Hello_T | Hello_V_1_s(tcv_CV) | | (3) |
| 5 | | START T_Resp | | | |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_2_r(tcv_CV) | (P) | (4) |
| 7 | | [tcv_CV <> tsp_NV] | | | |
| 8 | | Hello_PCO_1!Hello_T | Hello_I_1_s | | (5) |
| 9 | | START T_Resp | | | |
| 10 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | | (6) |
| 11 | | START T_Resp | | | |
| 12 | | Hello_PCO_1?Hello_T (tcv_CV := tcv_CV +1) CANCEL T_Resp | Hello_V_5_r | | (7) |
| 13 | | GOTO L1 | | | |
| 14 | | [tcv_CV = tsp_NV] | | (P) | |
| 15 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Starts the IUTs Hello-FSM (2) Receive the first Hello, and store in the local variables tsv_R_PID and tsv_R_HI the values of the port ID and Hello Interval fields respectively, from the received Hello. (3) Send a Hello with the Newest Version equal to the Oldest Version announced by the IUT. (4) Receive a Hello in response, where the Version field is equal to the advertised Newest version by the Tester, i.e. IUTs oldest version. (5) Bring the IUT back to Attempt sending a Hello with a mismatch in the node ID field. (6) Receive a Hello with the remote node ID and remote Port ID equal to zero and with the Version field equal to the Newest version field. (7) Repeat an equivalent sequence of event for checking each one of the sequences in the range, unless the version that has been checked is the last version of the range, then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V003 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that the Inactivity timer is set to the value, InactivityFactor times the HelloInterval from the most recent Hello received from the neighbor. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| 4 | | Hello_PCO_1!Hello_T | Hello_V_2_s | | (3) |
| 5 | | START T_Inact | | | |
| 6 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | (P) | (4) |
| 7 | | GOTO L1 | | | |
| 8 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | (P) | (5) |
| 9 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (6) |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | (6) |
| 12 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Starts the IUT Hello-FSM (2) Receive the first Hello, and store in the local variables tsv_R_PID and tsv_R_HI the values of the port ID and Hello Interval fields respectively, from the received Hello. (3) Send a Hello and start the Inact timer. (4) Receive Hellos where the remote Node ID and remote Port ID fields are not zero. (5) Receive a Hello with the remote NodeID, remote PortID equal to zero and with the Version field equal to the Newest Version. The Inact timer is stopped (6) Is checked that the Inact timer value is equal to InactivityFactor times the HelloInterval plus or minus the fractional variance of 25%. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V004 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that the Hello timer is restarted after an event-triggered Hello is transmitted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_2_s | | (2) |
| 3 | | START T_Hello | | | |
| 4 | | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | (P) | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt And the local variables tsv_R_PID and tsv_R_HI are already set in the Tester. (2) A Hello is sent to the IUT where the remote node ID and the remote Port ID fields are zero. (3) Receive a Hello where the remote NodeID field and remote PortID fields are set to the Tester's NodeID and PortID. The Hello timer is started (4) Receive a Hello where the remote NodeID and remote PortID fields are not zero. The Hello timer is stopped and its value should match with the tsv_R_HI plus or minus the fractional variance of 25%. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V005 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that if a Hello has a top level unknown TLV with the mandatory tag bit set, that the Hello packet is discarded. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_I_2_s | | (2) |
| 3 | | START T_Hello | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | (P) | (3) |
| 5 | | + PostambleHelloAttempt | | | |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_3_r | (F) | (4) |
| 7 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt And the local variables tsv_R_PID and tsv_R_HI are already set in the Tester. (2) Send a Hello with the remote NodeID and remote PortID fields equal to zero, and including an IG with the mandatory flag set to one. (3) Receive a Hello with the remote NodeID and remote PortID fields set to zero, then it means the IUT discarded the last packet it received. (4) If the IUT sends a hello where the remote node ID and remote port ID are equal to the Tester's node ID and port ID respectively, then the Test fails. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V006 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that if the hello interval in the Hello packet is set to zero, that the Hello packet is discarded. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_I_3_s | | (2) |
| 3 | | START T_Hello | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | (P) | (3) |
| 5 | | + PostambleHelloAttempt | | | |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_3_r | (F) | (4) |
| 7 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) The preamble gets the IUT to the state ATTEMPT. All the information from the first Hello received from the IUT has been stored in variables. (2) Send a Hello with the remote NodeID and remote PortID fields set to zero. Also the HelloInterval is set to zero. (one way inside Hello) (3) Receive a Hello where the remote NodeID and remote PortID fields set to zero. (4) If a Hello where the remote node ID and remote port ID fields are equal to the Tester's node ID and port ID respectively, that would mean that the IUT did not discard the last Hello sent by the Tester. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V007 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that if the port ID in the Hello packet is set to zero, that the Hello packet is discarded. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_I_4_s | | (2) |
| 3 | | START T_Hello | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | (P) | (3) |
| 5 | | + PostambleHelloAttempt | | | |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_3_r | (F) | (4) |
| 7 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) The preamble gets the IUT to the state ATTEMPT. All the information from the first Hello received from the IUT has been stored in variables. | | | | | |
| (2) Send a Hello with the remote NodeID and remote PortID fields set to zero, and also the Port ID field is set to zero. | | | | | |
| (3) Receive a Hello where the remote NodeID and remote PortID fields set to zero. | | | | | |
| (4) If a Hello where the remote node ID and remote port ID fields are equal to the Tester's node ID and port ID respectively, that would mean that the IUT did not discard the last Hello sent by the Tester. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V008_1 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that when in state One Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_2_s | | (2) |
| 3 | | START T_Period | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | | (3) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Period | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | ?TIMEOUT T_Period | | | (4) |
| 12 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a hello with the remote NodeID and remote PortID fields set to zero and the Peer Group ID matching the IUT's one. Start the Period and Hello timers. (one way inside) (3) Receive a hello with the remote NodeID and remote PortID fields equal to IUT's NodeID and PortID. And check that the value of the Hello timer is around tsv_R_HI plus or minus the fractional variance of 25%. Restart the Hello timer. (4) The Period is an interval timer. If hellos are received every tsv_R_HI time until the Period timer expires, then it is considered that this action happens periodically in the state One Way Inside. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V008_2 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that when in state Two Way Inside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_3_s | | (2) |
| 3 | | START T_Period | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | | (3) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Period | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | ?TIMEOUT T_Period | | | (4) |
| 12 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside. (2) Send a Hello with the remote NodeID and remote PortID fields equal to the IUT's Node ID and Port ID respectively. And the Peer Group ID matching the IUT's one. Start the Period and Hello timers. (3) Receive a Hello with the remote NodeID and remote PortID fields equal to IUT's NodeID and PortID. And check that the value of the Hello timer is around tsv_R_HI plus or minus the fractional variance of 25%. Restart the Hello timer. (4) The Period timer is an interval timer. If Hellos are received every tsv_R_HI time until the Period timer expires, then it is considered that this action happens periodically in the state Two Way Inside | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V008_3 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that when in state Attempt, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_5_r | | (2) |
| 4 | | START T_Period | | | |
| 5 | L1 | START T_Hello | | | |
| 6 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (3) |
| 7 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 8 | | GOTO L1 | | | |
| 9 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 10 | | +PostambleHelloAttempt | | | |
| 11 | | ?TIMEOUT T_Period | | | (4) |
| 12 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Receive a Hello with the remote NodeID and remote PortID fields set to zero. Start the Period and Hello timers. (3) Receive a Hello with the remote NodeID and remote PortID fields still set to zero. Check that the value of the Hello timer is around tsv_R_HI. Restart the Hello timer. (4) The Period timer is an interval timer. If Hellos are received every tsv_R_HI time until the Period timer expires, then it is considered that this action happens periodically in the state Attempt. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_1 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that a hello is sent upon state change from Attempt to One Way Inside subject to the HoldDown timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_5_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_2_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | (5) |
| 7 | | +CheckStateHelloOWI | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Receive a Hello with the remote NodeID and remote PortID fields set to zero. Start the Hold timer. (3) Send a hello with the remote node ID and remote port ID fields equal to zero and the peer group ID equal to the iut's peer group ID. Start the T_Resp timer. (One Way Inside). (4) No-message should be receive before the hold timer expires. (5) Receive in response a hello with the remote node id field and remote port id field equal to the tester's node id an port id respectively. otherwise if the resp timer expires than the test fails. (6) Check that the iut is in the state one way inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_2 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that a Hello is sent upon state change from Attempt to Two Way Inside subject to the HoldDown timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_5_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_3_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | (5) |
| 7 | | +CheckStateHelloTWI | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Receive a Hello with the remote NodeID and remote PortID fields set to zero. Start the Hold timer. (3) Send a Hello with the remote node ID and remote port ID fields equal the iut's node id and port id and the peer group ID equal to the iut's peer group ID. Start the resp timer. (two way inside). (4) No-message should be receive before the hold timer expires. (5) Receive in response a hello with the remote node id field and remote port id field equal to the tester's node id an port id respectively. (6) Check that the iut is in the state two way inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_3 Group : Bodies/Hello/SS_M/GENERAL/ Objective : To verify that a Hello is sent upon state change from One Way Inside to Attempt subject to the HoldDown timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_3_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (5) |
| 7 | | +CheckStateHelloAttempt | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state one way inside. (2) Receive a Hello with the remote NodeID and remote PortID fields equal to the testers node id and port id. Start the Hold timer. (3) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And the tester's node id with a wrong value (Hello mismatch). (4) No-message should be receive before the hold timer expires. (5) Receive in response a Hello with the remote node id field and remote port id field equal 0. And version field equal to the newest version. Otherwise, if the resp timer expires first -> test fails. (6) Check that the iut is in the state attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_4 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that a Hello is sent upon state change from Two Way Inside to Attempt subject to the HoldDown timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_3_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (5) |
| 7 | | +CheckStateHelloAttempt | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state two way inside. (2) Receive a Hello with the remote NodeID and remote PortID fields equal to the testers node id and port id. Start the Hold timer. (3) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And the tester's node id with a wrong value (Hello mismatch). (4) No-message should be receive before the hold timer expires. (5) Receive in response a Hello with the remote node id field and remote port id field equal 0 and version field equal to the newest version. Otherwise, if the resp timer expires first -> test fails. (6) Check that the iut is in the state attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_5 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that a Hello is not sent upon state change from One Way Inside to Two Way Inside. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_3_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | +CheckStateHelloTWI | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state one way inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And in the peer group id field it has the iut's pgid. (2 way inside hello). (3) T_NoResp timer expires, this means that no Hello was received in response to the last Hello sent by the tester. (4) Check that the IUT is in the state 2 way inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V009_6 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that a Hello is sent upon state change from Down to Attempt subject to the HoldDown timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| 4 | | +CheckStateHelloAttempt | | | (3) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Starts the IUT's Hello FSM (2) Receive a Hello with the remote Node ID field and remote Port ID field equal to zero, before the Resp timer fires, otherwise the test fails. (3) Check that the IUT is in the state ATTEMPT at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V010_1 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that when in state One Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_3_r | (P) | (2) |
| 4 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside (2) Receive a Hello with the remote node ID and remote Port ID fields equal to the Tester's Node ID and Port ID respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V010_2 | | | | | |
| Group : Bodies/Hello/SS_M/GENERAL/ | | | | | |
| Objective : To verify that when in state Two Way Inside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_3_r | (P) | (2) |
| 4 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside (2) Receive a Hello with the remote node ID and remote Port ID fields equal to the Tester's Node ID and Port ID respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V101 | | | | | |
| Group : Bodies/Hello/SS_M/DOWN/ | | | | | |
| Objective : To verify that while in the Down state and a Link Up event is generated, that a Hello is sent and the Attempt state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| 4 | | +CheckStateHelloAttempt | | | (3) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) A link_Up event is received by the IUT (2) Receive a Hello with the remote NodeID field and the remote PortID field equal to zero. (3) Check that the IUT is in the state ATTEMPT. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V201_1 | | | | | |
| Group : Bodies/Hello/SS_M/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a 1-Way Inside Received event is generated, sends a Hello, and enters the 1-Way Inside state. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_2_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | (3) |
| 4 | | +CheckStateHelloOWI | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Attempt. | | | | | |
| (2) Send a Hello with the remote NodeID and remote PortID fields equal to zero, and the Peer Group ID matches with the IUT's one. Start Resp timer. | | | | | |
| (3) Receive a Hello where the remote NodeID and remote PortID fields has the value of the previous advertised Tester's NodeID and PortID. Otherwise if the Resp timer expires the test fails. | | | | | |
| (4) Check that the IUT is in the state One Way Inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V201_2 Group : Bodies/Hello/SS_M/ATTEMPT/ Objective : To verify that while in the Attempt state and a 1-Way Inside Received event is generated, that the IUT starts the Inactivity timer, sends a Hello and restarts the Hello Timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp, START T_Inact | Hello_V_2_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Inact | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Hello, CANCEL T_Inact | Hello_V_4_r | | (5) |
| 12 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (6) |
| 13 | | + PostambleHelloAttempt | | | |
| 14 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | (6) |
| 15 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote NodeID and remote PortID fields equal to zero, and the Peer Group ID matches with the IUT's one. Start the Inact and Resp timers (One Way Inside Hello). (3) Receive a Hello where the remote NodeID and remote PortID fields has the value of the previous advertised Tester's NodeID and PortID. Otherwise if the Resp timer expires the test fails. (4) Receive some hellos with the remote NodeID and remote PortID fields has the value of the IUT's NodeID and PortID. And check that they are | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|--|--|
| <p>Detailed Comments : ...</p> <p>received in intervals of tsv_R_HI time.</p> <p>(5) Receive a Hello with the remote NodeID and remote PortID fields equal to zero and the Peer Group ID matches with the Tester's one. And the Version field equal to the newest version. Stop the Inact timer.</p> <p>(6) Check if the value of the T_Inact timer is Inactivity Factor times HelloInterval.</p> | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| <p>Test Step Name : Hello_FSM_V202_1</p> <p>Group : Bodies/Hello/SS_M/ATTEMPT/</p> <p>Objective : To verify that while in the Attempt state and a 2-Way Inside Received event is generated that a Hello is sent, and 2-Way Inside state is entered.</p> <p>Default : Default_Hello_PTC</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_3_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | (3) |
| 4 | | + CheckStateHelloTWI | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| <p>Detailed Comments : (1) Bring the IUT to the state Attempt.</p> <p>(2) Send a Hello with the receive a Hello with the remote NodeID, remote PortID and Peer Group ID equal to the IUT's NodeID, PortID and Peer Group ID. Start the T_Resp timer. (Two Way Inside Hello)</p> <p>(3) Receive in response a Hello with remote NodeID, remote PortID equal to the Tester's NodeID, PortID and the Peer Group ID matches with the Tester's one.</p> <p>(4) Check that the IUT is in the state Two Way Inside</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V202_2 Group : Bodies/Hello/SS_M/ATTEMPT/ Objective : To verify that while in the Attempt state and a 2-Way Inside Received event is generated that the Inactivity timer is restarted, a hello is sent, the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp, START T_Inact | Hello_V_3_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Inact | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Hello, CANCEL T_Inact | Hello_V_4_r | | (5) |
| 12 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (6) |
| 13 | | + PostambleHelloAttempt | | | |
| 14 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | (6) |
| 15 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote NodeID, remote PortID and Peer Group ID equal to the IUT's NodeID, PortID and Peer Group ID. Start the Resp timer. Start the Inact timer. (Two Way Inside Hello) (3) Receive a Hello with remote NodeID, remote PortID equal to the Tester's NodeID, PortID and the Peer Group ID matches with the Tester's one. Start the Hello timer. (4) Receive a Hello equal to the last received hello from the IUT. Stop the Hello timer and check that its value is r_HI, otherwise the test | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|---|--|
| <p>Detailed Comments : ...</p> <p>fails.</p> <p>(5) Receive a Hello with the remote NodeID, remote PortID equal to zero and the Peer Group ID matches with the Tester's Peer Group ID. And the Version field is equal to zero. Stop Inact timer.</p> <p>(6) Check the Inact timer value against InactivityFactor times HelloIntervals.</p> | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| <p>Test Step Name : Hello_FSM_V203</p> <p>Group : Bodies/Hello/SS_M/ATTEMPT/</p> <p>Objective : To verify that while in the Attempt state and a Two Way Outside Received event is generated, the IUT does nothing.</p> <p>Default : Default_Hello_PTC</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_7_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| <p>Detailed Comments :</p> <p>(1) Bring the IUT to the state Attempt.</p> <p>(2) Send a Hello where the Peer Group ID does not match with the IUT's peer group ID, and where the remote node ID and remote Port Id fields are equal to the IUT's Node ID and Port ID respectively. An empty hierarchy list is included also ULIA and aggregation token are included. (Two Way Outside Hello)</p> <p>(3) Check that no Hello is sent in response.</p> <p>(4) Check that the IUT remains in the state Attempt.</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| <p>Test Step Name : Hello_FSM_V204</p> <p>Group : Bodies/Hello/SS_M/ATTEMPT/</p> <p>Objective : To verify that while in the Attempt state and a Common Hierarchy Received event are generated that the IUT does nothing.</p> <p>Default : Default_Hello_PTC</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_6_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| <p>Detailed Comments :</p> <p>(1) Bring the IUT to the state Attempt.</p> <p>(2) Send a Hello with remote NodeID and remote PortID fields equal to the IUT's NodeID and PortID respectively and the PeerGroupID does not match with the IUT's one. And including a hierarchy list where there is a common higher level Peer Group.</p> <p>(3) If a Hello is received in response from the IUT then the Test fails, otherwise the T_NoResp timer expires and the test is successful.</p> <p>(4) Check that the IUT remains in the state Attempt.</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V205 | | | | | |
| Group : Bodies/Hello/SS_M/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a Hello Mismatch Received event is generated, the IUT does nothing. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_I_1_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote NodeID wrong. (It should be interpreted as a Hello Mismatch event). Start T_NoResp Timer. (3) If a Hello is received in response then the test fails otherwise the T_NoResp timer expires and the test success. (4) Check that the IUT remains in the state Attempt. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V206 | | | | | |
| Group : Bodies/Hello/SS_M/ATTEMPT/ | | | | | |
| Objective : To verify that when in the Attempt state, that the Hellos have their remote node ID and remote port ID set to zero. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_5_r | | (2) |
| 4 | | + CheckStateHelloAttempt | | | (3) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Wait until receive a Hello and check that the remote node ID and remote port ID fields are set to zero. (3) Check that the IUT remains in the state Attempt. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V207 | | | | | |
| Group : Bodies/Hello/SS_M/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT does nothing. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_4_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the peer group ID not matching the IUT's one and with the remote Node ID and remote Port ID fields equal to zero. (3) The T_NoResp timer expires and no Hello should be received. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V301 | | | | | |
| Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_2_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matching with the IUT's one. (One Way Inside Hello) (3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero. Stop the T_Inact timer and check that its value is THIXIF then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V302_1 | | | | | |
| Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that 2-Way Inside state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_3_s | | (2) |
| 3 | | + CheckStateHelloTWI | | | (3) |
| 4 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUTs Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. (3) Check that the IUT is in the state Two Way Inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V302_2 | | | | | |
| Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Inside state and a Two Way Inside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_3_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (5) |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | (5) |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUTs Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. (Two Way Inside Hello). Start T_Inact timer. (3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero. Stop the T_Inact timer. (5) Check that the value of the T_Inact timer when it was stopped it was IF times THI. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V303_1 | | | | | |
| Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, and the Attempt state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state One Way Inside. | | | | | |
| (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. And with an error in the Node ID field. (Hello Mismatch) | | | | | |
| (3) Receive a Hello in response with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Stop the T_Resp timer. | | | | | |
| (4) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V303_2 Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ Objective : To verify that while in the 1-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp, START T_Hello | Hello_V_4_r | | (3) |
| 4 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (4) |
| 5 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 6 | | +PostambleHelloAttempt | | | |
| 7 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. And with an error in the Node ID field. (Hello Mismatch) (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. Start T_Hello timer. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the T_Hello timer. If the T_Hello timer is equal to the IUT's tsv_R_HI then the test is successful. Note.- The Tester can not check that the Inactivity timer is disabled and that the information in the IUT's Hello data structure is also cleared. So this part has been deleted from the test purpose. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V304_1 Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ Objective : To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_2_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + CheckStateHelloAttempt | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| 9 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 10 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. (One Way Inside Hello). Start the Inactivity timer. (3) Receive several Hellos with the remote node ID and remote port ID m fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID matches with the Tester's one. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the Inact timer and compare its value against IF times HI. (5) Check that the IUT is in the state Attempt at the end of the test. Note.-in (2) triggered an event One Way Inside Received because in One Way Inside this event does not make any change, so the Inactivity timer can be initialized. That the Inactivity timer is disabled can not be tested. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V304_2 Group : Bodies/Hello/SS_M/ONE_WAY_INSIDE/ Objective : To verify that while in the 1-Way Inside state and Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_2_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, START T_Hello | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (5) |
| 8 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 9 | | + PostambleHelloAttempt | | | |
| 10 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)))] | | (F) | |
| 11 | | +PostambleHelloAttempt | | | |
| 12 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)))] | | (F) | |
| 13 | | CANCEL T_Hello | | | |
