ETSI Plugtests Test Plan V1.0 (2020-11)

2^{nd} and 3^{rd} mWT SDN Plugtests Event
Sophia Antipolis, France

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex – FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 – NAF 742 C
Association à but non 2tandardi enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Contents

List of Figures

List of Tables

1 Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents assist the user with regard of the Plugtests preparation.

[1] ACTN Framework https://tools.ietf.org/html/rfc8453

[2] I2RS Topology Model: https://tools.ietf.org/html/rfc8345

[3] TE Topology Model: https://datatracker.ietf.org/doc/draft-ietf-teas-yang-te-topo/

[4] MW Topology Model: https://datatracker.ietf.org/doc/draft-ye-ccamp-mw-topo-yang/

[5] Ethernet Topology Model: https://datatracker.ietf.org/doc/draft-zheng-ccamp-client-topo-yang/

[6] Ethernet Service Model: https://datatracker.ietf.org/doc/draft-zheng-ccamp-client-signal-yang/

[7] Restconf protocol: https://tools.ietf.org/html/rfc8040

[8] YANG Module Library: https://tools.ietf.org/html/rfc7895

[9] LLDP Standard: https://standards.ieee.org/standard/802_1AB-2016.html

[10] IEEE802.1Q https://standards.ieee.org/standard/802_1Q-2018.html

[11] Reference algorithm to calculate the inter-domain-plug-id value based on LLDP: https://wiki.plugtests.net/mWT-Plugtests2/index.php/Network_Information

Notes

- For the standards in draft status, the version mentioned in Annex C shall be used as a baseline. If the most recent version is published too late for a Participant to be implemented for this Plugtest, it may be accepted that the Domain Controller complies with an earlier version.

2.2 Informative references

[12] Applicability of YANG models for ACTN https://datatracker.ietf.org/doc/draft-ietf-teas-actn-yang/

[13] Plugtest Wiki: https://wiki.plugtests.net/mWT-Plugtests2/index.php/Main_Page

[14] Code Forge repository: https://forge.etsi.org/gitlab/sdn/mwt

[15] Working documents referenced in this document: https://wiki.plugtests.net/mWT-Plugtests2/index.php/Testing_Information

[16] Transport Northbound Interface Applicability Statement https://tools.ietf.org/html/draft-ietf-ccamp-transport-nbi-app-statement-10#page-34

3 Definitions and Abbreviations

3.1 Definitions

See Paragraph 6.1Reference ACTN Architecture” for the definitions of elements, and interfaces introduced in the ACTN framework.

1. The inter-domain-plug-id attribute is often referred to as “plug-id” for brevity
2. Some definitions of elements, interfaces and acronyms introduced by IETF’s ACTN framework are reported in Paragraph 6.1 Reference ACTN Architecture.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

DC	Domain Controller
DUT	Device Under Test
GE	Gigabit Ethernet
IFS	Interoperability Feature Statement
mmW	Millimetre wave
LLDP	Link Layer Discovery Protocol
MBH	Mobile BackHaul
MW	Microwave
NBI	Northbound Interface
NE	Network Element
NMS	Network Management System
SBI	Southbound Interface
SDN	Software Defined Network
TD	Test Description
TGA	Traffic Generator/Analyzer

4 Conventions

4.1 Common Rules

- The Domain Controller and its physical domain shall be provided as a unified bundle. A bundle may be formed by more than one component provider, but they shall be in agreement and jointly represent a single point of responsibility towards the Plugtests.

5 Scope

5.1 General Considerations

All the tests executed as part of this Plugtests™ concern the MPI interface.

The tests are executed by means of Postman scripts, which exercise the MPI according to the specifications detailed in Annexes.

Domain specific settings and information (e.g. the root of the data store, specific naming conventions for vendor-specific attribute values etc.) are configured and managed by means of environment variables: when testing each domain, the corresponding settings are recalled by setting the Postman script execution environment to the corresponding domain's one.

Use of environment variables can also be useful in order to segregate any vendor-confidential information from documents that are going to be accessible by the public.

5.1.1 Overall Test Structure

The Configurations to be tested are described in Clause 7.

The test suites cover:

a. General compliance to the Restconf [7] protocol, and implementation of the correct IETF YANG Data Models [Appendix C];
b. Network Topology Discovery (Microwave, Ethernet and Service);
c. Creation, validation and deletion of L2 services as commonly found in MW Mobile Backhaul networks. This includes a physical test showing actual L2 traffic flows by means of a test instrument;
d. Compliance to the use of the LLDP protocol with the TE topology in order to allow automatic discovery of inter-domain links.

5.2 Compliance Testing

These tests are strictly related to the compliance of the MPI to the specifications.