| 14 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. (One Way Inside Hello). Start the Inactivity timer. (3) Receive several Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID matches with the Tester's one. (4) Receive a hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Store in a local variable the value of the received Hello Interval. Stop the T_Inact timer and compare its value against IF times HI. Start the T_Hello timer. (5) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Compare the value of the T_Hello timer against tsv_R_HI. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V401_1 | | | | | |
| Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, a Hello is sent, and the 1-Way Inside state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_2_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | (3) |
| 4 | | + CheckStateHelloOWI | | | (3) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Two Way Inside. | | | | | |
| (2) Send a Hello with the remote Node ID and remote Port ID equal to zero, and with the Peer Group ID matching the IUT's peer group ID. (One Way Inside received event) | | | | | |
| (3) Receive a Hello in response with the remote node ID and remote port ID fields equal to the Tester's Node ID and Port ID respectively. | | | | | |
| (4) Check that the IUT is in the state One Way Inside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V401_2 Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ Objective : To verify that while in the 2-Way Inside state and a 1-Way Inside Received event is generated, that the Inactivity timer is restarted, a Hello is sent, that Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact, START T_Resp | Hello_V_2_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_3_r | | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_3_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)))] | | (F) | |
| 9 | | CANCEL T_Inact | | | |
| 10 | | +PostambleHelloAttempt | | | |
| 11 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, CANCEL T_Hello | Hello_V_4_r | | (5) |
| 12 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 13 | | +PostambleHelloAttempt | | | |
| 14 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)))] | | (F) | |
| 15 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero, and with the Peer Group ID matching the IUT's peer group ID. (One Way Inside received event). Start T_Inact Timer. Start T_Resp Timer. | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|--------------------------------|---|
| Detailed Comments : ... | <p>(3) Receive a Hello in response, with the remote node ID and remote port ID fields equal to the Tester's Node ID and Port ID respectively. Store the local variable tsv_R_HI with the Hello Interval value receive in the Hello.</p> <p>(4) Check that Hellos are received periodically every tsv_R_HI Intervals.</p> <p>(5) Receive a Hello with the remote Node ID and remote Port ID equal to zero and the Version field cleared. Stop the T_Inact timer and check that its value is Inactivity Factor times the Tester's Hello Interval.</p> |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V402 | | | | | |
| Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Inside state and a 2-Way Inside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_3_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | +PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Two Way Inside. | | | | | |
| (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively, and the Peer Group ID matching the IUT's one (2-Way Inside Received event). Start T_Inact Timer. | | | | | |
| (3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. | | | | | |
| (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matching the IUT's one. Cancel the Inact timer and check that its value is THIXIF then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V403_1 | | | | | |
| Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, a Hello is sent, and the Attempt state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. And with an error in the Node ID field, (Hello Mismatch received event). Start the T_Resp Timer. (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. (4) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V403_2 | | | | | |
| Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Inside state and a Hello Mismatch Received event is generated, that a Hello is sent, Hello Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_1_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp, START T_Hello | Hello_V_4_r | | (3) |
| 4 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (4) |
| 5 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 6 | | +PostambleHelloAttempt | | | |
| 7 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Two Way Inside. | | | | | |
| (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID matches with the IUT's one. And with an error in the Node ID field, (Hello Mismatch received event). Start the T_Resp Timer. | | | | | |
| (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. Start T_Hello timer. | | | | | |
| (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the T_Hello timer. If the Hello timer is equal to the tsv_R_HI (IUT's Hello Interval) then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V404_1 Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ Objective : To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_3_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + CheckStateHelloAttempt | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| 9 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 10 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively. Start the T_Inact Timer (Two Way Inside received event). Start T_Inact Timer. (3) Receive several Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID matches with the Tester's one. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the Inact timer and compare its value against IF times HI. (5) Check that the IUT is in the state Attempt at the end of the test. Note.-in (2) triggered an event Two Way Inside Received because in Two Way Inside this event does not make any transition or triggers any hello, so the Inactivity timer can be initialized and nothing changes. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V404_2 Group : Bodies/Hello/SS_M/TWO_WAY_INSIDE/ Objective : To verify that while in the 2-Way Inside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWI | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_3_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_3_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, START T_Hello | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (5) |
| 8 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 9 | | + PostambleHelloAttempt | | | |
| 10 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 11 | | +PostambleHelloAttempt | | | |
| 12 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 13 | | CANCEL T_Hello | | | |
| 14 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Inside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively. Start the T_Inact Timer (Two Way Inside received event). Start T_Inact Timer. (3) Receive several Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID matches with the Tester's one. (4) Receive a hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Store in a local variable the value of the received Hello Interval. Stop the T_Inact timer and compare its value against IF times HI. Start the T_Hello timer. (5) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Compare the value of the T_Hello timer against tsv_R_HI. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V501_1 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that when in state One Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_4_s | | (2) |
| 3 | | START T_Period | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_7_r | | (3) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Period | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | ?TIMEOUT T_Period | | | (4) |
| 12 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside. (2) Send a Hello with the remote NodeID and remote PortID fields set to zero and the Peer Group ID not matching the IUT's one. Start the Period and Hell timers. (3) Receive a Hello with the remote NodeID and remote PortID fields equal to IUT's NodeID and PortID. And check that the value of the Hell timer is around tsv_R_HI plus or minus the fractional variance of 25%. Restart the Hell timer. (4) The Period timer is an interval timer. If Hellos are received every tsv_R_HI time until the Period timer expires, then it is considered that this action happens periodically in the state One Way Outside. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V501_2 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that when in state Two Way Outside, that the IUT transmits Hellos periodically (i.e. every HelloInterval seconds). Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_7_s | | (2) |
| 3 | | START T_Period | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_7_r | | (3) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 9 | | CANCEL T_Period | | | |
| 10 | | + PostambleHelloAttempt | | | |
| 11 | | ?TIMEOUT T_Period | | | (4) |
| 12 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start the Period and Hell timers. (3) Receive a Hello with the remote NodeID and remote PortID fields equal to IUT's NodeID and PortID. And check that the value of the Hell timer is around tsv_R_HI plus or minus the fractional variance of 25%. Restart the Hell timer. (4) The Period timer is an interval timer. If Hellos are received every tsv_R_HI time until the Period timer expires, then it is considered that this action happens periodically in the state Two Way Outside. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V502_1 | | | | | |
| Group : Bodies/Hello/SS_B/GENERAL/ | | | | | |
| Objective : To verify that a Hello is sent upon state change from Attempt to One Way Outside subject to the HoldDown timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_5_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (5) |
| 7 | | +CheckStateHelloOWO | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Attempt. | | | | | |
| (2) Receive a Hello with the remote NodeID and remote PortID fields set to zero. Start the Hold timer. | | | | | |
| (3) Send a Hello with the remote node ID and remote port ID fields equal 0 and port id and the peer group ID not equal to the iut's peer group ID. Start the T_Resp timer. (one way outside). | | | | | |
| (4) No-message should be receive before the hold timer expires. | | | | | |
| (5) Receive in response a Hello with the remote node id field and remote port id field equal to the tester's node id an port id respectively., and including a hierachy list and outgoing service category metrics is's before the T_Resp timer expires. | | | | | |
| (6) Check that the IUT is in the state One Way Outside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V502_2 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that a Hello is sent upon state change from Attempt to Two Way Outside subject to the HoldDown timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_5_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_7_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (5) |
| 7 | | +CheckStateHelloTWO | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Receive a Hello with the remote NodeID and remote PortID fields set to zero. Start the Hold timer. (3) Send a Hello with the remote node ID and remote port ID fields equal to the IUTs node ID and port ID and port id and the peer group ID not equal to the iut's peer group ID. Start the T_Resp timer. (Two Way Outside). (4) No-message should be receive before the hold timer expires. (5) Receive in response a Hello with the remote node id field and remote port id field equal to the tester's node id an port id respectively., and including a hierachy list and outgoing service category metrics is's before the T_Resp timer expires. (6) Check that the IUT is in the state Two Way Outside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V502_3 | | | | | |
| Group : Bodies/Hello/SS_B/GENERAL/ | | | | | |
| Objective : To verify that a Hello is sent upon state change from One Way Outside to Attempt subject to the HoldDown timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_7_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (5) |
| 7 | | +CheckStateHelloAttempt | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state one way outside. | | | | | |
| (2) Receive a Hello with the remote NodeID and remote PortID fields equal to the testers node id and port id, and including a nodal hierachy list, ulia and aggregation token. Start the Hold timer. | | | | | |
| (3) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. The node id field has a wrong value (i.e. hello packet with a mismatch). | | | | | |
| (4) No-message should be receive before the hold timer expires. | | | | | |
| (5) Receive in response a Hello with the remote node id field and remote port id field equal 0. And version field equal to the newest version. Otherwise, if the resp timer expires first -> test fails. | | | | | |
| (6) Check that the iut is in the state attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V502_4 | | | | | |
| Group : Bodies/Hello/SS_B/GENERAL/ | | | | | |
| Objective : To verify that a Hello is sent upon state change from Two Way Outside to Attempt subject to the HoldDown timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello, START T_Hold | Hello_V_7_r | | (2) |
| 4 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (3) |
| 5 | | ?TIMEOUT T_Hold | | | (4) |
| 6 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (5) |
| 7 | | +CheckStateHelloAttempt | | | (6) |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Receive a Hello with the remote NodeID and remote PortID fields equal to the testers node id and port id, and including a nodal hierachy list, ulia and aggregation token. Start the Hold timer. (3) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. The node id field has a wrong value (i.e. hello packet with a mismatch). (4) No-message should be receive before the hold timer expires. (5) Receive in response a Hello with the remote node id field and remote port id field equal 0. And version field equal to the newest version. Otherwise, if the resp timer expires first -> test fails. (6) Check that the iut is in the state attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V502_5 | | | | | |
| Group : Bodies/Hello/SS_B/GENERAL/ | | | | | |
| Objective : To verify that a Hello is not sent upon state change from One Way Outside to Two Way Outside state. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_7_s | | (2) |
| 3 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| 4 | | +CheckStateHelloTWO | | | (4) |
| 5 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state one way outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (3) T_NoResp timer expires, this means that no Hello was received in response to the last Hello sent by the tester. (4) Check that the IUT can be in the state 2 way outside or common outside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|------------|
| Test Step Name : Hello_FSM_V503 | | | | | |
| Group : Bodies/Hello/SS_B/GENERAL/ | | | | | |
| Objective : To verify that when multiple event triggered Hellos are deferred because of the HoldDown timer, that the IUT sends only one Hello which contains the most current information for all IGs when the HoldDown timer expires. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tsp_THNID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tstv_R_PID := Hello_T.o_port, tstv_R_HI := Hello_T.hello_int) CANCEL T_Resp, START T_Hold | Hello_V_1_r | | (2) (3) |
| 4 | | Hello_PCO_1!Hello_T | Hello_V_7_s | | (4) |
| 5 | | Hello_PCO_1!Hello_T | Hello_I_5_s | | (5) |
| 6 | | Hello_PCO_1!Hello_T | Hello_V_2_s | | (6) |
| 7 | | ?TIMEOUT T_Hold | | | (7) |
| 8 | | Hello_PCO_1?Hello_T | Hello_V_3_r | (P) | (8) |
| 9 | | +PostambleHelloAttempt | | | |
| 10 | | Hello_PCO_1?Hello_T | Hello_V_7_r | (F) | |
| 11 | | +PostambleHelloAttempt | | | |
| 12 | | Hello_PCO_1?Hello_T | Hello_V_4_r | (F) | |
| 13 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state Down. | | | | | |
| (2) Receive a Hello with the remote NodeID and remote PortID equal to zero. Start the Hold timer. Store in variables the values from the received hello. | | | | | |
| (3) Next, three Hellos are going to be sent by the Tester, forcing them to be sent before the HOLD timer expires | | | | | |
| (4) Send a Hello with the remote NodeID, remote PortID fields equal to the IUT's NodeID and PortID, includes a empty hierarchy list, and IGs. The Peer Group ID does not match with the IUT's one. This Hello should be interpreted as a Two Way Outside. In response the IUT should send a Hello with the remote node ID and remote Port ID equal to the Tester's Node ID and Port ID, and including a hierarchy list and all outgoing IGs. | | | | | |
| (5) Send a Hello with a mismatch, in this case the Tester's node ID is different to the previous advertised. This should be interpreted as a Hello Mismatch received event. So in response the IUT should send a Hello with the remote node ID, remote Port ID fields equal to zero. And the Version field with the value of the newest version. | | | | | |
| (6) Send a Hello with the remote NodeID and remote PortID equal to zero, and the Peer Group Id that matches with the IUT's peer group ID. This should be interpreted as a One Way Inside received event, and in response it should send a Hello with the remote node ID and remote Port ID equal to the Tester's Node ID and Port ID respectively. | | | | | |
| (7) HOLD timer expires. No hello should have been received before the HOLD timer expires. | | | | | |
| (8) Receive a hello with the remote node ID and remote PortID equal to the last advertised Tester's NodeID and PortID and with no Hierarchy list or Outgoing Information groups included. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V504_1 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that when in state One Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_7_r | (P) | (2) |
| 4 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside (2) Receive a Hello with the remote node ID and remote Port ID fields equal to the Tester's Node ID and Port ID respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V504_2 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that when in state Two Way Outside, that the Hellos have their remote node ID and remote port ID fields set to the neighbor node's node ID and port ID. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_7_r | (P) | (2) |
| 4 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside (2) Receive a Hello with the remote node ID and remote Port ID fields equal to the Tester's Node ID and Port ID respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V505 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that the sequence number of the first instance of the nodal hierarchy list sent to any neighbor is greater than zero. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T [OCT_TO_INT(Hello_T.ig.nhl.seq_num) > 0] CANCEL T_Resp | Hello_V_7_r | (P) | (3) (4) |
| 4 | | +PostambleHelloAttempt | | | |
| 5 | | Hello_PCO_1?Hello_T [OCT_TO_INT(Hello_T.ig.nhl.seq_num) = 0] CANCEL T_Resp | Hello_V_7_r | (F) | (5) |
| 6 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero and the Peer Group ID not matching the IUT's Peer Group ID. (One Way Outside received event) (3) Receive a Hello including a Nodal hierarchy list. (4) Check that the sequence number of the Nodal hierarchy list is greater than zero. (5) Otherwise, the test fails | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V506 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that if no higher level is known, that an empty nodal hierarchy list is included in the Hello. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T [OCT_TO_INT(Hello_T.ig.nhl.level)] = 0] CANCEL T_Resp | Hello_V_7_r | (P) | (3) (4) |
| 4 | | +PostambleHelloAttempt | | | |
| 5 | | Hello_PCO_1?Hello_T [OCT_TO_INT(Hello_T.ig.nhl.level) > 0] CANCEL T_Resp | Hello_V_7_r | (F) | (5) |
| 6 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt and is suppose that no higher level in the hierarchy is known. (2) Send a Hello with the remote Node ID and remote Port ID fields equal to zero. And the Peer Group ID not matching the IUT's one. (One Way Outside received event) (3) Receive a response from the IUT with the remote Node ID and remote Port ID fields equal to the Tester's Node ID and Port ID , and including a Hierarchy List. (4) Check that the value of Level in the Hierarchy List is equal to zero, i.e. that the nodal hierarchy list is empty. (5) Otherwise if the Level is greater than zero test fails. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V507_1 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that the ULIA information group is included in all Hellos while in the states: 1-Way Outside. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Period | Hello_V_4_s | | (2) |
| 3 | | START T_Hello | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_7_r | (P) | (3) (4) |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_Period | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside. (2) Send a Hello with the remote NodeID and remote PortID fields set to zero and the Peer Group ID not matching the IUT's one. Start the Period and Hell timers. (3) Receive a Hello where the remote node ID and remote Port ID fields are equal to the Tester's node ID and Port ID respectively. (4) Check that a ULIA IG is included in the received Hello. Restart the Hell timer. (5) This sequence of events is repeated until the Period timer expires. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V507_2 Group : Bodies/Hello/SS_B/GENERAL/ Objective : To verify that the ULIA information group is included in all Hellos while in the states: 2-Way Outside. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_7_s | | (2) |
| 3 | | START T_Period | | | |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_7_r | (P) | (3) (4) |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_Period | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start the Period and Hell timers. (3) Receive a Hello where the remote node ID and remote Port ID fields are equal to the Tester's node ID and Port ID respectively. (4) Check that a ULIA IG is included in the received Hello. Restart the Hell timer. (5) This sequence of events is repeated until the Period timer expires. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V601_1 Group : Bodies/Hello/SS_B/ATTEMPT/ Objective : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (3) |
| 4 | | + CheckStateHelloOWO | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero and with the Peer Group ID not matching the IUT's one. (One Way Outside received event) (3) Receive a Hello in response the Tester's node ID and port ID in the remote node ID and remote Port ID fields, and including a hierarchical list and all outgoing IGs. The Hello_V_7_r messages is used since the Hierarchy List is known that is included, but its content is unknown by the tester, and not necessary for this test. (4) Check that the IUT is in the state One Way Out at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V601_2 | | | | | |
| Group : Bodies/Hello/SS_B/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact, START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_7_r | | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_7_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)))] | | (F) | |
| 9 | | CANCEL T_Inact | | | |
| 10 | | +PostambleHelloAttempt | | | |
| 11 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, CANCEL T_Hello | Hello_V_4_r | | (5) |
| 12 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (6) |
| 13 | | +PostambleHelloAttempt | | | |
| 14 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)))] | | (F) | (6) |
| 15 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero and with the Peer Group ID not matching the IUT's one. (One Way Outside received event) (3) Receive in response a Hello with the Tester's node ID and port ID in | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|--------------------------------|--|
| Detailed Comments : ... | the remote node ID and remote Port ID fields, and including a hierarchical list. (4) Receive Hellos with the same content as the last received Hello. (5) Receive a Hello with the remote Node ID and remote Port ID and the Version field equal to zero. Stop the T_Inact timer. (6) Check that the Inactivity timer is equal to Inactivity factors times Hello Intervals. |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V602_1 | | | | | |
| Group : Bodies/Hello/SS_B/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 2-Way Outside state. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_7_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (3) |
| 4 | | + CheckStateHelloTWO | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) (3) Receive a Hello in response the Tester's node ID and port ID in the remote node ID and remote Port ID fields, and including a hierarchical list and all outgoing IGs. The Hello_V_7_r messages is used since the Hierarchy List is known that is included, but its content is unknown by the tester, and not necessary for this test. (4) Check that the IUT is in the state Two Way Out at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V602_2 | | | | | |
| Group : Bodies/Hello/SS_B/ATTEMPT/ | | | | | |
| Objective : To verify that while in the Attempt state and a 2-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact, START T_Resp | Hello_V_7_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_7_r | | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_7_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)))] | | (F) | |
| 9 | | CANCEL T_Inact | | | |
| 10 | | +PostambleHelloAttempt | | | |
| 11 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, CANCEL T_Hello | Hello_V_4_r | | (5) |
| 12 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | (6) |
| 13 | | +PostambleHelloAttempt | | | |
| 14 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)))] | | (F) | (6) |
| 15 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) (3) Receive in response a Hello with the Tester's node ID and port ID in | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|-----------------------------|---|
| Detailed Comments : | <p>... the remote node ID and remote Port ID fields, and including a hierarchical list.</p> <p>(4) Receive Hellos with the same content as the last received Hello.</p> <p>(5) Receive a Hello with the remote Node ID and remote Port ID and the Version field equal to zero. Stop the T_Inact timer.</p> <p>(6) Check that the Inactivity timer is equal to Inactivity factors times Hello Intervals.</p> |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V701 | | | | | |
| Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Outside state and a 1-Way Outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_4_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| <p>(1) Bring the IUT to the state One Way Outside.</p> <p>(2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively, and the Peer Group ID does not match with the IUT's one. (One Way Outside received event). Start T_Inact Timer.</p> <p>(3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the IUT's one.</p> <p>(4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID does not match with the IUT's one. Cancel the Inact timer and check that its value is THixIF then the test is successful.</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|-------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V702_1 | | | | | |
| Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the IUT enters the 2-Way Outside state. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_7_s | | (2) |
| 3 | | + CheckStateHelloTWO | | | (3) |
| 4 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) (3) Check that the IUT is in the state Two Way Outside at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V702_2 | | | | | |
| Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_7_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start T_Inact Timer. (3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the IUT's one. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID does not match with the IUT's one. Cancel the Inact timer and check that its value is THixIF then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V703_1 Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ Objective : To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID not matching with the IUT's one. And with an error in the Node ID field. (Hello Mismatch received event) (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. (4) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V703_2 | | | | | |
| Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 1-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp, START T_Hello | Hello_V_4_r | | (3) |
| 4 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (4) |
| 5 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 6 | | +PostambleHelloAttempt | | | |
| 7 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : | | | | | |
| (1) Bring the IUT to the state One Way Outside. | | | | | |
| (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID not matching with the IUT's one. And with an error in the Node ID field. (Hello Mismatch received event) | | | | | |
| (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. Start T_Hello timer. | | | | | |
| (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the T_Hello timer. If the Hello timer is equal to the tsv_R_HI (IUT's Hello Interval) then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V704_1 Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ Objective : To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_4_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + CheckStateHelloAttempt | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| 9 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 10 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside (2) Send a Hello with the remote node ID and remote port ID fields equal zero and the Peer Group ID does not match with the IUT's one. Start the Inactivity timer. (3) Receive several Hello with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the Tester's one. And includes a hierarchy list and outgoing IGs. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the Inact timer and compare its value against IF times HI. (5) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V704_2 Group : Bodies/Hello/SS_B/ONE_WAY_OUTSIDE/ Objective : To verify that while in the 1-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloOWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_4_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, START T_Hello | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (5) |
| 8 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 9 | | + PostambleHelloAttempt | | | |
| 10 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 11 | | +PostambleHelloAttempt | | | |
| 12 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 13 | | CANCEL T_Hello | | | |
| 14 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state One Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal zero and the Peer Group ID does not match with the IUT's one. Start the Inactivity timer. (3) Receive several Hello with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the Tester's one. And includes a hierarchy list and outgoing IGs. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Store in a local variable the value of the received Hello Interval. Stop the T_Inact timer and compare its value against IF times HI. Start the T_Hello timer. (5) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Compare the value of the T_Hello timer against tsv_R_HI. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V801_1 Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ Objective : To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT sends a Hello with nodal hierarchy information and enters the 1-Way Outside state. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (3) |
| 4 | | + CheckStateHelloOWO | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two-Way Outside. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero and with the Peer Group ID not matching the IUT's one. (One Way Outside received event) (3) Receive a Hello in response the Tester's node ID and port ID in the remote node ID and remote Port ID fields, and including a hierarchical list and all outgoing IGs. (4) Check that the IUT is in the state One Way Out at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V801_2 Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ Objective : To verify that while in the 2-Way Outside state and a 1-Way Outside Received event is generated, that the IUT starts the Inactivity Timer, sends a Hello with nodal hierarchy information and restarts the Hello Timer. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact, START T_Resp | Hello_V_4_s | | (2) |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_7_r | (P) | (3) |
| 4 | L1 | START T_Hello | | | |
| 5 | | Hello_PCO_1?Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_7_r | | (4) |
| 6 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 7 | | GOTO L1 | | | |
| 8 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] CANCEL T_Inact | | (F) | |
| 9 | | +PostambleHelloAttempt | | | |
| 10 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, CANCEL T_Hello | Hello_V_4_r | | (5) |
| 11 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 12 | | +PostambleHelloAttempt | | | |
| 13 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 14 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two-Way Outside. (2) Send a Hello with the remote Node ID and remote Port ID equal to zero and with the Peer Group ID not matching the IUT's one. (One Way Outside received event) (3) Receive a Hello in response the Tester's node ID and port ID in the remote node ID and remote Port ID fields, and including a | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|--------------------------------|---|
| Detailed Comments : ... | hierarchical list and all outgoing IGs. (4) Check that Hellos are received periodically every tsv_R_HI Intervals. (5) Receive a Hello with the remote Node ID and remote Port ID equal to zero and the Version field cleared. Stop the T_Inact timer and check that its value is Inactivity Factor times the Tester's Hello Interval. |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V802 | | | | | |
| Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Outside state and a 2-Way Outside Received event is generated, that the Inactivity Timer is restarted. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_7_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + PostambleHelloAttempt | | | |
| 8 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 9 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start T_Inact Timer. (3) Receive some Hellos with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the IUT's one. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID does not match with the IUT's one. Cancel the Inact timer and check that its value is THixIF then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V803_1 | | | | | |
| Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ | | | | | |
| Objective : To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and the Attempt state is entered. | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp | Hello_V_4_r | (P) | (3) |
| 4 | | + CheckStateHelloAttempt | | | (4) |
| 5 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID not matching with the IUT's one. And with an error in the Node ID field. (Hello Mismatch received event) (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. (4) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V803_2 Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ Objective : To verify that while in the 2-Way Outside state and a Hello Mismatch Received event is generated, that a Hello is sent and Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Resp | Hello_I_5_s | | (2) |
| 3 | | Hello_PCO_1 ? Hello_T CANCEL T_Resp, START T_Hello | Hello_V_4_r | | (3) |
| 4 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (4) |
| 5 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 6 | | +PostambleHelloAttempt | | | |
| 7 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 8 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively and the Peer Group ID not matching with the IUT's one. And with an error in the Node ID field. (Hello Mismatch received event) (3) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one and the Version field is also set to zero. Cancel the T_Resp timer. Start T_Hello timer. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the T_Hello timer. If the Hello timer is equal to the tsv_R_HI (IUT's Hello Interval) then the test is successful. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V804_1 Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ Objective : To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Attempt state is entered. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_7_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1?Hello_T READTIMER T_Inact(tcv_TIME), CANCEL T_Inact | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | + CheckStateHelloAttempt | | | (5) |
| 8 | | + PostambleHelloAttempt | | | |
| 9 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 10 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start the Inactivity timer. (3) Receive several Hello with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the Tester's one. And includes a hierarchy list and outgoing IGs. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Stop the Inact timer and compare its value against IF times HI. (5) Check that the IUT is in the state Attempt at the end of the test. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : Hello_FSM_V804_2 Group : Bodies/Hello/SS_B/TWO_WAY_OUTSIDE/ Objective : To verify that while in the 2-Way Outside state and the Inactivity Timer expires, that a Hello is sent and the Hello Timer is restarted. Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloTWO | | | (1) |
| 2 | | Hello_PCO_1!Hello_T START T_Inact | Hello_V_7_s | | (2) |
| 3 | L1 | Hello_PCO_1?Hello_T | Hello_V_7_r | | (3) |
| 4 | | GOTO L1 | | | |
| 5 | | Hello_PCO_1 ? Hello_T (tsv_R_HI := Hello_T.hello_int) READTIMER T_Inact(tcv_TIME), CANCEL T_Inact, START T_Hello | Hello_V_4_r | | (4) |
| 6 | | [(tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4)] | | (P) | |
| 7 | | Hello_PCO_1?Hello_T READTIMER T_Hello(tcv_TIME), CANCEL T_Hello | Hello_V_5_r | | (5) |
| 8 | | [(tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250)] | | (P) | |
| 9 | | + PostambleHelloAttempt | | | |
| 10 | | [NOT((tcv_TIME >= OCT_TO_INT(tsv_R_HI) * 1000 - OCT_TO_INT(tsv_R_HI) * 250) AND (tcv_TIME <= OCT_TO_INT(tsv_R_HI) * 1000 + OCT_TO_INT(tsv_R_HI) * 250))] | | (F) | |
| 11 | | +PostambleHelloAttempt | | | |
| 12 | | [NOT((tcv_TIME >= tsp_IF * tsp_THI - (tsp_IF * tsp_THI)/4) AND (tcv_TIME <= tsp_IF * tsp_THI + (tsp_IF * tsp_THI)/4))] | | (F) | |
| 13 | | CANCEL T_Hello | | | |
| 14 | | + PostambleHelloAttempt | | | |
| Detailed Comments : (1) Bring the IUT to the state Two Way Outside (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) Start the Inactivity timer. (3) Receive several Hello with the remote node ID and remote port ID fields equal to Tester's Node ID and Port ID respectively and the Peer Group ID does not match with the Tester's one. And includes a hierarchy list and outgoing IGs. (4) Receive a Hello with the remote node ID and remote port ID fields equal to zero and the Peer Group ID matches with the Tester's one. Store in a local variable the value of the received Hello Interval. Stop the T_Inact timer and compare its value against IF times HI. Start the T_Hello timer. (5) Receive a Hello with the remote node ID and remote port ID fields | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | | | | | |
|---|--|--|--|--|--|
| Detailed Comments : ... | | | | | |
| equal to zero and the Peer Group ID matches with the Tester's one. Compare the value of the T_Hello timer against tsv_R_HI. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V001_MS | | | | | |
| Group : Bodies/DBSynchronization/NPDown/ | | | | | |
| Objective : To verify that when a link reaches the Hello state Two Way Inside, that the event AddPort is triggered. | | | | | |
| Default : Default_Neighb_Peer_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_Neighb_Peer_CP_1?AddPort_T | AddPort_V_1 | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | (P) | (a) |
| 4 | | ?TIMEOUT T_Resp | | (F) | (b) |
| Detailed Comments : The IUT must begin sending a summary of the contents of its topology database to the neighboring peer in Database Summary packets. The topology database consists of the PTSEs either originated or received by this node, at the level of this node's peer group or at a higher level. | | | | | |
| Two possibilities: | | | | | |
| (a) Receive DS packet with Initialize bit set to one. Stop Response timer. -> Pass | | | | | |
| (b) Timeout Response timer -> Fail | | | | | |
| PNNI 1.0 5.7 PICS 3.14.8 | | | | | |
| atm98-0466: DS.10 | | | | | |
| Applies to Master and Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V101_MS | | | | | |
| Group : Bodies/DBSynchronization/Negotiating/ | | | | | |
| Objective : To verify that when in the Negotiating state, that the IUT sends empty Database Summary packets with the I, M and MS bits set. | | | | | |
| Default : Default_Neighb_Peer_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_TLN ID) | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | (P) | (1) |
| 4 | | ?TIMEOUT T_Resp | | (F) | |
| Detailed Comments : In the Negotiating state, the node sends empty Database Summary packets, with the Initialize, More and Master bits set. | | | | | |
| (1) Accept the first empty DS packet from IUT | | | | | |
| PNNI 1.0 5.7.5 PICS 3.14.52 | | | | | |
| Applies to Master and Slave role of the IUT | | | | | |
| atm98-0466: DS.54 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V102_MS Group : Bodies/DBSynchronization/Negotiating/ Objective : To verify that the initial empty Database Summary packets that are not acknowledged are retransmitted every DSRxmtInterval seconds. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_TLN ID) | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | | (1) |
| 4 | | START T_DSRxmt | | | (2) |
| 5 | | Neighb_Peer_PCO_1 ?DBSP_T READTIMER T_DSRxmt(tcv_TIME), CANCEL T_DSRxmt | DBSP_V_1_r | (P) | (3) |
| 6 | | [(tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4)] | | (P) | |
| 7 | | [NOT((tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4))] | | (F) | |
| 8 | | ?TIMEOUT T_DSRxmt | | (F) | (4) |
| 9 | | ?TIMEOUT T_Resp | | (I) | |
| Detailed Comments : In the Negotiating state, the node sends empty Database Summary packets, with the Initialize, More and Master bits set. When sending such Database Summary packets, the DS Rxmt Timer must be restarted. These packets are retransmitted every DSRxmtInterval seconds, when the DS Rxmt Timer fires. (1) Accept the first empty DS packet from IUT (2) Start the DSRxmtInterval Timer (3) Retransmitted first empty DS packet from IUT as expected within the DSRxmt period (4) No retransmitted first empty DS packet from IUT PNNI 1.0 5.7.5 PICS 3.14.54 Applies to Master and Slave role of the IUT atm98-0466: DS.56 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V103_MS Group : Bodies/DBSynchronization/Negotiating/ Objective : To verify that the DSRxmt timer is restarted after sending the initial empty Database Summary packet. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_TLN ID) | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | | (1) |
| 4 | | START T_DSRxmt | | | (2) |
| 5 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_DSRxmt | DBSP_V_1_r | (P) | (3) |
| 6 | | START T_DSRxmt | | | (2) |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T READTIMER T_DSRxmt(tcv_TIME), CANCEL T_DSRxmt | DBSP_V_1_r | (P) | (3) |
| 8 | | [(tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4)] | | (P) | |
| 9 | | [NOT((tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4))] | | (F) | |
| 10 | | ?TIMEOUT T_DSRxmt | | (F) | (4) |
| 11 | | ?TIMEOUT T_DSRxmt | | (F) | (4) |
| 12 | | ?TIMEOUT T_Resp | | (I) | |
| Detailed Comments : In the Negotiating state, the node sends empty Database Summary packets, with the Initialize, More and Master bits set. When sending such Database Summary packets, the DS Rxmt Timer must be restarted. These packets are retransmitted every DSRxmtInterval seconds, when the DS Rxmt Timer fires. | | | | | |
| (1) Accept the first empty DS packet from IUT (2) Start the DSRxmtInterval Timer (3) Retransmitted first empty DS packet from IUT as expected within the DSRxmt period (4) No retransmitted first empty DS packet from IUT | | | | | |
| PNNI 1.0 5.7.5 PICS 3.14.53 Applies to Master and Slave role of the IUT atm98-0466: DS.55 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V104_M Group : Bodies/DBSynchronization/Negotiating/ Objective : To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Master in the database synchronization. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_TLN ID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_1_r | | (2) |
| 4 | | Neighb_Peer_PCO_1 !DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (3) |
| 5 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | (P) | (4) |
| Detailed Comments : Preamble: (1) Negotiating State (2) Receive an empty DS packet with the Initialize, More and Master bit set to one (first init packet). (3) Send a DS packet with the Initialize, More and Master bit set to zero, copying the DS sequence number of the received packet (resulting in the NegotiationDone event in the IUT). (4) Receive a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one (first DS packet with information) PNNI 1.0 5.7.4 Table 5-12 Ds2 PICS 3.14.30 atm98-0466: DS.32a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|---------------------------------------|---------|----------|
| Test Step Name : DBSync_FSM_V105_S Group : Bodies/DBSynchronization/Negotiating/ Objective : To verify that when in the Negotiating state and the NegotiationDone event occurs, that the IUT begins sending Database Summary packets with information. The IUT takes the position of Slave in the database synchronization. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_THN ID) | | | |
| 2 | | (tsv_DSno:= DSno_init()) | | | |
| 3 | | START T_Resp | | | |
| 4 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | | (1) |
| 5 | | Neighb_Peer_PCO_1 !DBSP_T START T_Resp | DBSP_V_1_s(tsv_DSno) | | (2) |
| 6 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '0'B, tsv_DSno) | (P) | (3) |
| Detailed Comments : (1) Receive an empty DS packet with the Initialize, More and Master bit set to one (first init packet). (2) Send an empty DS packet with the Initialize, More and Master bit set to one, with a unique serial number (resulting in the NegotiationDone event in the IUT). (3) Receive a non-empty DS packet with the Initialize and Master bit set to zero and the sequence number the same as the one sent in the last packet (first DS packet with information) PNNI 1.0 5.7.4 Table 5-12 Ds2 PICS 3.14.30 atm98-0466: DS.32b Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V201_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that if Master, the DSRxmt Timer is restarted when the node receives a correct Database Summary packet. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (2) |
| 4 | | START T_DSRxmt | | | |
| 5 | | Neighb_Peer_PCO_1 ?DBSP_T READTIMER T_DSRxmt(tcv_TIME), CANCEL T_DSRxmt | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (3) |
| 6 | | [(tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4)] | | (P) | (4a) |
| 7 | | [NOT((tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4))] | | (F) | (4b) |
| 8 | | ?TIMEOUT T_DSRxmt | | (F) | |
| 9 | | ?TIMEOUT T_Resp | | (F) | |
| Detailed Comments : Preamble: (1) Exchanging State Slave Database Summary Answer on a received empty Database Summary Packet, additional PTSEs have to be summarized. Body: (2) Wait for a DS packet with the Initialize set to zero and Master set to one. Start timer. (3) Wait for the copy of the last DS packet with the Initialize set to zero and Master set to one. Stop timer. (4) Two possibilities: (a) If the timer value is equal to DSRxmtInterval (taking into account jitter) -> Pass (b) If the timer value is not equal to DSRxmtInterval (taking into account jitter) -> Fail PNNI 1.0 5.7.1 PICS 3.14.17 atm98-0466: DS.19 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V202_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Master, that Database Summary packets are sent when the Slave acknowledges the previous Database Summary packet and it has DS packets to send. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | (P) | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | (P) | (1b) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. Respond with a DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. PNNI 1.0 5.7.5 PICS 3.14.55 atm98-0466: DS.57 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V203_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Master and this packet includes the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to zero. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_NoResp | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | |
| 8 | L2 | Neighb_Peer_PCO_1?PTSP_T | PTSP_V_1_r | | (2) |
| 9 | | GOTO L2 | | | |
| 10 | | ?TIMEOUT T_NoResp | | (P) | (3) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. Respond with a DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. (Re-)start NoResponse timer. (2) Ignore Link advertisements (3) Timeout NoResponse timer -> Pass PNNI 1.0 5.7.5 PICS 3.14.56 atm98-0466: DS.58 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V204_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Master and all of the database summary has already been sent to the Slave, that the More (M) bit in the Database Summary packet is set to zero and the contents are empty. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (2) |
| 8 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '1'B, tsv_DSno+1) | (P) | (3) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (3) Receive an empty DS packet with Initialize and More set to zero, and Master set to one. PNNI 1.0 5.7.5 PICS 3.14.57 atm98-0466: DS.59 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V205_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Master and this packet does not include the last portions of the database summary to be sent to the Slave, that the more (M) bit is set to one. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | (P) | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | (P) | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (2) |
| 8 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (3) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (3) Receive an empty DS packet with Initialize and More set to zero, and Master set to one. PNNI 1.0 5.7.5 PICS 3.14.58 atm98-0466: DS.60 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V206_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Slave, that Database Summary packets are sent only in response to Database Summary packets received. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '0'B, tsv_DSno) | | (1) |
| 4 | | START T_DSRxmt(tsp_DSRxmt - tsp_DSRxmt/4) | | | |
| 5 | L1 | Neighb_Peer_PCO_1?PTSP_T | PTSP_V_1_r | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_DSRxmt | | (P) | (3) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bit set to zero. Stop Response timer. Start DSRxmt timer. (2) Ignore Link advertisements (3) Timeout DSRxmt timer (IUT doesn't take initiative in sending DS packet) -> Pass PNNI 1.0 5.7.5 PICS 3.14.59 atm98-0466: DS.61 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V207_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Slave and all of the database summary has already been previously sent to the Master, that the More (M) bit in the Database Summary packet is set to zero. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| 8 | | (tsv_DSno := tsv_DSno +1) | | | |
| 9 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (4) |
| 10 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | (P) | (5) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. (2) Respond with a DS packet setting the Initialize bit to zero and the More + Master bits to one. (3) Receive a non-empty DS packet with the Initialize, More and Master bits set to zero. Increase the DS sequence number with one. (4) Send a DS packet setting the Initialize bit to zero and the More + Master bits to one. (5) Receive an empty DS packet with Initialize, More and Master bits set to zero. -> Pass PNNI 1.0 5.7.5 PICS 3.14.60 atm98-0466: DS.62 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V208_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging when the node is Slave and this packet contains at least one item of the database summary to be sent to the Master, that the more (M) bit is set to one. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | (P) | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | (P) | (3) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. (2) Respond with a DS packet setting the Initialize bit to zero and the More + Master bits to one. (3) Receive a DS packet with the Initialize, More and Master bits set to zero. -> Pass PNNI 1.0 5.7.5 PICS 3.14.61 atm98-0466: DS.63 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V209_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is not empty, that the DS Rxmt Timer is stopped and (thus) no more DS packets are sent. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_DSRxmt(tsp_DSRxmt - tsp_DSRxmt/4) | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 8 | L2 | Neighb_Peer_PCO_1?PTSE_Req_T | PTSE_Req_V_1_r | | (3) |
| 9 | | GOTO L2 | | | |
| 10 | | ?TIMEOUT T_DSRxmt | | (P) | (4) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a non-empty DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. Start DSR timer. (2) Ignore Link advertisements (3) Timeout DSR timer -> Pass PNNI 1.0 5.7.6 PICS 3.14.86 atm98-0466: DS.88 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V210_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the DS Rxmt Timer is stopped, (thus) no more DS packets are sent. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_DSRxmt(tsp_DSRxmt - tsp_DSRxmt/4) | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 8 | L2 | Neighb_Peer_PCO_1?PTSP_T | PTSP_V_1_r | | (3) |
| 9 | | GOTO L2 | | | |
| 10 | | ?TIMEOUT T_DSRxmt | | (P) | (4) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send empty DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. Start DSR timer. (3) Ignore Link advertisements (4) Timeout DSR timer -> Pass PNNI 1.0 5.7.6 PICS 3.14.87 atm98-0466: DS.89a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V211_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has not sent its entire database that a new Database Summary packet is sent and the DS Rxmt Timer is restarted. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (1) |
| 4 | | START T_DSRxmt | | | |
| 5 | | Neighb_Peer_PCO_1 ?DBSP_T READTIMER T_DSRxmt(tcv_TIME), CANCEL T_DSRxmt | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (2) |
| 6 | | [(tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4)] | | (P) | (3) |
| 7 | | [NOT((tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4))] | | (F) | |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one. The DS sequence number is incremented with one compared to the one sent to the IUT in the last packet. Stop Response timer. Start timer. (2) Receive a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one. The DS sequence number is incremented with one compared to the one sent to the IUT in the last packet. Stop timer. (3) Timer value == DSRxmtInterval (taking jitter into account) -> Pass PNNI 1.0 5.7.6 PICS 3.14.88 atm98-0466: DS.90 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V212_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and the node is Master and a duplicate Database Summary packet is received, that the processing of this packet is stopped. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (1) |
| 4 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (3) |
| 5 | | START T_DSRxmt | | | |
| 6 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_DSRxmt | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | (P) | (2) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one. The DS sequence number is incremented with one compared to the one sent to the IUT in the last packet. Stop Response timer. (2) Re-send the last sent DS packet to the IUT. (IUT should view this packet as a duplicate according to the protocol) Start DS Rxmt timer. (3) Receive the retransmitted a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one. PNNI 1.0 5.7.6 PICS 3.14.89 atm98-0466: DS.91 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V213_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and this node is Slave and the packet's DS sequence number is one more than this node's DS sequence number, that the packet is accepted. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_4_r('0'B, '?B, '0'B, tsv_DSno) | (P) | (3) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero. The DS sequence number has to be identical to the one in the last packet sent by the Tester. (2) Send an empty DS packet with the More and Master bit set to one. The DS sequence number is increased with one from the last sent DS packet (This packet should be accepted by the IUT). Start Response timer. (3) Receive a DS packet with the Initialize and Master bit set to zero. The DS sequence number is identical to the one in the last packet sent to the IUT. (The IUT accepted the packet). Stop Response timer. -> Pass PNNI 1.0 5.7.5 PICS 3.14.90 atm98-0466: DS.92 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V214_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that while in Exchanging and this node is Slave and a duplicate Database Summary packet is received, that the last Database Summary packet sent to the Master is retransmitted. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1 !DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_4_r('0'B, '?B, '0'B, tsv_DSno) | | (3) |
| 7 | | Neighb_Peer_PCO_1 !DBSP_T START T_Resp | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (4) |
| 8 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_4_r('0'B, '?B, '0'B, tsv_DSno) | (P) | (5) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero. The DS sequence number has to be identical to the one in the last packet sent by the Tester. (2) Send an empty DS packet with the More and Master bit set to one. The DS sequence number is increased with one from the last sent DS packet (This packet should be accepted by the IUT). Start Response timer. (3) Receive a DS packet with the Initialize and Master bit set to zero. The DS sequence number is identical to the one in the last packet sent to the IUT. (The IUT accepted the packet). Stop Response timer. (4) Re-send the last sent DS packet to the IUT. Start Response timer. (5) Receive DS packet that is identical to last received packet. Stop Response timer. -> Pass PNNI 1.0 5.7.5 PICS 3.14.95 atm98-0466: DS.97 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V215_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T (tcv_PTSE_TYPE := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_type, tcv_PTSE_ID := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_ident, tcv_PTSE_SEQ := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_seq_no, tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (1) |
| 4 | | (tcv_PTSE_SEQ := INT_TO_OCT(OCT_TO_INT(tcv_PTSE _SEQ)+1, 4)) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_5_s('0'B, '1'B, '0'B, tsv_DSno) | | (2) |
| 6 | L1 | Neighb_Peer_PCO_1?PTSP_T [OCT_TO_INT(PTSP_T.ptse_seq. [0].ptse_seq_no) > OCT_TO_INT(tcv_PTSE_SEQ)] CANCEL T_Resp | PTSP_V_2_r | (P) | (3) |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (4) |