5.2.1 Network Discovery

These tests retrieve the Microwave Topology, Ethernet Topology and the Ethernet Service Topology information from the PNCs.

The discovered topology information are compared to the expected response for the Configuration under test as per Annex C JSON Code, and a Pass/Fail outcome is generated and recorded.

5.2.2 E2E L2 Service Provisioning

These tests create, validate and delete L2 services. The L2 VPN is based on the QinQ use of an S-VLAN tag to identify the physical site, and a C-VLAN tag to identify the specific port/service.

The EtherType values for C-VLAN and S-VLAN are 0x8100 and 0x88a8 respectively, as defined in [10], see Figure 1 IEEE802.1Q EtherType Allocations.

Tag Type	Name	Value
Customer VLAN Tag	IEEE 802.1Q Tag Protocol EtherType (802.1QTagType)	81-00
Service VLAN Tag or Backbone VLAN Tag	IEEE 802.1Q Service Tag EtherType (802.1QSTagType)	88-a8
Backbone Service Instance Tag	IEEE 802.1Q Backbone Service Instance Tag EtherType (802.1QITagType)	88-e7

Figure 1 IEEE802.1Q EtherType Allocations

The relevant YANG definitions in draft-zheng-ccamp-client-signal-yang are:

```
identity classify-c-vlan {
  base eth-vlan-tag-classify;
  description
    "Classify 802.1Q Customer VLAN tag.
    Only C-tag type is accepted";
}


identity classify-s-vlan {
    base eth-vlan-tag-classify;
    description
      "Classify 802.1Q Service VLAN (QinQ) tag.
       Only S-tag type is accepted";
  }
```

5.3 Interoperability Testing

5.3.1 LLDP Use for Inter-Domain Link Discovery

This test suite is intended to validate the proposed use of LLDP information to calculate the inter-domain-plug-id ("plug-id" in the following) value defined in the *te-topology* Data Model.

Based on the LLDP packets transmitted and received over the wire, the reference plug-id value is calculated by means of a shared, reference version of the algorithm described in **Error! Reference source not found.**a separate document.

The values of plug-id calculated by each domain are compared to the reference value, providing the outcome of the compliance test for each domain.

6 Architecture

6.1 Reference ACTN Architecture

Architecture and terminology used in this document refer to the ACTN framework defined by IETF in RFC8453 [1] and other relevant RFCs/drafts.

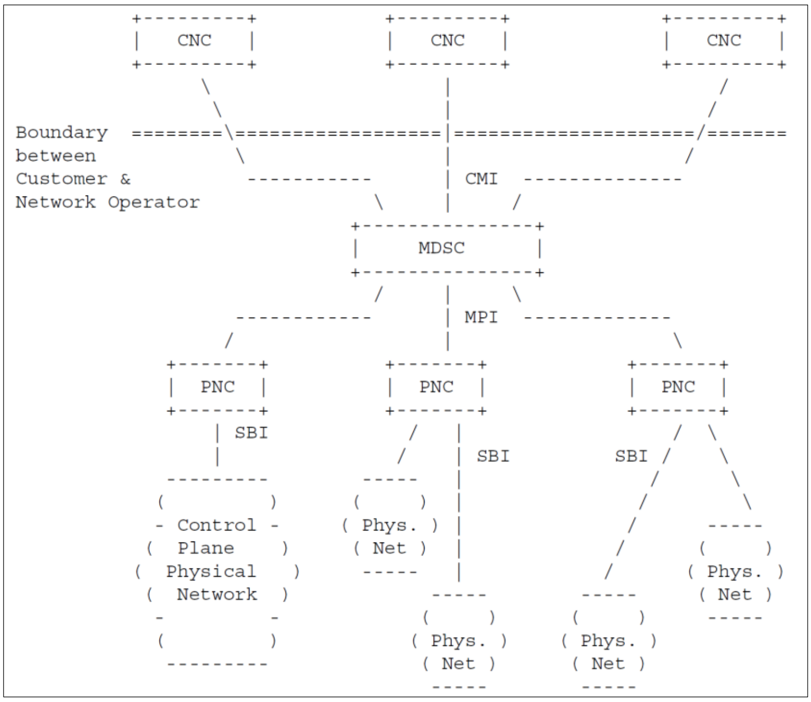


Figure 2: ACTN reference architecture from RFC8453

With reference to Figure 2, the interface between the PNC and the Multi-Domain Service Coordinator (MDSC), is called **MPI** (MDSC-PNC interface), and is the main object of this Plugtest.