| 8 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 9 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize bit set to zero and the Master bit set to one advertising a valid, self-originated PTSE. (2) Send DS packet advertising a newer instance (higher sequence number) of the PTSE in question. Start Reoriginate timer. (3) Receive PTSP containing a newer version of the PTSE in question with a larger sequence-number. Stop Reoriginate timer. -> Pass (4) Ignore the next DBSum packet. PNNI 1.0 5.7.6 PICS 3.14.97 atm98-0466: DS.99 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V215_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that if a PTSE summary is received which is newer than that in the database and is one of this node's self-originated PTSEs and this node still has a valid instance of the PTSE, that a newer version of the PTSE with a larger sequence number is re-originated. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T (tcv_PTSE_TYPE := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_type, tcv_PTSE_ID := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_ident, tcv_PTSE_SEQ := DBSP_T.ptsp_seq.[0].ptse_seq.[0] .ptse_seq_no) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1, tcv_PTSE_SEQ := INT_TO_OCT(OCT_TO_INT(tcv_PTSE _SEQ)+1, 4)) | | | |
| 5 | | Neighb_Peer_PCO_1 !DBSP_T START T_Resp | DBSP_V_5_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | L1 | Neighb_Peer_PCO_1 ?PTSP_T [OCT_TO_INT(PTSP_T.ptse_seq. [0].ptse_seq_no) > OCT_TO_INT(tcv_PTSE_SEQ)] CANCEL T_Resp | PTSP_V_2_r | (P) | (3) |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (4) |
| 8 | | (tsv_DSno := tsv_DSno +1) | | | |
| 9 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ?DBSP_T | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (4) |
| 12 | | (tsv_DSno := tsv_DSno +1) | | | |
| 13 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | |
| 14 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero advertising a valid, self-originated PTSE. (2) Send DS packet advertising a newer instance (higher sequence number) of the PTSE in question. Start Reoriginate timer. (3) Receive PTSP containing a newer version of the PTSE in question with a larger sequence-number. Stop Reoriginate timer. -> Pass (4) Ignore the next DBSum packet. PNNI 1.0 5.7.6 PICS 3.14.97 atm98-0466: DS.99 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V216_M Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_4_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_I_1_s('0'B, '1'B, '0'B, tsv_DSno) | | (2) |
| 8 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_V_3_r | (P) | (3) |
| 9 | | Neighb_Peer_PCO_1 ?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) | DBSP_V_3_r('0'B, '?B, '1'B, tsv_DSno+1) | | (4) |
| 10 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 11 | | GOTO L2 | | | |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send DS packet advertising a PTSE, stating that it was originated by the IUT (by setting the originating nodeid to the node id of the IUT), but the IUT itself does not have. Start Flush timer. (3) Receive PTSP containing an instance of the PTSE in question, which is without content and has the Remaining Lifetime set to ExpiredAge. Stop Flush timer. -> Pass (4) Ignore the next DBSum packet. PNNI 1.0 5.7.6 PICS 3.14.98 atm98-0466: DS.100 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V216_S Group : Bodies/DBSynchronization/Exchanging/ Objective : To verify that if a PTSE summary is received which is one of this node's self-originated PTSE and this node does not have a valid instance of the PTSE, that the PTSE is flushed from the routing domain with the remaining lifetime set to ExpiredAge. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_4_r('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| 8 | | (tsv_DSno := tsv_DSno +1) | | | |
| 9 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_I_1_s('0'B, '1'B, '1'B, tsv_DSno) | | (4) |
| 10 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_V_3_r | (P) | (5) |
| 11 | | Neighb_Peer_PCO_1 ?DBSP_T | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (6) |
| 12 | | (tsv_DSno := tsv_DSno +1) | | | |
| 13 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | |
| 14 | | GOTO L2 | | | |
| 15 | | Neighb_Peer_PCO_1 ?DBSP_T | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (6) |
| 16 | | (tsv_DSno := tsv_DSno +1) | | | |
| 17 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | |
| 18 | | GOTO L2 | | | |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. (2) Respond with a DS packet setting the Initialize bit to zero and the More + Master bits to one. (3) Receive a DS packet with the Initialize, More and Master bits set to zero. (4) Send DS packet advertising a PTSE, stating that it was originated by the IUT (by setting the originating nodeid to the node id of the IUT), but the IUT itself does not have. Start Flush timer. (5) Receive PTSP containing an instance of the PTSE in question, which is without content and has the Remaining Lifetime set to ExpiredAge. Stop Flush timer. -> Pass (6) Ignore next DBSP packet PNNI 1.0 5.7.6 PICS 3.14.98 atm98-0466: DS.100 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V301_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | (P) | (2) |
| Detailed Comments : (1) Send DS packet advertising a PTSE that is not originated by the IUT and which is not expired (IUT will (hopefully) put PTSE on PTSE Request list). Start Request timer. (2) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). Stop Request timer -> Pass PNNI 1.0 5.7.6 PICS 3.14.100 atm98-0466: DS.102b Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V301_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that if a PTSE summary is received which is not in the node's database and that does not satisfy the conditions of PICS 3.14.97 and PICS 3.14.99, that the PTSE is put on the PTSE request list. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | (P) | |
| Detailed Comments : (1) Send DS packet advertising a PTSE that is not originated by the IUT and which is not expired (IUT will (hopefully) put PTSE on PTSE Request list). Start Request timer. (2) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). Stop Request timer -> Pass PNNI 1.0 5.7.6 PICS 3.14.100 atm98-0466: DS.102b Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V302_1_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno+10) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet that is different from the previous one that was sent to the IUT (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|---|---------|----------|
| Test Step Name : DBSync_FSM_V302_1_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno+10) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet that is different from the previous one that was sent to the IUT (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V302_2_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno+10) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103b Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V302_2_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno+10) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103b Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V302_3_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno+10) | | (2) |
| 6 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V302_3_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno+10) | | (2) |
| 5 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.101 atm98-0466: DS.103d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V303_1_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet with the Master bit set opposite as in the last sent packet (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V303_1_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet with the Master bit set opposite as in the last sent packet (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V303_2_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet that with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104b Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V303_2_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet that with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104b Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V303_3_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno-1) | | (2) |
| 6 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V303_3_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that while in Loading, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 5 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.102 atm98-0466: DS.104d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V304_1_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('1'B, '0'B, '0'B, tsv_DSno) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V304_1_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tcv_CRC := PTSE_crc(PTSP_V_3_s)) | | | |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_3_s | | (2) |
| 6 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_4_s,'00000002'O,'00000001'O)) | | | |
| 7 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('1'B, '0'B, '1'B, tsv_DSno) | | (3) |
| 8 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (4) |
| 9 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_4_s,'00000002'O,'00000001'O) | | | (5) |
| 10 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (3) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start PDA timer. (4) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (5) Premature aging of the PTSE sent to the IUT. (6) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V304_2_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('1'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105b Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V304_2_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, that the PTSE Request Rxmt timer is stopped and the PTSE Request list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T START T_ReqRxmt | DBSP_V_4_s('1'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 5 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 6 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (4) |
| 7 | | GOTO L1 | | | |
| Detailed Comments : (1) During synchronization, receive a PTSE Request for the PTSE that was advertised. (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start Request Rxmt timer. (3) Timeout Request Rxmt timer (PTSE Request list is cleared and PTSE Request Rxmt timer was stopped). -> Pass (4) Ignore DBSummary Packets in the Negotiating state. PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105b Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V304_3_M Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('1'B, '0'B, '0'B, tsv_DSno-1) | | (2) |
| 6 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : DBSync_FSM_V304_3_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when in the Loading state and if a Database Summary packet is received that has the Initialize bit set, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('1'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 5 | | +CheckStateNeighbNegotiating | | | (3) |
| Detailed Comments : (1) Receive PTSE request for the PTSE in question (IUT put the PTSE on the PTSE Request list). (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). (3) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.103 atm98-0466: DS.105d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V305_M Group : Bodies/DBSynchronization>Loading/ Objective : To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp, START T_ReqRxmt | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1?PTSE_Req_T READTIMER T_ReqRxmt(tcv_TIME), CANCEL T_ReqRxmt | PTSE_Req_V_2_r | | (2) |
| 5 | | [(tcv_TIME >= tsp_ReqRxmt - tsp_ReqRxmt/4) AND (tcv_TIME <= tsp_ReqRxmt + tsp_ReqRxmt/4)] | | (P) | (3) |
| 6 | | [NOT((tcv_TIME >= tsp_ReqRxmt - tsp_ReqRxmt/4) AND (tcv_TIME <= tsp_ReqRxmt + tsp_ReqRxmt/4))] | | (F) | |
| Detailed Comments : (1) Receive a PTSE Request packet for the PTSE that the Tester advertised. Start timer. (2) Receive another PTSE Request packet for the PTSE that the Tester advertised. Stop timer. (3) Timer value == RequestRxmtInterval (taking jitter into account) (Request Rxmt Timer in the IUT was restarted). -> Pass PNNI 1.0 5.7.7 PICS 3.14.108 atm98-0466: DS.110 Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V305_S Group : Bodies/DBSynchronization>Loading/ Objective : To verify that when a PTSE request packets is sent, that the Request Rxmt Timer is restarted. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp, START T_ReqRxmt | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1?PTSE_Req_T READTIMER T_ReqRxmt(tcv_TIME), CANCEL T_ReqRxmt | PTSE_Req_V_2_r | | (2) |
| 5 | | [(tcv_TIME >= tsp_ReqRxmt - tsp_ReqRxmt/4) AND (tcv_TIME <= tsp_ReqRxmt + tsp_ReqRxmt/4)] | | (P) | (3) |
| 6 | | [NOT((tcv_TIME >= tsp_ReqRxmt - tsp_ReqRxmt/4) AND (tcv_TIME <= tsp_ReqRxmt + tsp_ReqRxmt/4))] | | (F) | |
| Detailed Comments : (1) Receive a PTSE Request packet for the PTSE that the Tester advertised. Start timer. (2) Receive another PTSE Request packet for the PTSE that the Tester advertised. Stop timer. (3) Timer value == RequestRxmtInterval (taking jitter into account) (Request Rxmt Timer in the IUT was restarted). -> Pass PNNI 1.0 5.7.7 PICS 3.14.108 atm98-0466: DS.110 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V306_M | | | | | |
| Group : Bodies/DBSynchronization/Loading/ | | | | | |
| Objective : To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list | | | | | |
| Default : Default_Neighb_Peer_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp, START T_ReqRxmt | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (2) |
| 5 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_2_s, '00000001'O, '00000001'O)) | | | |
| 6 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 7 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_2_s, '00000001'O, '00000001'O) | | | (4) |
| 8 | | Neighb_Peer_PCO_1?PTSP_T | PTSP_V_1_r | | (5) |
| 9 | | GOTO L1 | | | |
| 10 | | Neighb_Peer_PCO_1?PTSE_Ack_T | PTSE_Ack_V_1_r('0000001'O, '00000001'O, tcv_CRC) | | (6) |
| 11 | | GOTO L1 | | | |
| <p>Detailed Comments :</p> <ul style="list-style-type: none"> (1) Receive a PTSE Request packet for the PTSE that the Tester advertised. Start RR timer. (2) Send a PTSP containing the PTSE that was requested by the IUT (PTSE should be removed from the PTSE Request list). (3) Timeout RR timer (PTSE was removed from the PTSE Request list) -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore Link Advertisement by the IUT (6) Ignore PTSE Acknowledgement for the PTSE sent to the IUT <p>PNNI 1.0 5.7.7 PICS 3.14.109 atm98-0466: DS.111 Applies to Master role of the IUT</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : DBSync_FSM_V306_S Group : Bodies/DBSynchronization/Loading/ Objective : To verify that when the proper PTSEs are received in response to requests, that those PTSEs are removed from the PTSE request list Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbLoading_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSE_Req_T CANCEL T_Resp, START T_ReqRxmt | PTSE_Req_V_2_r | | (1) |
| 4 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (2) |
| 5 | | ACTIVATE(Default_Neighb_Peer_PTC_with_Postamble(PTSP_V_2_s, '00000001'O, '00000001'O)) | | | |
| 6 | L1 | ?TIMEOUT T_ReqRxmt | | (P) | (3) |
| 7 | | +PostambleNeighb_Peer_Premature_PTSE_aging(PTSP_V_2_s, '00000001'O, '00000001'O) | | | (4) |
| 8 | | Neighb_Peer_PCO_1?PTSP_T | PTSP_V_1_r | | (5) |
| 9 | | GOTO L1 | | | |
| 10 | | Neighb_Peer_PCO_1?PTSE_Ack_T | PTSE_Ack_V_1_r('0000001'O, '00000001'O, tcv_CRC) | | (6) |
| 11 | | GOTO L1 | | | |
| Detailed Comments : (1) Receive a PTSE Request packet for the PTSE that the Tester advertised. Start RR timer. (2) Send a PTSP containing the PTSE that was requested by the IUT (PTSE should be removed from the PTSE Request list). (3) Timeout RR timer (PTSE was removed from the PTSE Request list) -> Pass PNNI 1.0 5.7.7 PICS 3.14.109 atm98-0466: DS.111 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V401_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Exchanging and the node is Master, if a packet is received that has the DS sequence number equal to this node's own DS sequence number and the M bit is set to zero and this node has already sent its entire database and the PTSE Request List is empty, that the link is advertised in a PTSE. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_V_1_r | (P) | |
| Detailed Comments : (1) Start LinkAdvertise timer (2) Receive a PTSP containing PTSE that advertises the link. Stop LinkAdvertise timer. -> Pass PNNI 1.0 5.7.6 PICS 3.14.87 atm98-0466: DS.89b Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : DBSync_FSM_V401_S | | | | | |
| Group : Bodies/DBSynchronization/Full/ | | | | | |
| Objective : To verify that while in Exchanging and the node is Slave, if a packet is received that has the DS sequence number one more than this node's own DS sequence number, the More bit set to zero and the just transmitted Database Summary packet had the M bit is set to zero and the PTSE Request List is empty, that the link is advertised in a PTSE. | | | | | |
| Default : Default_Neighb_Peer_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_V_1_r | (P) | |
| Detailed Comments : (1) Start LinkAdvertise timer (2) Receive a PTSP containing PTSE that advertises the link. Stop LinkAdvertise timer. -> Pass PNNI 1.0 5.7.6 PICS 3.14.94 atm98-0466: DS.96 Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V402_1_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno+1) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet that is different from the previous one that was sent to the IUT (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V402_1_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno+1) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet that is different from the previous one that was sent to the IUT (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V402_2_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | (tsv_DSno := tsv_DSno +1) | | | |
| 3 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (1) |
| 4 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V402_2_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that is not a duplicate, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno+1) | | (1) |
| 3 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet that is different from the previously sent to the IUT (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.104 atm98-0466: DS.106d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V403_1_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq. [0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq. [0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq. [0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0]. ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts p_NID,tcv_PTSE_ID,tcv_ PTSE_SEQ,tcv_PTSE_RLT, tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet with the Master bit set opposite as in the last sent packet (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V403_1_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet with the Master bit set opposite as in the last sent packet (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V403_2_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | (tsv_DSno := tsv_DSno +1) | | | |
| 3 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno-1) | | (1) |
| 4 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V403_2_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in Full, if a Database Summary packet is received that has an inconsistent MS-bit, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('0'B, '0'B, '0'B, tsv_DSno) | | (1) |
| 3 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet with the Master bit set opposite as in the last sent packet. (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.105 atm98-0466: DS.107d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V404_1_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('1'B, '0'B, '0'B, tsv_DSno) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108a Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : DBSync_FSM_V404_1_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that the Peer Delayed Ack timer is stopped and the Peer Delayed Acks list is cleared. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_V_1_s | | (1) |
| 3 | | Neighb_Peer_PCO_1!DBSP_T START T_PeerDelayedAck | DBSP_V_4_s('1'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 4 | L1 | ?TIMEOUT T_PeerDelayedAck | | (P) | (3) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP_V_2_s | | (4) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r('00000001'0,'00000001'0,tcv_CRC) | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 8 | | GOTO L2 | | | |
| 9 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (5) |
| 10 | | GOTO L1 | | | |
| 11 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (6) |
| 12 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (7) |
| 13 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP containing a PTSE that is not originated by the IUT and is not expired (Acknowledgement is put on the Peer Delayed Acks list). (2) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). Start PDA timer. (3) Timeout PDA timer (Peer Delayed Acks list is cleared and Peer Delayed Ack timer was stopped). -> Pass (4) Premature aging of the PTSE sent to the IUT. (5) Ignore DBSummary Packets in the Negotiating state. (6) Receive an unexpected PTSE from the IUT. (7) Acknowledge the received PTSE. PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108a Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V404_2_M Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_M | | | |
| 2 | | (tsv_DSno := tsv_DSno +1) | | | |
| 3 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('1'B, '0'B, '0'B, tsv_DSno-1) | | (1) |
| 4 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108d Applies to Master role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|--|---------|----------|
| Test Step Name : DBSync_FSM_V404_2_S Group : Bodies/DBSynchronization/Full/ Objective : To verify that while in the Full if a Database Summary packet is received that has the Initialize bit set, that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. Default : Default_Neighb_Peer_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_LinkAdvertized_S | | | |
| 2 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_4_s('1'B, '0'B, '1'B, tsv_DSno) | | (1) |
| 3 | | +CheckStateNeighbNegotiating | | | (2) |
| Detailed Comments : (1) Send DS packet with the Initialize bit set to one. (This will lead to the event DSMismatch in the IUT). (2) CheckDSMismatch PNNI 1.0 5.7.6 PICS 3.14.106 atm98-0466: DS.108d Applies to Slave role of the IUT | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|-----------------|---------|----------|
| Test Step Name : Hello_FSM_Emull(tnid: OCTETSTRING) Group : Bodies/DBSynchronization/Hello_FSM_Emulation/ Objective : Emulate the Hello FSM for Database Synchronization Tests Default : Default_Hello_Emull Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tnid) | | | |
| 2 | | Hello_PCO_1!Hello_T START T_Hello(tsp_THI) | Hello_V_1_s(1) | | |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Resp, START T_Inact(tsp_IF * tsp_THI) | Hello_V_3_r | | (1) |
| 5 | | Hello_Neighb_Peer_CP_1 !AddPort_T | AddPort_V_1 | | (2) |
| 6 | L1 | ?TIMEOUT T_Hello | | | (3a) |
| 7 | | Hello_PCO_1!Hello_T START T_Hello(tsp_THI) | Hello_V_3_s | | (3b) |
| 8 | | GOTO L1 | | | (3c) |
| 9 | | Hello_PCO_1?Hello_T START T_Inact(tsp_IF * tsp_THI) | Hello_V_3_r | | (3d) |
| 10 | | GOTO L1 | | | (3e) |
| 11 | | ?TIMEOUT T_Inact | | (I) | (4) |
| 12 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 13 | | +PostambleHelloAttempt | | | |
| Detailed Comments : (1) Enter Two-Way-Inside (2) Initiate Database Synchronization (3) Repeat and accept Hellos (4) InactivityTimer forces Attempt State and the end of the test case | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V001_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_1_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.2, Table 5-35 PICS 3.15.1 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|------------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V001_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the nodal information from the second node, the IUT floods a PTSP to the first node with the following nodal information of the second node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_1_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_1_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1, tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.2, Table 5-35 PICS 3.15.1 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V002_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_2_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.2, 5.8.5.2.5.6 PICS 3.15.3 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V002_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR and Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CDV (Cell Delay Variation) is present for CBR and Real Time VBR service categories.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_2_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_2_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,2,1,tcv_CRC),'00000002'O,'00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(2,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,2,1,tcv_CRC),'00000002'O,'00000001'O)) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.2, 5.8.5.2.5.6 PICS 3.15.3</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V003_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_3_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.3, 5.8.5.2.5.5 PICS 3.15.4 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V003_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCTD (Maximum Cell Transfer Delay) is present for CBR, Real Time VBR and Non-Real Time VBR service categories.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbFull_S(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_3_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_3_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(3, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.3, 5.8.5.2.5.5 PICS 3.15.4</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V004_First</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_4_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| <p>Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT.</p> <p>PNNI 1.0 5.8.1.1.3.4, 5.8.5.2.5.1 PICS 3.15.5</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V004_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for all service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas Administrative Weight is present for all service categories.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_4_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_4_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,4,1,tcv_CRC), '00000004'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(4, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,4,1,tcv_CRC), '00000004'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.4, 5.8.5.2.5.1 PICS 3.15.5</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V005_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 (Cell Loss Ratio for CLP=0) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_5_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.2 PICS 3.15.6 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V005_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0 is present for CBR, Real Time VBR and Non-Real Time VBR service categories.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_3_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_3_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(3, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.2 PICS 3.15.6</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V006_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_6_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.3 PICS 3.15.7 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V006_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR and Non-Real Time VBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CLR0+1 (Cell Loss Ratio for CLP=0+1) is present for CBR, Real Time VBR and Non-Real Time VBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_3_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_3_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(3, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,3,1,tcv_CRC), '00000003'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.1.3.5, 5.8.5.2.5.3 PICS 3.15.7 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V007_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_7_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.7, 5.8.5.2.5.7 PICS 3.15.8 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V007_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for ABR and UBR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas MaxCR (Maximum Cell Rate) is present for ABR and UBR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_5_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_5_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,5,1,tcv_CRC), '00000005'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2 ? PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(5, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,5,1,tcv_CRC), '00000005'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.1.3.7, 5.8.5.2.5.7 PICS 3.15.8 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V008_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_8_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.8 PICS 3.15.9 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V008_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas AvCR (Available Cell Rate) is present for CBR, Real Time VBR, Non-Real Time VBR and ABR service categories. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_6_s)) | | | |
| 3 | | Neighb_Peer_PCO_2 ! PTSP_T | PTSP_Fldg_V_6_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,6,1,tcv_CRC), '00000006'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(6, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,6,1,tcv_CRC), '00000006'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.1.3.8 PICS 3.15.9 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V009_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_9_r(tsp_TH NID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V009_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_7_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_7_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,7,1,tcv_CRC), '00000007'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(7, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?Terminate Req_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,7,1,tcv_CRC), '00000007'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V010_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_10_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V010_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Real Time VBR service category.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_7_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_7_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,7,1,tcv_CRC),'00000007'O,'00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(7,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,7,1,tcv_CRC),'00000007'O,'00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V011_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_11_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V011_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas CRM (Cell Rate Margin) is present for Non-Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_8_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_8_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,8,1,tcv_CRC), '00000008'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(8, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,8,1,tcv_CRC), '00000008'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.1.3.9, 5.8.5.2.5.8 PICS 3.15.10 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V012_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_12_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V012_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with the outgoing resource availability information for Non-Real Time VBR service category (with optional GCAC information) encapsulated in the Nodal State Parameter IG from the second node, the IUT floods a PTSP to the first node with the outgoing resource availability information of the second node encapsulated in the Nodal State Parameter IG, whereas VF (Variance Factor) is present for Non-Real Time VBR service category.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_8_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_8_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,8,1,tcv_CRC), '00000008'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(8, 1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(96,8,1,tcv_CRC), '00000008'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.1.3.10, 5.8.5.2.5.8 PICS 3.15.11</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V013_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_13_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.3.1 PICS 3.15.14 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|------------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V013_Second</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_9_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_9_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(224,9,1,tcv_CRC), '00000009'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(9, 1, tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(224,9,1,tcv_CRC), '00000009'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.1.3.1 PICS 3.15.14</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V014_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_14_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.3.1 PICS 3.15.14 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V014_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Internal Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Internal Reachable ATM Address IG of the second node, whereas the resource availability information is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_10_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_10_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(224,10,1,tcv_CRC),'0000000A'O,'00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(10,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(224,10,1,tcv_CRC),'0000000A'O,'00000001'O)) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.3.1 PICS 3.15.14 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| <p>Test Step Name : Fldg_FSM_V015_First</p> <p>Group : Bodies/Flooding/</p> <p>Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix.</p> <p>Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1)</p> <p>Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_15_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| <p>Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT.</p> <p>PNNI 1.0 5.8.1.3.2 PICS 3.15.17</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V015_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG, the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the following information is included: - Port ID, - Scope of advertisement, - Address information length, - address information count, - pairs of prefix length and prefix. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_11_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_11_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,11,1,tcv_CRC), '0000000B'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(11,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,11,1,tcv_CRC), '0000000B'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V016_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_16_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V016_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional resource availability information), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the resource availability information is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_12_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_12_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,12,1,tcv_CRC),'0000000C'O,'00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(12,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,12,1,tcv_CRC),'0000000C'O,'00000001'O)) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V017_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_17_r(tsp_T HNID) | (P) | (2) |
| 5 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 6 | | GOTO L1 | | | |
| 7 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive the expected PTSP flooded by the IUT. PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V017_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a PTSP with an Exterior Reachable ATM Address IG (with the optional Transit Network ID), the IUT floods a PTSP to the first node with the Exterior Reachable ATM Address IG of the second node, whereas the Transit Network ID is present. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_13_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_13_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,13,1,tcv_CRC),'0000000D'O,'00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(13,1,tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(256,13,1,tcv_CRC),'0000000D'O,'00000001'O)) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.1.3.2 PICS 3.15.17 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : Fldg_FSM_V018_First Group : Bodies/Flooding/ Objective : To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq . [0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0] .ptse_crc) CANCEL T_FldgL | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(1, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 6 | | START T_Resp(tsp_Resp_Time+tsp_Min PTSEInterval) | | | (4) |
| 7 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_Fldg_V_18_r(tsp_T HNID, 2) | (P) | (5) |
| 8 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 9 | | GOTO L2 | | | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSE flooded by the IUT. (3) Send a PTSE Acknowledgement. (4) Start timer T_Resp with an offset of MinPTSEInterval. (5) Receive the same PTSE flooded by the IUT, with the sequence number incremented. PNNI 1.0 5.8.2.2.4 PICS 3.15.22 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V018_Second | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, during flooding, on receipt of a PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the sequence number of the second PTSE is larger than the sequence number of the previous one, the IUT floods the second PTSE to the first node. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 600))) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (2) |
| 7 | | START T_NoResp(tsp_MinPTSEInterval) | | | (3) |
| 8 | L2 | ?TIMEOUT T_NoResp | | | (4) |
| 9 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(2, 600))) | | | |
| 10 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(2, 600) | | (5) |
| 11 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,2,tcv_CRC), '00000001'O, '00000002'O)) | | | |
| 12 | | START T_Resp | | | |
| 13 | L3 | Neighb_Peer_PCO_2? PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 2,tcv_CRC) | (P) | (6) |
| 14 | L4 | Neighb_Peer_CP_2? TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (7) |
| 15 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,2,tcv_CRC), '00000001'O, '00000002'O) | | | (8) |
| 16 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 17 | | GOTO L4 | | | |
| 18 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 19 | | GOTO L3 | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | | | | | |
|---|-------|-------------------------------------|-----------------|---------|----------|
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 20 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 21 | | GOTO L2 | | | |
| 22 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 23 | | GOTO L1 | | | |
| <p>Detailed Comments : (1) Send a PTSE to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Start a timer with MinPTSEInterval (the min. PTSE update interval). (4) Timeout of MinPTSEInterval. (5) Send the same PTSE with the sequence number incremented. (6) Receive a PTSE Acknowledgement from the IUT. (7) Receive a Terminate request from the MTC. (8) Send a PTSP to the IUT that causes the previously sent PTSE(s). removed from the IUT's database.</p> <p>PNNI 1.0 5.8.2.2.4 PICS 3.15.22</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : Fldg_FSM_V019_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq . [0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0] .ptse_crc) CANCEL T_FldgL | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(1, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 6 | | START T_Resp(tsp_Resp_Time+tsp_Min PTSEInterval) | | | (4) |
| 7 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_Resp | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (5) |
| 8 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 9 | | GOTO L2 | | | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSE flooded by the IUT. (3) Send a PTSE Acknowledgement. (4) Start timer T_Resp with an offset of MinPTSEInterval. (5) Receive the same PTSE flooded by the IUT, with the sequence number incremented. PNNI 1.0 5.8.2.2.4 PICS 3.15.22 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V019_Second | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, when the IUT is in the Full state for the second link, on receipt of a second PTSE from the second node that has the same originating node ID and PTSE identifier as the previously received one, and the remaining lifetime is equal to ExpiredAge, the IUT floods the second PTSE to the first node. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 600))) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (2) |
| 7 | | START T_NoResp(tsp_MinPTSEInterval) | | | (3) |
| 8 | L2 | ?TIMEOUT T_NoResp | | | (4) |
| 9 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, EXPIRED_AGE))) | | | |
| 10 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, EXPIRED_AGE) | | (5) |
| 11 | | START T_Resp | | | |
| 12 | L3 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (6) |
| 13 | L4 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | R | (7) |
| 14 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 15 | | GOTO L4 | | | |
| 16 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 17 | | GOTO L3 | | | |
| 18 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 19 | | GOTO L2 | | | |
| 20 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 21 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSE to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Start a timer with MinPTSEInterval (the min. PTSE update interval). (4) Timeout of MinPTSEInterval. (5) Send the same PTSE with the sequence number incremented. (6) Receive a PTSE Acknowledgement from the IUT. (7) Receive a Terminate request from the MTC. | | | | | |
| PNNI 1.0 5.8.2.2.4 PICS 3.15.22 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V020_First Group : Bodies/Flooding/ Objective : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_FldgL | PTSP_Fldg_V_1_r(tsp_TH NID) | | (2) |
| 5 | | START T_PTSERetr(tsp_PTSERetr + tsp_PTSERetr/4) | | | (3) |
| 6 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_PTSERetr | PTSP_Fldg_V_1_r(tsp_TH NID) | (P) | (4) |
| 7 | | +Fldg_Unexpected(Neighb_Peer_PCO_1) | | | |
| 8 | | GOTO L2 | | | |
| 9 | | ?TIMEOUT T_PTSERetr | | (F) | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_PCO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSP flooded by the IUT. (3) Start the PTSE retransmission timer with 25% offset. (4) Receive the PTSP retransmitted by the IUT. PNNI 1.0 5.8.3.2 PICS 3.15.31 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V020_Second Group : Bodies/Flooding/ Objective : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := | | | |
| 3 | | PTSE_crc(PTSP_Fldg_V_1_s)) | | | |
| 4 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_1_s | | (1) |
| 5 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 6 | L1 | START T_Resp | | | |
| 7 | L2 | Neighb_Peer_PCO_2?PTSE_Ack_T | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (2) |
| 8 | | CANCEL T_Resp | | | |
| 9 | | Neighb_Peer_CP_2?Terminate Req_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 10 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O) | | | (4) |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L2 | | | |
| | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.3.2 PICS 3.15.31 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : Fldg_FSM_V021_First | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq . [0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0] .ptse_crc) CANCEL T_FldgL | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(1, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 6 | | START T_PTSERetr(OCT_TO_INT(tcv_PT SE_RLT)*1000*5/4) | | | (4) |
| 7 | L2 | Neighb_Peer_PCO_1?PTSP_T CANCEL T_PTSERetr | PTSP_Fldg_V_0_r(tsp_TH NID,97,1,1) | (P) | (5) |
| 8 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 9 | | GOTO L2 | | | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSE flooded by the IUT. (3) Send a PTSE Acknowledgement. (4) Start timer with 25% offset. (5) Receive a PTSE without content. | | | | | |
| PNNI 1.0 5.8.3.2 PICS 3.15.33 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|------------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V021_Second | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, when the IUT is in the Full state, in response to the expiration of a PTSE, the IUT floods the PTSE without content to peers. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 120))) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 60) | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'o, '00000001'o)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (2) |
| 7 | | START T_PTSERetr(60*1000*5/4) | | | |
| 8 | L2 | Neighb_Peer_PCO_2?PTSP_T CANCEL T_PTSERetr | PTSP_Fldg_V_0_r(tsp_TH NID,97,1,1) | (P) | (4) |
| 9 | L3 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | R | (3) |
| 10 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 11 | | GOTO L3 | | | |
| 12 | | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 13 | L4 | Neighb_Peer_PCO_2?PTSP_T CANCEL T_PTSERetr | PTSP_Fldg_V_0_r(tsp_TH NID,97,1,1) | (P) | (4) |
| 14 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 15 | | GOTO L4 | | | |
| 16 | | ?TIMEOUT T_PTSERetr | | (F) | |
| 17 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 18 | | GOTO L2 | | | |
| 19 | | ?TIMEOUT T_PTSERetr | | (F) | |
| 20 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 21 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSE to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Receive the PTSE with expired remaining lifetime. | | | | | |
| PNNI 1.0 5.8.3.2 PICS 3.15.33 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|-------------------------------------|----------------------|---------|----------|
| Test Step Name : Fldg_FSM_V022_First Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | L1 | Neighb_Peer_CP_1?TestBodyStartReq_T | TestBodyStartReq_V_1 | | (2) |
| 4 | | START T_NoResp | | | |
| 5 | L2 | ?TIMEOUT T_NoResp | | (P) | (3) |
| 6 | | +Fldg_Unexpected(Neighb_Peer_PCO_1) | | | |
| 7 | | GOTO L2 | | | |
| 8 | | +Fldg_Unexpected(Neighb_Peer_PCO_1) | | | |
| 9 | | GOTO L1 | | | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a request to start the test body execution. (3) No response from the IUT. PNNI 1.0 5.8.3.3 PICS 3.15.39 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|-------------------------------------|--------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V022_Second Group : Bodies/Flooding/ Objective : To verify, when the IUT is in the Full state, on receipt of a PTSE from the second node with invalid PTSE checksum, the IUT complete the processing of PTSE, without sending PTSE Acknowledgement to the second node and without flooding the PTSE to the first node. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | Neighb_Peer_CP_2!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (2) |
| 4 | | START T_NoResp | | | |
| 5 | L1 | ?TIMEOUT T_NoResp | | (P) | (3) |
| 6 | L2 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | R | (4) |
| 7 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 8 | | GOTO L2 | | | |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L1 | | | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Send a PTSE with invalid checksum to the IUT. (3) No response from the IUT. (4) Receive a Terminate request from the MTC. PNNI 1.0 5.8.3.3 PICS 3.15.39 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|------------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V023_First Group : Bodies/Flooding/ Objective : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq . [0].ptse_ttl) CANCEL T_FldgL | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (2) |
| 5 | | START T_PTSERetr(tsp_PTSERetr + tsp_PTSERetr/4) | | | (3) |
| 6 | L2 | Neighb_Peer_PCO_1 ? PTSP_T [OCT_TO_INT(tcv_PTSE_RLT) > OCT_TO_INT(PTSP_T.ptse_seq.[0].ptse_ttl)] CANCEL T_PTSERetr | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (4) |
| 7 | | +Fldg_Unexpected(Neighb_Peer _PCO_1) | | | |
| 8 | | GOTO L2 | | | |
| 9 | | ?TIMEOUT T_PTSERetr | | (F) | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSP flooded by the IUT. (3) Start the PTSE retransmission timer. (4) Receive the PTSP retransmitted by the IUT, with the remaining lifetime decremented by one. PNNI 1.0 5.8.3.4 PICS 3.15.41 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V023_Second Group : Bodies/Flooding/ Objective : To verify, during flooding, when the PTSE retransmission timer expires, the IUT retransmits the PTSE and the PTSE lifetime is decremented. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := | | | |
| 3 | | PTSE_crc(PTSP_Fldg_V_1_s)) | | | |
| 4 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_1_s | | (1) |
| 5 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC),'00000001'O,'00000001'O)) | | | |
| 6 | L1 | START T_Resp | | | |
| 7 | L2 | Neighb_Peer_PCO_2?PTSE_Ack_T | PTSE_Ack_Fldg_V_1_r(1,1,tcv_CRC) | (P) | (2) |
| 8 | | CANCEL T_Resp | | | |
| 9 | | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (3) |
| 10 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC),'00000001'O,'00000001'O)) | | | (4) |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L2 | | | |
| | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.3.4 PICS 3.15.41 | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : Fldg_FSM_V024_First | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) CANCEL T_FldgL | PTSP_Fldg_V_18_r(tsp_T HNID, 1) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(1, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 6 | | START T_Resp(tsp_Resp_Time+tsp_Min PTSEInterval) | | | (4) |
| 7 | L2 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse _seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq .[0].ptse_crc) CANCEL T_Resp | PTSP_Fldg_V_18_r(tsp_T HNID, 2) | (P) | (5) |
| 8 | | Neighb_Peer_PCO_1!PTSE_Ac k_T | PTSE_Ack_Fldg_V_1_s(2, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 9 | | START T_NoResp(tsp_Resp_Time+t sp_MinPTSEInterval) | | | |
| 10 | L3 | ?TIMEOUT T_NoResp | | (P) | |
| 11 | | +Fldg_Unexpected(Neigh b_Peer_PCO_1) | | | |
| 12 | | GOTO L3 | | | |
| 13 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 14 | | GOTO L2 | | | |
| 15 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 16 | | GOTO L1 | | | |
| 17 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSE flooded by the IUT. (3) Send a PTSE Acknowledgement. (4) Start timer T_Resp with an offset of MinPTSEInterval. (5) Receive the same PTSE flooded by the IUT, with the sequence number incremented. | | | | | |
| PNNI 1.0 5.8.3.3, 3b | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V024_Second Group : Bodies/Flooding/ Objective : To verify, during flooding, on receipt of a PTSE instance that is less recent than the the PTSE instance in the database (the sequence number of the received PTSE instance is smaller than the sequence number of the PTSE instance in the database), the IUT floods the database copy encapsulated in a PTSP back to the sender. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 600))) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1,tcv_CRC) | (P) | (2) |
| 7 | | START T_NoResp(tsp_MinPTSEInterval) | | | (3) |
| 8 | L2 | ?TIMEOUT T_NoResp | | | (4) |
| 9 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(2, 600))) | | | |
| 10 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(2, 600) | | (5) |
| 11 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,2,tcv_CRC), '00000001'O, '00000002'O)) | | | |
| 12 | | START T_Resp | | | |
| 13 | L3 | Neighb_Peer_PCO_2? PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 2,tcv_CRC) | (P) | (6) |
| 14 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 600))) | | | |
| 15 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (7) |
| 16 | | START T_Resp | | | |
| 17 | L4 | Neighb_Peer_PCO_2?PTSP_T (tcv_PTSE_RLT :=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) CANCEL T_Resp | PTSP_Fldg_V_18_r(tsp_T HNID, 2) | (P) | (8) |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | | | | | |
|--|-------------------------------------|--|--|---------|----------|
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 18 | L5 | Neighb_Peer_PCO_2!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(2, tsp_THNID, tcv_PTSE_RLT, tcv_CRC_r) | | (9) |
| 19 | | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (10) |
| 20 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_agin g(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97, 1, 2, tcv_CRC), '00000001'O, '00000002'O) | | | (11) |
| 21 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 22 | | GOTO L5 | | | |
| 23 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 24 | | GOTO L4 | | | |
| 25 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 26 | | GOTO L3 | | | |
| 27 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 28 | | GOTO L2 | | | |
| 29 | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | | |
| 30 | GOTO L1 | | | | |
| <p>Detailed Comments : (1) Send a PTSE to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Start a timer with MinPTSEInterval (the min. PTSE update interval). (4) Timeout of MinPTSEInterval. (5) Send the same PTSE with the sequence number incremented. (6) Receive a PTSE Acknowledgement from the IUT. (7) Send the first PTSE again. (8) Receive the more recent database copy of the PTSE. (9) Send a PTSE Acknowledgement. (10) Receive the PTSE copy with the higher sequence number. (11) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database.</p> <p>PNNI 1.0 5.8.3.3, 3b</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : Fldg_FSM_V025_First Group : Bodies/Flooding/ Objective : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq .[0].ptse_ttl, tcv_CRC:=PTSP_T.ptse_seq.[0].p tse_crc) CANCEL T_FldgL | PTSP_Fldg_V_1_r(tsp_TH NID) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSP_T | PTSP_Fldg_V_14_s(1, OCT_TO_INT(tcv_PTSE_RL T)-1) | | (3) |
| 6 | | START T_NoResp | | | |
| 7 | L2 | ?TIMEOUT T_NoResp | | (P) | |
| 8 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 9 | | GOTO L2 | | | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSP flooded by the IUT. (3) Send the same PTSE with decremented remaining lifetime back to the IUT. PNNI 1.0 5.8.3.3, 4a | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|------------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V025_Second Group : Bodies/Flooding/ Objective : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is contained on the receiving link's Peer Retransmission List, the IUT completes the processing of PTSE without further flooding the PTSE. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_1_s)) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_1_s | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, 1, tcv_CRC) | (P) | (2) |
| 7 | L2 | Neighb_Peer_CP_2?Terminate Req_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 8 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O) | | | (4) |
| 9 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 10 | | GOTO L2 | | | |
| 11 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 12 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSP to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Receive a Terminate request from the MTC. (4) Send a PTSP to the IUT that causes the previously sent PTSE(s) removed from the IUT's database. PNNI 1.0 5.8.3.3, 4a | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : Fldg_FSM_V026_First Group : Bodies/Flooding/ Objective : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_First | | | |
| 2 | | Neighb_Peer_CP_1!NeighbFullInd_T | NeighbFullInd_V_1 | | (1) |
| 3 | | START T_FldgL | | | |
| 4 | L1 | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_RLT:=PTSP_T.ptse_seq . [0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0] .ptse_crc) CANCEL T_FldgL | PTSP_Fldg_V_1_r(tsp_TH NID) | (P) | (2) |
| 5 | | Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_1_s(1, tsp_THNID,tcv_PTSE_RLT ,tcv_CRC_r) | | (3) |
| 6 | | START T_NoResp(tsp_Resp_Time+tsp_M inPTSEInterval) | | | (4) |
| 7 | L2 | ?TIMEOUT T_NoResp | | (P) | (5) |
| 8 | | +Fldg_Unexpected(Neighb_Pe er_PCO_1) | | | |
| 9 | | GOTO L2 | | | |
| 10 | | +Fldg_Unexpected(Neighb_Peer_P CO_1) | | | |
| 11 | | GOTO L1 | | | |
| 12 | | ?TIMEOUT T_FldgL | | (F) | |
| Detailed Comments : (1) Send an indication to MTC that the Full state is reached. (2) Receive a PTSP flooded by the IUT. (3) Send a PTSE Acknowledgement. (4) Start timer T_NoResp with an offset of MinPTSEInterval. (5) No response from the IUT prior to timeout PNNI 1.0 5.8.3.3, 4b | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---------------------------------|---------|----------|
| Test Step Name : Fldg_FSM_V026_Second | | | | | |
| Group : Bodies/Flooding/ | | | | | |
| Objective : To verify, during flooding, on receipt of a PTSE instance that is the same as the PTSE instance in the database, and the PTSE is not contained on the receiving link's Peer Retransmission List, the IUT acknowledges the PTSE and completes the processing of PTSE without further flooding the PTSE. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_Fldg_Second | | | |
| 2 | | (tcv_CRC := PTSE_crc(PTSP_Fldg_V_14_s(1, 600))) | | | |
| 3 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (1) |
| 4 | | ACTIVATE(DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O)) | | | |
| 5 | | START T_Resp | | | |
| 6 | L1 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, tcv_CRC) | (P) | (2) |
| 7 | | START T_NoResp(tsp_MinPTSEInterval) | | | (3) |
| 8 | L2 | ?TIMEOUT T_NoResp | | | (4) |
| 9 | | Neighb_Peer_PCO_2!PTSP_T | PTSP_Fldg_V_14_s(1, 600) | | (5) |
| 10 | | START T_Resp | | | |
| 11 | L3 | Neighb_Peer_PCO_2?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_Fldg_V_1_r(1, tcv_CRC) | (P) | (6) |
| 12 | L4 | Neighb_Peer_CP_2?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | | (7) |
| 13 | | +PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO_2, PTSP_Fldg_V_0_s(97,1,1,tcv_CRC), '00000001'O, '00000001'O) | | | (8) |
| 14 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 15 | | GOTO L4 | | | |
| 16 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 17 | | GOTO L3 | | | |
| 18 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 19 | | GOTO L2 | | | |
| 20 | | +Fldg_Unexpected(Neighb_Peer_PCO_2) | | | |
| 21 | | GOTO L1 | | | |
| Detailed Comments : (1) Send a PTSE to the IUT. (2) Receive a PTSE Acknowledgement from the IUT. (3) Start a timer with MinPTSEInterval (the min. PTSE update interval). (4) Timeout of MinPTSEInterval. (5) Send the same PTSE instance again. (6) Receive a PTSE Acknowledgement from the IUT. (7) Receive a Terminate request from the MTC. | | | | | |

Continued on next page

Continued from previous page

| Test Step Dynamic Behaviour | |
|--|--|
| <p>Detailed Comments : ... (8) Send a PTSP to the IUT that causes the previously sent PTSE(s). removed from the IUT's database. PNNI 1.0 5.8.3.3, 4a</p> | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| <p>Test Step Name : Fldg_Unexpected(Neighb_Peer_PCO:R_SAP) Group : Bodies/Flooding/Fldg_Unexpected/ Objective : Default : Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_PCO?PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (1) |
| 2 | | Neighb_Peer_PCO!PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts_p_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (2) |
| <p>Detailed Comments : (1) Receive an unexpected PTSE from the IUT. (2) Acknowledge the received PTSE.</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| <p>Test Step Name : Fldg_Unexpected_Second Group : Bodies/Flooding/Fldg_Unexpected/ Objective : Default : Comments :</p> | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_PCO_2 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (1) |
| 2 | | Neighb_Peer_PCO_2 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts_p_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (2) |
| <p>Detailed Comments : (1) Receive an unexpected PTSE from the IUT. (2) Acknowledge the received PTSE.</p> | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : TwoLinks_Hello_FSM_Emul(tnid,taesa: OCTETSTRING;Hello_CP, Hello_Neighb_Peer_CP:CP; Hello_PCO:R_SAP) | | | | | |
| Group : Bodies/Flooding/Hello_FSM_Emulation/ | | | | | |
| Objective : Emulate the Hello FSM | | | | | |
| Default : DefTwoLinks_Hello_Emul(Hello_CP, Hello_Neighb_Peer_CP, Hello_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_HelloAttempt(tnid, taesa, Hello_CP, Hello_Neighb_Peer_CP, Hello_PCO) | | | |
| 2 | | Hello_PCO!Hello_T START T_Hello(tsp_THI) | Hello_V_8_s(1) | | |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO?Hello_T CANCEL T_Resp, START T_Inact(tsp_IF * tsp_THI) | Hello_V_3_r | | (1) |
| 5 | | Hello_Neighb_Peer_CP !AddPort_T | AddPort_V_1 | | (2) |
| 6 | L1 | ?TIMEOUT T_Hello | | | (3a) |
| 7 | | Hello_PCO!Hello_T START T_Hello(tsp_THI) | Hello_V_9_s | | (3b) |
| 8 | | GOTO L1 | | | (3c) |
| 9 | | Hello_PCO?Hello_T START T_Inact(tsp_IF * tsp_THI) | Hello_V_3_r | | (3d) |
| 10 | | GOTO L1 | | | (3e) |
| 11 | | ?TIMEOUT T_Inact | | (I) | (4) |
| 12 | | Hello_Neighb_Peer_CP !DropPort_T | DropPort_V_1 | | |
| 13 | | +PostTwoLinks_HelloAttempt(Hello_PCO) | | | |
| Detailed Comments : (1) Enter Two-Way-Inside (2) Initiate Database Synchronization (3) Repeat and accept Hellos (4) InactivityTimer forces Attempt State and the end of the test case | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : CheckStateNeighbNegotiating | | | | | |
| Group : CheckState/ | | | | | |
| Objective : Check that a DS packet is sent with the DS sequence number incremented, the DS Rxmt timer is restarted and the Negotiating state is entered. | | | | | |
| Default : Default_Neighb_Peer_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | START T_Resp | | | (1) |
| 2 | L1 | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_2_r('1'B, '1'B, '1'B, tsv_DSno+1) | | (2) |
| 3 | | START T_DSRxmt | | | (2) |
| 4 | | Neighb_Peer_PCO_1?DBSP_T READTIMER T_DSRxmt(tcv_TIME), CANCEL T_DSRxmt | DBSP_V_2_r('1'B, '1'B, '1'B, tsv_DSno+1) | (P) | (3) |
| 5 | | [(tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4)] | | (P) | (4) |
| 6 | | [NOT((tcv_TIME >= tsp_DSRxmt - tsp_DSRxmt/4) AND (tcv_TIME <= tsp_DSRxmt + tsp_DSRxmt/4))] | | (F) | |
| 7 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0] .ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0] .ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0] .ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].pt se_crc) | PTSP_V_4_r | | (5) |
| 8 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts p_NID,tcv_PTSE_ID,tcv_ PTSE_SEQ,tcv_PTSE_RLT, tcv_CRC_r) | | (6) |
| 9 | | GOTO L1 | | | |
| Detailed Comments : | | | | | |
| (1) Start Response timer. | | | | | |
| (2) Receive a Database Summary packet with the Initialize, More and Master bits set to one and a sequence number that is incremented with one compared to the sequence number in the last packet. Start DS Rxmt timer. | | | | | |
| (3) Receive a copy of the last received DS packet. Stop DS Rxmt timer. | | | | | |
| (4) Timer value == DSRxmtInterval (taking jitter into account) (DS Rxmt timer was restarted) -> Pass | | | | | |
| (5) Receive an unexpected PTSE from the IUT. | | | | | |
| (6) Acknowledge the received PTSE. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : CheckStateHelloAttempt | | | | | |
| Group : CheckState/ | | | | | |
| Objective : Check the IUT state Attempt | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_3_s | | (1) |
| 2 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | (P) | |
| 3 | | ?TIMEOUT T_Resp | | (F) | |
| Detailed Comments : (1) Attempt is the only state that the event Two Way Inside recieved triggers a Hello in response. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : CheckStateHelloOWI | | | | | |
| Group : CheckState/ | | | | | |
| Objective : Check the IUT state One Way Inside at the end of the test | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_2_s | | (1) |
| 2 | | ?TIMEOUT T_NoResp | | (P) | |
| 3 | | Hello_PCO_1?OTHERWISE CANCEL T_NoResp | | (F) | |
| Detailed Comments : The state One Way Inside is the only state that on receiving a One Way Inside Hello it does not send any Hello back. (1) Send a Hello with the remote node ID filed and remote Port ID field equal to zero and with the peer group ID matching the IUT's peer group ID (one way inside received event). | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : CheckStateHelloOWO | | | | | |
| Group : CheckState/ | | | | | |
| Objective : Check the IUT state One Way Outside at the end of the test | | | | | |
| Default : Default_Hello_PTC | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_4_s | | (1) |
| 2 | | ?TIMEOUT T_NoResp | | (P) | |
| 3 | | Hello_PCO_1?OTHERWISE CANCEL T_NoResp | | (F) | |
| Detailed Comments : The state One Way Outside is the only state that on receiving a One Way Outside Hello it does not send in response any Hello back. (1) Send a Hello with the remote node ID filed and remote Port ID field equal to zero and with the peer group ID matching the IUT's peer group ID (One Way Outside received event). | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : CheckStateHelloTWI Group : CheckState/ Objective : Check the IUT state Two Way Inside at the end of the test Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T START T_NoResp | Hello_V_3_s | | (1) |
| 2 | | ?TIMEOUT T_NoResp | | (P) | |
| 3 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_2_s | | (2) |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | | (3) |
| 5 | | ?TIMEOUT T_Resp | | (F) | |
| 6 | | Hello_PCO_1?OTHERWISE CANCEL T_NoResp | | (F) | |
| Detailed Comments : The state Two Way Inside in the only state that on receiving a Two Way Inside Hello it does not send back any Hello, and remains in the same state. Then on sending a One Way Inside Hello it sends in response a Two Way Inside Hello. <ol style="list-style-type: none"> (1) Send a Hello with the remote node ID and remote port ID fields equal the iut's node id and port id and the peer group ID equal to the iut's peer group ID. (two way inside). Start the timer T_NoResp. (2) The T_NoResp timer has expired and the tester sends a Hello with the remote node id field and remote port id field equal to zero and with the peer group id mathcing the iut's peer group id (one way inside received event). start the T_Resp timer. (3) Receive in response a Hello with the remote node id and remote port id fields equals to the testers node id and port id respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : CheckStateHelloTWO Group : CheckState/ Objective : Check the IUT states Two Way Outside at the end of the test Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T START T_Resp | Hello_V_4_s | | (1) |
| 2 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | (P) | (2) |
| 3 | | ?TIMEOUT T_Resp | | (F) | |
| Detailed Comments : The state Two Way Outside is the only one receiving a One Way Outside Hello, send a Hello back in response. <ol style="list-style-type: none"> (1) Send a Hello with the remote node ID filed and remote Port ID field equal to zero and with the peer group ID matching the IUT's peer group ID (one way outside received event). (2) Receive a Hello in response with the remote node id field and remote port id field equal to the tester's node id and port id respectively. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|----------------------------|-----------------------------------|---------|----------|
| Test Step Name : PostambleDBSync_PTCs Group : Postambles/ Objective : Postamble for all DBSynchronization Test Cases Default : Default_DBSync_MTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | ?DONE() | | R | (1) |
| 2 | | ?DONE(Neighb_Peer_FSM_1) | | | |
| 3 | | Hello_CP_1 !TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (2) |
| 4 | | ?DONE() | | R | |
| Detailed Comments : (1) Hello_FSM terminated due to exceptional cases and requested also the Neighb_Peer_FSM to terminate (2) Neighb_Peer_FSM terminated and requests the Hello_FSM to terminate This Postamble is used in all DBSynchronization Test Cases | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : PostambleHelloAttempt Group : Postambles/ Objective : To bring the IUT back to Attempt Default : Default_Hello_PTC Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO_1!Hello_T | Hello_I_1_s | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | R | (2) |
| Detailed Comments : (1) Bring the IUT back to Attempt sending a Hello with a mismatch in the node ID field. (2) Receive a Hello with the remote node ID and remote Port ID equal to zero and with the Version field equal to the Newest version field. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|------------------------------------|---------|----------|
| Test Step Name : PostambleNeighb_Peer_Premature_PTSE_aging(PTSP:PTSP_T; id,seq:OCTETSTRING) | | | | | |
| Group : Postambles/ | | | | | |
| Objective : Premature aging of the PTSE sent to the IUT | | | | | |
| Default : Default_Neighb_Peer_PTC_with_Postamble(PTSP,id,seq) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP | | (1) |
| 2 | L1 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r(id,seq, tcv_CRC) | R | |
| 3 | | +Fldg_Unexpected(Neighb_Peer_PCO_1) | | | |
| 4 | | GOTO L1 | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_1_r | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1!OTHERWISE CANCEL T_Resp | | F | |
| 8 | | ?TIMEOUT T_Resp | | F | |
| Detailed Comments : (1) Send a self-originated PTSE without content and with Remaining Lifetime set to ExpiredAge to remove it from the topology database of the IUT. (2) Ignore DBSummary Packets in the Negotiating state. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---------------------------------|-----------------------------------|---------|----------|
| Test Step Name : PostTwoLinks_Fldg | | | | | |
| Group : Postambles/PostTwoLinks/ | | | | | |
| Objective : Postamble for all Flooding Test Cases | | | | | |
| Default : | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | ?DONE(Neighb_Peer_FSM_1) | | | |
| 2 | | Neighb_Peer_CP_2!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (1) |
| 3 | | ?DONE(Neighb_Peer_FSM_2) | | | |
| 4 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (2) |
| 5 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | (3) |
| 6 | | ?DONE() | | R | |
| Detailed Comments : (1) Request the second Neighb_Peer_FSM to terminate (2) Request the first Hello_FSM to terminate (3) Request the second Hello_FSM to terminate This Postamble is used in Flooding Test Cases | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|------------------------------------|-----------------|---------|----------|
| Test Step Name : PostTwoLinks_HelloAttempt(Hello_PCO:R_SAP) | | | | | |
| Group : Postambles/PostTwoLinks/ | | | | | |
| Objective : To bring the IUT back to Attempt | | | | | |
| Default : | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_PCO!Hello_T | Hello_I_1_s | | (1) |
| 2 | | START T_Hello | | | |
| 3 | | Hello_PCO?Hello_T CANCEL T_Hello | Hello_V_4_r | R | (2) |
| 4 | | Hello_PCO?OTHERWISE CANCEL T_Hello | | F | |
| 5 | | ?TIMEOUT T_Hello | | F | |
| Detailed Comments : (1) Bring the IUT back to Attempt sending a Hello with a mismatch in the node ID field. (2) Receive a Hello with the remote node ID and remote Port ID equal to zero and with the Version field equal to the Newest version field. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--------------------------------|---------|----------|
| Test Step Name : PostTwoLinks_Neighb_Peer_Premature_PTSE_aging(Neighb_Peer_PCO:R_SAP; PTSP:PTSP_T; id,seq:OCTETSTRING) | | | | | |
| Group : Postambles/PostTwoLinks/ | | | | | |
| Objective : Premature aging of the PTSE sent to the IUT | | | | | |
| Default : | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_PCO!PTSP_T START T_Resp | PTSP | | (1) |
| 2 | L1 | Neighb_Peer_PCO?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r(id,seq,tcv_CRC) | R | |
| 3 | | +Fldg_Unexpected(Neighb_Peer_PCO) | | | |
| 4 | | GOTO L1 | | | |
| 5 | | Neighb_Peer_PCO?OTHERWISE CANCEL T_Resp | | F | |
| 6 | | ?TIMEOUT T_Resp | | F | |
| Detailed Comments : (1) Send a self-originated PTSE without content and with Remaining Lifetime set to ExpiredAge to remove it from the topology database of the IUT. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : PreTwoLinks_PreambleInit(tnid, taesa:OCTETSTRING) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To perform some initialization. | | | | | |
| Default : | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | (tcv_TNID := tnid, tcv_TAESA := taesa) | | | |
| Detailed Comments : | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|--------------------|
| Test Step Name : PreTwoLinks_HelloDown(tnid,taesa:OCTETSTRING; Hello_CP, Hello_Neighb_Peer_CP:CP; Hello_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To start the IUT Hello-FSM | | | | | |
| Default : DefTwoLinks_Hello_PTC_Preamble(Hello_CP, Hello_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_PreambleInit(tnid,taesa) | | | |
| 2 | | (tcv_LINK_UP := HelloLinkUp()) | | | |
| 3 | | [tcv_LINK_UP] | | | Link up successful |
| 4 | | [NOT(tcv_LINK_UP)] | | I | Link up failed |
| Detailed Comments : The IUT receives a Link_Up event from the lower layers and this starts the FSM. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|-----------------|---------|----------|
| Test Step Name : PreTwoLinks_HelloAttempt(tnid,taesa:OCTETSTRING; Hello_CP, Hello_Neighb_Peer_CP:CP; Hello_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT in the Hello state Attempt. | | | | | |
| Default : DefTwoLinks_Hello_PTC_Preamble(Hello_CP, Hello_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_HelloDown(tnid,taesa, Hello_CP, Hello_Neighb_Peer_CP, Hello_PCO) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int) CANCEL T_Resp | Hello_V_1_r | (P) | (2) |
| Detailed Comments : (1) Starts the IUT Hello-FSM (2) Receive the first Hello, and store in the local variables tsv_R_PID and tsv_R_HI the values of the port ID and Hello Interval fields respectively, from the received Hello. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-----------------|---------|----------|
| Test Step Name : PreTwoLinks_NeighbNegotiating(tnid,taesa:OCTETSTRING; Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the Neighbouring Peer state Negotiating | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_PreambleInit(tnid,taesa) | | | |
| 2 | | Hello_Neighb_Peer_CP?AddPort_T | AddPort_V_1 | (P) | (1) |
| Detailed Comments : (1) Hello FSM is in state Two-Way-Inside. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : PreTwoLinks_NeighbExchanging_M(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the Neighboring Peer state Exchanging as Master | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbNegotiating(tsp_TLNID, tsp_TAESA, Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO ?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_1_r | | (2) |
| 4 | | Neighb_Peer_PCO !DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (3) |
| 5 | | ?TIMEOUT T_Resp | | (I) | |
| Detailed Comments : (1) Hello FSM is in state Two-Way-Inside. (2) Tester accepts initial DS number and becomes Slave (3) Tester acknowledges | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|----------------------|---------|----------|
| Test Step Name : PreTwoLinks_NeighbExchanging_S(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the Neighbouring Peer state Exchanging as Slave | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbNegotiating(tsp_THNID, TAESA_2ND, Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | |
| 2 | | (tsv_DSno:= DSno_init()) | | | |
| 3 | | START T_Resp | | | |
| 4 | | Neighb_Peer_PCO?DBSP_T CANCEL T_Resp | DBSP_V_1_r | | (1) |
| 5 | | Neighb_Peer_PCO!DBSP_T | DBSP_V_1_s(tsv_DSno) | | (2) |
| Detailed Comments : (1) Receive an empty DS packet with the Initialize, More and Master bit set to one (first init packet). (2) Send an empty DS packet with the Initialize, More and Master bit set to one, with a unique serial number (resulting in the NegotiationDone event in the IUT). | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|--|---------|----------|
| Test Step Name : PreTwoLinks_NeighbFull_M(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the Neighboring Peer state Full as Master | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbExchanging_M(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | (P) | (1a) |
| 4 | | Neighb_Peer_PCO!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_4_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO!DBSP_T | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a empty DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--|--|---------|----------|
| Test Step Name : PreTwoLinks_NeighbFull_S(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the Neighboring Peer state Full as Slave. | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbExchanging_S(Neighb_Peer_CP, Hello_Neighb_Peer_CP, Neighb_Peer_PCO) | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO!DBSP_T | DBSP_V_3_s('0'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. (2) Respond with a DS packet setting the Initialize + Master bits to zero and the More bit to one. (3) Receive an empty DS packet with the Initialize, More and Master bits set to zero. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : PreTwoLinks_Fldg_First | | | | | |
| Group : Preambles/PreTwoLinks/ | | | | | |
| Objective : To bring the IUT into the state before fooding begins | | | | | |
| Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbFull_M(Neighb_Peer_CP_1, Hello_Neighb_Peer_CP_1, Neighb_Peer_PCO_1) | | | |
| 2 | | START T_FldgS | | | |
| 3 | | Neighb_Peer_PCO_1 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) CANCEL T_FldgS | PTSP_V_4_r | | (1) |
| 4 | | Neighb_Peer_PCO_1 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT,tcv_CRC_r) | | (2) |
| 5 | | ?TIMEOUT T_FldgS | | | |
| Detailed Comments : (1) Receive a PTSE originated by the IUT. (2) Acknowledge the PTSE. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : PreTwoLinks_Fldg_Second Group : Preambles/PreTwoLinks/ Objective : To bring the IUT into the state before fooding begins Default : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreTwoLinks_NeighbFull_S(Neighb_Peer_CP_2, Hello_Neighb_Peer_CP_2, Neighb_Peer_PCO_2) | | | |
| 2 | | START T_FldgS | | | |
| 3 | | Neighb_Peer_PCO_2 ? PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_4_r | | (1) |
| 4 | | CANCEL T_FldgS Neighb_Peer_PCO_2 ! PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(tsp_NID, tcv_PTSE_ID, tcv_PTSE_SEQ, tcv_PTSE_RLT, tcv_CRC_r) | | (2) |
| 5 | | ?TIMEOUT T_FldgS | | | |
| Detailed Comments : (1) Receive a PTSE originated by the IUT. (2) Acknowledge the PTSE. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|-----------------|---------|----------|
| Test Step Name : PreambleHelloAttempt(tnid:OCTETSTRING) Group : Preambles/ Objective : To bring the IUT in the Hello state Attempt. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloDown(tnid) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Hello_PCO_1?Hello_T (tsv_R_PID := Hello_T.o_port, tsv_R_HI := Hello_T.hello_int) | Hello_V_1_r | (P) | (2) |
| CANCEL T_Resp Detailed Comments : (1) Starts the IUT Hello-FSM (2) Receive the first Hello, and store in the local variables tsv_R_PID and tsv_R_HI the values of the port ID and Hello Interval fields respectively, from the received Hello. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------|-----------------|---------|--------------------|
| Test Step Name : PreambleHelloDown(tnid:OCTETSTRING) Group : Preambles/ Objective : To start the IUT Hello-FSM Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleInit(tnid) | | | |
| 2 | | (tcv_LINK_UP := HelloLinkUp()) | | | |
| 3 | | [tcv_LINK_UP] | | | Link up successful |
| 4 | | [NOT(tcv_LINK_UP)] | | I | Link up failed |
| Detailed Comments : The IUT receives a Link_Up event from the lower layers and this starts the FSM. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : PreambleHelloOWI Group : Preambles/ Objective : To bring the IUT in the Hello state One-Way-Inside. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_2_s | | (2) |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | | (3) |
| Detailed Comments : (1) Bring the IUT to the state Attempt (2) Send a Hello with the remote Node ID and remote Port ID field equal to zero, and with the Peer Group ID matching the IUT's Peer group ID. (one way inside received event) (3) Receive a hello in response with the remote Node ID and remote Port ID field equal to the Tester's Node ID and Port ID. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : PreambleHelloOWO Group : Preambles/ Objective : To bring the IUT in the Hello state One-Way-Outside. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_4_s | | (2) |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_7_r | | (3) |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote Node ID and remote Port ID field equal to zero, and with the Peer Group ID not matching the IUT's Peer group ID. (one way outside received event) (3) Receive a response with the remote Node ID and remote Port ID field equal to the Tester's Node ID and Port ID. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|--------------------------------------|-----------------|---------|----------|
| Test Step Name : PreambleHelloTWI Group : Preambles/ Objective : To bring the IUT in the Hello state Two-Way-Inside. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_3_s | | (2) |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T CANCEL T_Resp | Hello_V_3_r | | (3) |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote Node ID and remote Port ID field equal to the IUT's Node ID and Port ID, and with the Peer Group ID matching the IUT's Peer group ID. (two way inside received event) (3) Receive a response with the remote Node ID and remote Port ID field equal to the Tester's Node ID and Port ID. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : PreambleHelloTWO Group : Preambles/ Objective : To bring the IUT in the Hello state Two-Way-Outside. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_7_s | | (2) |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T (tcv_NHL_NO := OCT_TO_INT(Hello_T.ig.nhl.seq_ num), tcv_ULIA_NO := OCT_TO_INT(Hello_T.ig.ulia.seq_ num)) CANCEL T_Resp | Hello_V_7_r | | (3) |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote node ID and remote port ID fields equal to the iut's node id and port id respectively. And a hierarchy list and all outgoing ig's. (Two Way Outside received event) (3) Receive a response with the remote Node ID and remote Port ID field equal to the Tester's Node ID and Port ID. Save the Nodal Hierarchy List Sequence Number and the Uplink Information Attribut Sequence Number. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|-----------------------|-----------------|---------|----------|
| Test Step Name : PreambleInit(tnid:OCTETSTRING) Group : Preambles/ Objective : To perform some initialization. Default : Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | (tcv_TNID := tnid) | | | |
| Detailed Comments : | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : PreambleNeighbExchanging_M Group : Preambles/ Objective : To bring the IUT into the Neighboring Peer state Exchanging as Master Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_TLN ID) | | | (1) |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_1_r | | (2) |
| 4 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | (3) |
| 5 | | ?TIMEOUT T_Resp | | (I) | |
| Detailed Comments : (1) Hello FSM is in state Two-Way-Inside. (2) Tester accepts initial DS number and becomes Slave (3) Tester acknowledges | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|----------------------|---------|----------|
| Test Step Name : PreambleNeighbExchanging_S Group : Preambles/ Objective : To bring the IUT into the Neighbouring Peer state Exchanging as Slave Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbNegotiating(tsp_THN ID) | | | |
| 2 | | (tsv_DSno:= DSno_init()) | | | |
| 3 | | START T_Resp | | | |
| 4 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_1_r | | (1) |
| 5 | | Neighb_Peer_PCO_1 !DBSP_T | DBSP_V_1_s(tsv_DSno) | | (2) |
| Detailed Comments : (1) Receive an empty DS packet with the Initialize, More and Master bit set to one (first init packet). (2) Send an empty DS packet with the Initialize, More and Master bit set to one, with a unique serial number (resulting in the NegotiationDone event in the IUT). | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|--|---------|----------|
| Test Step Name : PreambleNeighbFull_LinkAdvertized_M Group : Preambles/ Objective : To bring the IUT into the Neighboring Peer state Full as Master and acknowledge the link advertizement Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_1_r | | (1) |
| 4 | | CANCEL T_Resp Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts p_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT, tcv_CRC_r) | | (2) |
| Detailed Comments : (1) Receive a PTSE originated by the IUT. (2) Acknowledge the PTSE. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|--|---------|----------|
| Test Step Name : PreambleNeighbFull_LinkAdvertized_S Group : Preambles/ Objective : To bring the IUT into the Neighboring Peer state Full as Slave and acknowledge the link advertizement Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbFull_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?PTSP_T (tcv_PTSE_ID:=PTSP_T.ptse_seq.[0].ptse_id, tcv_PTSE_SEQ:=PTSP_T.ptse_seq.[0].ptse_seq_no, tcv_PTSE_RLT:=PTSP_T.ptse_seq.[0].ptse_ttl, tcv_CRC_r:=PTSP_T.ptse_seq.[0].ptse_crc) | PTSP_V_1_r | | (1) |
| 4 | | CANCEL T_Resp Neighb_Peer_PCO_1!PTSE_Ack_T | PTSE_Ack_Fldg_V_2_s(ts p_NID,tcv_PTSE_ID,tcv_PTSE_SEQ,tcv_PTSE_RLT, tcv_CRC_r) | | (2) |
| Detailed Comments : (1) Receive a PTSE originated by the IUT. (2) Acknowledge the PTSE. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|---|---------|----------|
| Test Step Name : PreambleNeighbFull_M Group : Preambles/ Objective : To bring the IUT into the Neighboring Peer state Full as Master Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_M | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | (P) | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_4_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | (2) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a empty DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : PreambleNeighbFull_S Group : Preambles/ Objective : To bring the IUT into the Neighboring Peer state Full as Slave. Default : Default_Neighb_Peer_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | L1 | +PreambleNeighbExchanging_S | | | |
| 2 | | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '0'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. (2) Respond with a DS packet setting the Initialize + Master bits to zero and the More bit to one. (3) Receive an empty DS packet with the Initialize, More and Master bits set to zero. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|---|---|---------|----------|
| Test Step Name : PreambleNeighbLoading_M | | | | | |
| Group : Preambles/ | | | | | |
| Objective : To bring the IUT into the Neighboring Peer state Loading as Master | | | | | |
| Default : Default_Neighb_Peer_PTC_Preamble | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_M | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '1'B, tsv_DSno+1) | (P) | (1a) |
| 4 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '0'B, tsv_DSno) | | |
| 5 | | GOTO L1 | | | |
| 6 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_4_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (1b) |
| 7 | | (tcv_CRC := PTSE_crc(PTSP_V_1_s)) | | | |
| 8 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_4_s('0'B, '1'B, '0'B, tsv_DSno) | | (2) |
| 9 | | Neighb_Peer_PCO_1?DBSP_T (tsv_DSno:= OCT_TO_INT(DBSP_T.ds_seq_no)) CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '1'B, tsv_DSno+1) | | (3) |
| 10 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '0'B, '0'B, tsv_DSno) | | (4) |
| Detailed Comments : Two possibilities: (1a) Receive a non-empty DS packet with the Initialize bit set to zero and the More + Master bit set to one. Respond with a DS packet setting the Initialize and Master bit to zero, the More bit to one and copying the DS sequence number from the received packet. (Re-)start Response timer. Goto L1 (1b) Receive a DS packet with the Initialize and More bits set to zero and the Master bit set to one. (2) Send a non-empty DS packet setting the Initialize, More and Master bit to zero and copying the DS sequence number from the received packet. (3) Receive an empty DS packet. (4) Send an empty DS packet. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|--|---|---------|----------|
| Test Step Name : PreambleNeighbLoading_S | | | | | |
| Group : Preambles/ | | | | | |
| Objective : To bring the IUT into the Neighboring Peer state Loading as Slave. | | | | | |
| Default : Default_Neighb_Peer_PTC_Preamble | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleNeighbExchanging_S | | | |
| 2 | L1 | START T_Resp | | | |
| 3 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_3_r('0'B, '1'B, '0'B, tsv_DSno) | | (1) |
| 4 | | (tsv_DSno := tsv_DSno +1) | | | |
| 5 | | Neighb_Peer_PCO_1!DBSP_T | DBSP_V_3_s('0'B, '1'B, '1'B, tsv_DSno) | | (2) |
| 6 | | GOTO L1 | | | |
| 7 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (3) |
| 8 | | (tsv_DSno := tsv_DSno +1, tcv_CRC := PTSE_crc(PTSP_V_1_s)) | | | |
| 9 | | Neighb_Peer_PCO_1!DBSP_T START T_Resp | DBSP_V_4_s('0'B, '0'B, '1'B, tsv_DSno) | | (4) |
| 10 | | Neighb_Peer_PCO_1 ?DBSP_T CANCEL T_Resp | DBSP_V_2_r('0'B, '0'B, '0'B, tsv_DSno) | | (5) |
| Detailed Comments : (1) Receive a non-empty DS packet with the Initialize and Master bits set to zero and the More bit set to one. Increase the DS sequence number with one. | | | | | |
| (2) Respond with a DS packet setting the Initialize bit to zero and the More + Master bits to one. | | | | | |
| (3) Receive an empty DS packet with the Initialize, More and Master bits set to zero. Increase the DS sequence number with one. | | | | | |
| (4) Send a non-empty DS packet setting the Initialize and More bit to zero. | | | | | |
| (5) Receive an empty DS packet with the Initialize, More and Master bits set to zero. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|--|-------|----------------------------------|-----------------|---------|----------|
| Test Step Name : PreambleNeighbNegotiating(tnid:OCTETSTRING) | | | | | |
| Group : Preambles/ | | | | | |
| Objective : To bring the IUT into the Neighbouring Peer state Negotiating | | | | | |
| Default : Default_Neighb_Peer_PTC_Preamble | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleInit(tnid) | | | |
| 2 | | Hello_Neighb_Peer_CP_1?AddPort_T | AddPort_V_1 | (P) | (1) |
| Detailed Comments : (1) Hello FSM is in state Two-Way-Inside. | | | | | |

| Test Step Dynamic Behaviour | | | | | |
|---|-------|---|-----------------|---------|----------|
| Test Step Name : PreambleHelloCO Group : Preambles/ Objective : To bring the IUT in the Hello state Common-Outside. Default : Default_Hello_PTC_Preamble Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | +PreambleHelloAttempt(tsp_THNID) | | | (1) |
| 2 | | Hello_PCO_1!Hello_T | Hello_V_6_s | | (2) |
| 3 | | START T_Resp | | | |
| 4 | | Hello_PCO_1?Hello_T (tcv_NHL_NO := OCT_TO_INT(Hello_T.ig.nhl.seq_ num), tcv_ULIA_NO := OCT_TO_INT(Hello_T.ig.ulia.seq_ num)) CANCEL T_Resp | Hello_V_7_r | | (3) |
| Detailed Comments : (1) Bring the IUT to the state Attempt. (2) Send a Hello with the remote node ID and remote port ID fields equal to IUT's Node ID and Port ID respectively, and the Peer Group ID does not match with the IUT's one. And includes a hierarchy list with a common higher level peer group in the second level, between the IUT and the Tester. (Common Outside received event) (3) Receive a response with the remote Node ID and remote Port ID field equal to the Tester's Node ID and Port ID. Save the Nodal Hierarchy List Sequence Number and the Uplink Information Attribute Sequence Number. | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|---------------------------------|-----------------------------------|---------|----------|
| Default Name : DefTwoLinks_Fldg_MTC_one | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_Flooding test component configurations. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | ?DONE(Neighb_Peer_FSM_1) | | (F) | |
| 2 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | |
| 3 | | ?DONE() | | R | |
| 4 | | ?DONE>Hello_FSM_1) | | (F) | |
| 5 | | Neighb_Peer_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 6 | | ?DONE() | | R | |
| 7 | | Hello_CP_1?OTHERWISE | | (F) | |
| 8 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 9 | | ?DONE() | | R | |
| 10 | | Neighb_Peer_CP_1?OTHERWISE | | (F) | |
| 11 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 12 | | ?DONE() | | R | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|---------------------------------|-----------------------------------|---------|----------|
| Default Name : DefTwoLinks_Fldg_MTC_two | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_Flooding test component configurations. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | ?DONE(Neighb_Peer_FSM_1) | | (F) | |
| 2 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | |
| 3 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 4 | | ?DONE() | | R | |
| 5 | | ?DONE(Neighb_Peer_FSM_2) | | (F) | |
| 6 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | |
| 7 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 8 | | ?DONE() | | R | |
| 9 | | ?DONE>Hello_FSM_1) | | (F) | |
| 10 | | Neighb_Peer_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 11 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 12 | | ?DONE() | | R | |
| 13 | | ?DONE>Hello_FSM_2) | | (F) | |
| 14 | | Neighb_Peer_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 15 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 16 | | ?DONE() | | R | |
| 17 | | Hello_CP_1?OTHERWISE | | (F) | |
| 18 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 19 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 20 | | ?DONE() | | R | |
| 21 | | Neighb_Peer_CP_1?OTHERWISE | | (F) | |
| 22 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 23 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 24 | | ?DONE() | | R | |
| 25 | | Hello_CP_2?OTHERWISE | | (F) | |
| 26 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 27 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 28 | | ?DONE() | | R | |
| 29 | | Neighb_Peer_CP_2?OTHERWISE | | (F) | |
| 30 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 31 | | Hello_CP_2!TerminateReq_T | TerminateReq_V_1(MTC_F AILURE) | | |
| 32 | | ?DONE() | | R | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|--|--------------------------------|---------|----------|
| Default Name : DefTwoLinks_Hello_Emul(Hello_CP, Hello_Neighb_Peer_CP:CP; Hello_PCO:R_SAP) | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC and Neighb_Peer_FSM. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP?TerminateReq_T CANCEL T_Hello | TerminateReq_V_1(DBSyn c_DONE) | | |
| 2 | | +LocalPostambleHelloAttempt | | | |
| 3 | | Hello_CP?TerminateReq_T | TerminateReq_V_1(MTC_F ALLURE) | | |
| 4 | | Hello_Neighb_Peer_CP!DropPort_T | DropPort_V_1 | | |
| 5 | | +LocalPostambleHelloAttempt | | | |
| 6 | | Hello_Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(?) | | |
| 7 | | +LocalPostambleHelloAttempt | | | |
| 8 | | Hello_CP?OTHERWISE | | (I) | |
| 9 | | Hello_Neighb_Peer_CP!DropPort_T | DropPort_V_1 | | |
| 10 | | +LocalPostambleHelloAttempt | | | |
| 11 | | Hello_Neighb_Peer_CP?OTHERWISE | | (I) | |
| 12 | | Hello_Neighb_Peer_CP!DropPort_T | DropPort_V_1 | | |
| 13 | | +LocalPostambleHelloAttempt | | | |
| 14 | | Hello_PCO?OTHERWISE | | (I) | |
| 15 | | Hello_Neighb_Peer_CP!DropPort_T | DropPort_V_1 | | |
| 16 | | +LocalPostambleHelloAttempt | | | |
| 17 | | ?TIMEOUT | | (I) | |
| 18 | | Hello_Neighb_Peer_CP!DropPort_T | DropPort_V_1 | | |
| 19 | | +LocalPostambleHelloAttempt | | | |
| | | LocalPostambleHelloAttempt | | | |
| 20 | | Hello_PCO!Hello_T | Hello_I_1_s | | |
| 21 | | START T_Hello | | | |
| 22 | | Hello_PCO?Hello_T CANCEL T_Hello | Hello_V_4_r | R | |
| 23 | | Hello_PCO?OTHERWISE CANCEL T_Hello | | F | |
| 24 | | ?TIMEOUT T_Hello | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-------------------------------|---------|----------|
| Default Name : DefTwoLinks_Hello_PTC_Preamble(Hello_CP:CP;Hello_PCO:R_SAP) | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | |
| 2 | | +LocalPostambleHelloAttempt | | | |
| 3 | | Hello_CP?OTHERWISE | | (I) | |
| 4 | | +LocalPostambleHelloAttempt | | | |
| 5 | | Hello_PCO?OTHERWISE | | (I) | |
| 6 | | +LocalPostambleHelloAttempt | | | |
| 7 | | ?TIMEOUT | | (I) | |
| 8 | | +LocalPostambleHelloAttempt | | | |
| 9 | | LocalPostambleHelloAttempt | | | |
| 9 | | Hello_PCO!Hello_T | Hello_I_1_s | | |
| 10 | | START T_Hello | | | |
| 11 | | Hello_PCO?Hello_T CANCEL T_Hello | Hello_V_4_r | R | |
| 12 | | Hello_PCO?OTHERWISE CANCEL T_Hello | | F | |
| 13 | | ?TIMEOUT T_Hello | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|---------------------------------|-------------------------------|---------|----------|
| Default Name : DefTwoLinks_Neighb_Peer_PTC(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_Neighb_Peer_CP?DropPort_T | DropPort_V_1 | R | |
| 2 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | R | |
| 3 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(DBSync_DONE) | R | |
| 4 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1>Hello_DONE) | R | |
| 5 | | Neighb_Peer_CP?OTHERWISE | | F | |
| 6 | | Hello_Neighb_Peer_CP?OTHERWISE | | F | |
| 7 | | Neighb_Peer_PCO?OTHERWISE | | F | |
| 8 | | ?TIMEOUT | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|----------------------------------|---------------------------------|---------|----------|
| Default Name : DefTwoLinks_Neighb_Peer_PTC_Preamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP) | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_Neighb_Peer_CP ?DropPort_T | DropPort_V_1 | R | |
| 2 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(MTC_F AILLURE) | R | |
| 3 | | Neighb_Peer_CP?OTHERWISE | | I | |
| 4 | | Hello_Neighb_Peer_CP?OTHERWISE | | I | |
| 5 | | Neighb_Peer_PCO?OTHERWISE | | I | |
| 6 | | ?TIMEOUT | | I | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|--|---------------------------------|---------|----------|
| Default Name : DefTwoLinks_Neighb_Peer_PTC_with_Postamble(Neighb_Peer_CP, Hello_Neighb_Peer_CP:CP; Neighb_Peer_PCO:R_SAP; PTSP:PTSP_T; id,seq:OCTETSTRING) | | | | | |
| Group : DefTwoLinks/ | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_Neighb_Peer_CP?DropPort_T | DropPort_V_1 | R | |
| 2 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(MTC_F AILLURE) | R | |
| 3 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(Hello_DONE) | R | |
| 4 | | Neighb_Peer_CP?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | |
| 5 | | +Premature_PTSE_aging | | | |
| 6 | | Neighb_Peer_CP?OTHERWISE | | (F) | |
| 7 | | +Premature_PTSE_aging | | | |
| 8 | | Hello_Neighb_Peer_CP?OTHERWISE | | (F) | |
| 9 | | +Premature_PTSE_aging | | | |
| 10 | | Neighb_Peer_PCO?OTHERWISE | | (F) | |
| 11 | | +Premature_PTSE_aging | | | |
| 12 | | ?TIMEOUT | | (F) | |
| 13 | | +Premature_PTSE_aging | | | |
| 14 | | Premature_PTSE_aging | | | |
| 14 | | Neighb_Peer_PCO!PTSP_T START T_Resp | PTSP | | (1) |
| 15 | | Neighb_Peer_PCO?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r(id,seq, tcv_CRC) | R | |
| 16 | | Neighb_Peer_PCO?OTHERWISE CANCEL T_Resp | | F | |
| 17 | | ?TIMEOUT T_Resp | | F | |
| Detailed Comments : (1) Send a self-originated PTSE without content and with Remaining Lifetime set to ExpiredAge to remove it from the topology database of the IUT. | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|---------------------------------|-------------------------------|---------|----------|
| Default Name : Default_DBSync_MTC | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs in tcc_DBSync test component configurations. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP_1?OTHERWISE | | (F) | |
| 2 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | |
| 3 | | ?DONE() | | R | |
| 4 | | Neighb_Peer_CP_1?OTHERWISE | | (F) | |
| 5 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | |
| 6 | | ?DONE() | | R | |
| 7 | | ?TIMEOUT | | (F) | |
| 8 | | Hello_CP_1!TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | |
| 9 | | ?DONE() | | R | |
| 10 | | ?DONE(Hello_FSM_1) | | | |
| 11 | | Neighb_Peer_CP_1!TerminateReq_T | TerminateReq_V_1(Hello_DONE) | | |
| 12 | | ?DONE() | | R | |
| 13 | | ?DONE() | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|------------------------------------|---------|----------|
| Default Name : Default_Hello_Emull | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC and Neighb_Peer_FSM. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP_1?TerminateReq_T | TerminateReq_V_1(DBSyn c_DONE) | | |
| 2 | | +LocalPostambleHelloAttempt | | | |
| 3 | | Hello_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_F AILLURE) | | |
| 4 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 5 | | +LocalPostambleHelloAttempt | | | |
| 6 | | Hello_CP_1?OTHERWISE | | (I) | |
| 7 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 8 | | +LocalPostambleHelloAttempt | | | |
| 9 | | Hello_Neighb_Peer_CP_1?OTHERWISE | | (I) | |
| 10 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 11 | | +LocalPostambleHelloAttempt | | | |
| 12 | | Hello_PCO_1?OTHERWISE | | (I) | |
| 13 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 14 | | +LocalPostambleHelloAttempt | | | |
| 15 | | ?TIMEOUT | | (I) | |
| 16 | | Hello_Neighb_Peer_CP_1 !DropPort_T | DropPort_V_1 | | |
| 17 | | +LocalPostambleHelloAttempt | | | |
| 18 | | LocalPostambleHelloAttempt | | | |
| 19 | | Hello_PCO_1!Hello_T START T_Hello | Hello_I_1_s | | |
| 20 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | R | |
| 21 | | Hello_PCO_1?OTHERWISE | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|---------------------------|-------------------------------|---------|----------|
| Default Name : Default_Hello_MTC | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and unforeseen termination of PTCs. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP_1?OTHERWISE | TerminateReq_V_1(MTC_FAILURE) | (F) | |
| 2 | | Hello_CP_1!TerminateReq_T | | | |
| 3 | | ?DONE() | | R | |
| 4 | | ?TIMEOUT | | (F) | |
| 5 | | Hello_CP_1!TerminateReq_T | | | |
| 6 | | ?DONE() | | R | |
| 7 | | ?DONE() | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | | |
|---|-------|-----------------------------|-------------------------------|---------|----------|-------------|
| Default Name : Default_Hello_PTC | | | | | | |
| Group : | | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | | | | | | |
| Comments : | | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments | |
| 1 | | Hello_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | | |
| 2 | | +LocalPostambleHelloAttempt | | | | |
| 3 | | Hello_CP_1?OTHERWISE | | | | (F) |
| 4 | | +LocalPostambleHelloAttempt | | | | |
| 5 | | Hello_PCO_1?OTHERWISE | | | | (F) |
| 6 | | +LocalPostambleHelloAttempt | | | | |
| 7 | | ?TIMEOUT | | | | (F) |
| 8 | | +LocalPostambleHelloAttempt | | | | |
| 9 | | LocalPostambleHelloAttempt | | | | |
| 10 | | Hello_PCO_1!Hello_T | | | | Hello_I_1_s |
| 11 | | START T_Hello | | | | |
| 12 | | Hello_PCO_1?Hello_T CANCEL | | | | Hello_V_4_r |
| | | T_Hello | | | | |
| | | Hello_PCO_1?OTHERWISE | | F | | |
| Detailed Comments : | | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|---------------------------------------|-------------------------------|---------|----------|
| Default Name : Default_Hello_PTC_Preamble | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | | |
| 2 | | +LocalPostambleHelloAttempt | | | |
| 3 | | Hello_CP_1?OTHERWISE | | (I) | |
| 4 | | +LocalPostambleHelloAttempt | | | |
| 5 | | Hello_PCO_1?OTHERWISE | | (I) | |
| 6 | | +LocalPostambleHelloAttempt | | | |
| 7 | | ?TIMEOUT | | (I) | |
| 8 | | +LocalPostambleHelloAttempt | | | |
| 9 | | LocalPostambleHelloAttempt | | | |
| 9 | | Hello_PCO_1!Hello_T | Hello_I_1_s | | |
| 10 | | START T_Hello | | | |
| 11 | | Hello_PCO_1?Hello_T CANCEL T_Hello | Hello_V_4_r | R | |
| 12 | | Hello_PCO_1?OTHERWISE | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|-----------------------------------|-------------------------------|---------|----------|
| Default Name : Default_Neighb_Peer_PTC | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | R | |
| 2 | | Neighb_Peer_CP_1?TerminateReq_T | TerminateReq_V_1(Hello_DONE) | R | |
| 3 | | Hello_Neighb_Peer_CP_1?DropPort_T | DropPort_V_1 | R | |
| 4 | | Neighb_Peer_CP_1?OTHERWISE | | F | |
| 5 | | Hello_Neighb_Peer_CP_1?OTHERWISE | | F | |
| 6 | | Neighb_Peer_PCO_1?OTHERWISE | | F | |
| 7 | | ?TIMEOUT | | F | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|---|-------|------------------------------------|-------------------------------|---------|----------|
| Default Name : Default_Neighb_Peer_PTC_Preamble | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers, the TerminateReq from the MTC, and the DropPort from the Hello FSM Emulation. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Hello_Neighb_Peer_CP_1 ?DropPort_T | DropPort_V_1 | R | |
| 2 | | Neighb_Peer_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | R | |
| 3 | | Neighb_Peer_CP_1?OTHERWISE | | I | |
| 4 | | Hello_Neighb_Peer_CP_1?OTHERWISE | | I | |
| 5 | | Neighb_Peer_PCO_1?OTHERWISE | | I | |
| 6 | | ?TIMEOUT | | I | |
| Detailed Comments : | | | | | |

| Default Dynamic Behaviour | | | | | |
|--|-------|--|--------------------------------|---------|----------|
| Default Name : Default_Neighb_Peer_PTC_with_Postamble(PTSP:PTSP_T; id,seq:OCTETSTRING) | | | | | |
| Group : | | | | | |
| Objective : Handles the OTHERWISE statement, the expiration of timers and the TerminateReq from the MTC. | | | | | |
| Comments : | | | | | |
| Nr | Label | Behaviour Description | Constraints Ref | Verdict | Comments |
| 1 | | Neighb_Peer_CP_1?TerminateReq_T | TerminateReq_V_1(MTC_FAILURE) | R | |
| 2 | | Neighb_Peer_CP_1?TerminateReq_T | TerminateReq_V_1>Hello_DONE) | R | |
| 3 | | Hello_Neighb_Peer_CP_1?DropPort_T | DropPort_V_1 | R | |
| 4 | | Neighb_Peer_CP_1?OTHERWISE | | (F) | |
| 5 | | +Premature_PTSE_aging | | | |
| 6 | | Hello_Neighb_Peer_CP_1?OTHERWISE | | (F) | |
| 7 | | +Premature_PTSE_aging | | | |
| 8 | | Neighb_Peer_PCO_1?OTHERWISE | | (F) | |
| 9 | | +Premature_PTSE_aging | | | |
| 10 | | ?TIMEOUT | | (F) | |
| 11 | | +Premature_PTSE_aging | | | |
| 12 | | Premature_PTSE_aging Neighb_Peer_PCO_1!PTSP_T START T_Resp | PTSP | | (1) |
| 13 | L1 | Neighb_Peer_PCO_1?PTSE_Ack_T CANCEL T_Resp | PTSE_Ack_V_1_r(id,seq,tcv_CRC) | R | |
| 14 | | Neighb_Peer_PCO_1?DBSP_T | DBSP_V_1_r | | (2) |
| 15 | | GOTO L1 | | | |
| 16 | | Neighb_Peer_PCO_1?OTHERWISE CANCEL T_Resp | | F | |
| 17 | | ?TIMEOUT T_Resp | | F | |
| Detailed Comments : (1) Send a self-originated PTSE without content and with Remaining Lifetime set to ExpiredAge to remove it from the topology database of the IUT. (2) Ignore DBSummary Packets in the Negotiating state. | | | | | |

9 Protocol Implementation eXtra Information for Testing (PIXIT) Proforma for the ATS for PNNI Routing

IUT

Name:

Version:

Machine Configuration:

Operating System Identification:

IUT Identification:

PICS Reference for IUT:

Limitations of the IUT:

Instructions for Completing the PIXIT Proforma

The Protocol Implementation eXtra Information for Testing (PIXIT) is a document which is to be completed by the user submitting an implementation for testing. It contains information related to the Implementation Under Test (IUT) and the test environment which is required by the IUT. The PIXIT information is beyond that provided by the Protocol Implementation Conformance Statement (PICS).

This section contains the PIXIT proforma which meets the requirements of this test suite. The test suite developer and/ or test laboratory may provide additional questions to this proforma, as needed.

The user should fill in all sections that are applicable to the implementation, and leave blank those that are not. This is done by either checking a ballot box, or by writing an answer in the provided space. In some cases the type of value to be provided is specified (e.g., a decimal number) along with the proper units (e.g., seconds). When the user is required to check a ballot box and more than one alternative value is listed, the first listed alternative value shall be considered to be the default value for the corresponding PIXIT parameter, unless otherwise indicated in the corresponding Question or Value fields. For a more detailed meaning for each of the possible value choices, the user may refer to ATM Forum "Private Network-Network Interface Specification Version 1.0 (PNNI 1.0)" [af-pnni-0055.000].

Addressing

| Item | Question | Value | Answer |
|------|--|-------------|--------|
| A.1 | Node ID of the IUT. | Octetstring | |
| A.2 | ATM End System Address of the IUT. | Octetstring | |
| A.3 | Peer Group ID of the IUT. | Octetstring | |
| A.4 | Next higher level LGN ID of the IUT. | Octetstring | |
| A.5 | Next higher level LGN Peer Group ID of the IUT. | Octetstring | |
| A.6 | Node ID of the Tester, smaller than the IUT s node ID but in the same PNNI level. | Octetstring | |
| A.7 | Node ID of the Tester, higher than the IUT s node ID but in the same PNNI level. | Octetstring | |
| A.8 | ATM End System Address of the Tester. | Octetstring | |
| A.9 | Peer Group ID of the Tester different from the IUTs Peer Group ID. (used for outside messages) | Octetstring | |
| A.10 | Port ID of the Tester. | Octetstring | |
| A.11 | Next higher level LGN ID of the Tester. | Octetstring | |
| A.12 | Next higher level LGN Peer Group ID of the Tester. | Octetstring | |

System Parameter

| Item | Question | Value | Answer |
|------|---|-------------------------------------|--------|
| SP.1 | Newest version supported by the IUT. | Integer Default: 1 | |
| SP.2 | Oldest version supported by the IUT. | Integer Default: 1 | |
| SP.3 | Newest version supported by the Tester. | Integer Default: 1 | |
| SP.4 | Newest version supported by the Tester. | Integer Default: 1 | |
| SP.5 | Default value for the configured Aggregation Token. | Octetstring Default: '00000000'O | |
| SP.6 | The number of HelloIntervals allowed to pass without receiving a Hello, before the Hello FSM declares that a link is down. (Inactivity factor). | Integer Default: 5 | |

Timer

| Item | Question | Value | Answer |
|------|--|---|--------|
| T.1 | Maximum time in seconds allowed for a response by the IUT. Must be longer than the HoldDown timer plus a 25% jitter. | Unit: seconds Default: 2 | |
| T.2 | Maximum time in seconds where no response is allowed by the IUT. Must be longer than the Hold down timer plus a 25% jitter and shorter than the HelloInterval time minus a 25% jitter. | Unit: seconds Default: 3 | |
| T.3 | Hello Interval time of the tester. | Unit: milliseconds Default: 1500 | |
| T.4 | Minimum interval between successive Hello transmissions (HoldDown timer). | Unit: milliseconds Default: 1000 (Minimum: 100) | |
| T.5 | The interval at which unacknowledged PTSEs will be retransmitted (PTSERetransmissionInterval). | Unit: milliseconds Default: 5000 | |
| T.6 | This is the time in seconds between reoriginations of a self originated PTSE in the absence of triggered updates (PTSERefreshInterval). | Unit: milliseconds Default: 1800 | |
| T.8 | The minimum number of milliseconds between transmissions of delayed PTSE acknowledgment packets (PeerDelayedAck timer). | Unit: milliseconds Default: 1000 | |
| T.8 | The amount of time, in milliseconds, a node waits before it sends the previous Database Summary packet again (DSRxmtInterval). | Unit: milliseconds Default: 5000 | |
| T.9 | The amount of time, in milliseconds, before a node sends a new PTSE Request Packet requesting PTSEs of the last PTSE Request Packet that have not been received yet (RequestRxmtInterval). | Unit: milliseconds Default: 5000 | |
| T.10 | Minimal time interval between PTSE updates. That is, a node is limited to updating any particular PTSE no more than once every MinPTSEInterval seconds. | Unit: seconds Default: 1 | |