Quoting RFC8453 [1]:

- “MPI: The MDSC-PNC Interface (MPI) is an interface between an MDSC and a PNC. It communicates requests for new connectivity or for bandwidth changes in the physical network. In multi-domain environments, the MDSC needs to communicate with multiple PNCs, each responsible for control of a domain. The MPI presents an abstracted topology to the MDSC hiding technology-specific aspects of the network and hiding topology according to policy.”

Quoting draft-ietf-teas-actn-yang-04 [12]:

- “The MDSC-PNC Interface (MPI) is an interface between a MDSC and a PNC. It allows the MDSC to communicate requests to create/delete connectivity or to modify bandwidth reservations in the physical network. In multi-domain environments, each PNC is responsible for a separate domain. The MDSC needs to establish multiple MPIs, one for each PNC and perform coordination between them to provide cross-domain connectivity. MPI plays an important role for multi-vendor mechanism, interoperability can be achieved by standardized interface modules.

The SBI (South Bound Interface) between PNC and NEs is out of the scope of this Plugtest.

6.2 Test Architecture

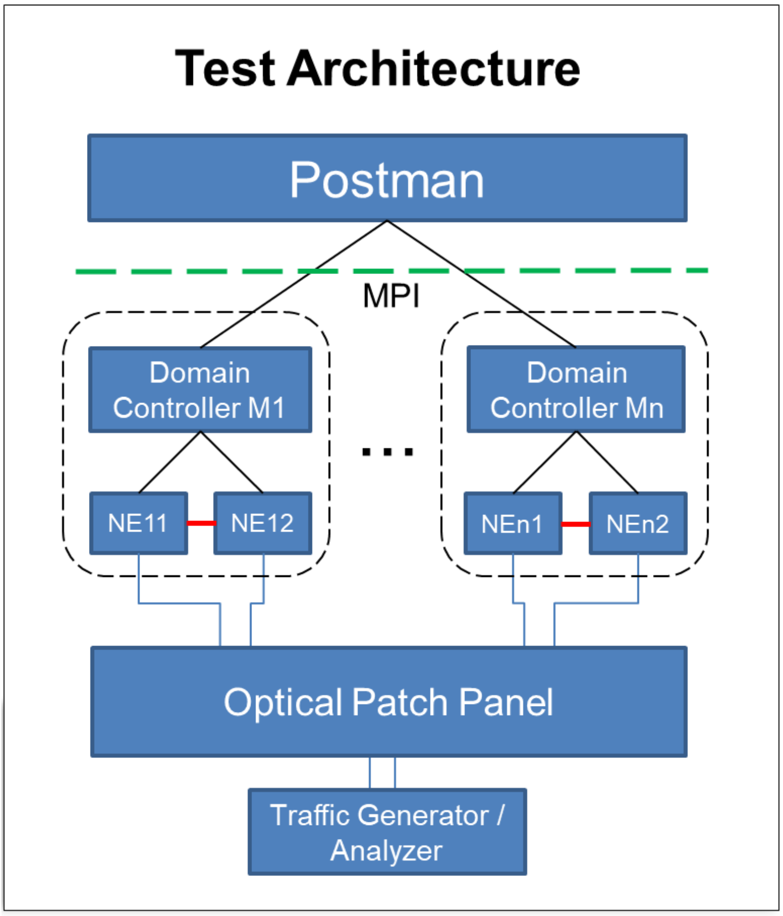


Figure 3 Generic Multi-Domain, Multi-Vendor SDN Architecture

With reference to Figure 2, each domain shall be composed by one MW link and the corresponding SDN controller, as was done in the 1st mWT Plugtest.

The basis for the MPI is based on the Restconf protocol (RFC 8040) and the YANG DM library provided by IETF (RFCs and relevant drafts, as specified in Annex C).

As depicted in figure 2, In order to simplify the test specification and implementation, the testing is performed by using an API Development and Testing environment, namely the Postman system.

Tests are be performed by exploiting the automation (via scripting) capability of Postman, with a single set of scripts being jointly developed specifically for these Plugtests by the Participants and stored in the ETSI Forge code repository [14].

Specifying a single set of scripts and the expected format and content of the related responses by the DCs, it is possible to univocally determine the compliance of the DCs to the relevant standards and confirm the multi-domain interoperability of the systems under test and the specified MPI.

The physical interconnections among the MW equipment required by the test configurations (see Chapter 6) are implemented by means of an optical patch panel. All domains are connected to the patch panel, and the connections among domains (when required) and between the NEs and the traffic generator/analyzer (TGA) are made using optical patch cords on the front of the patch panel.

6.3 Data Model Architecture

The IETF RFC and draft DMs to be used in this Plugtest are specified in Annex B IETF Data Model Selection, including the selection of the required subsets of attributes defined therein.

Figure 4, Figure 5 and Figure 6 depict a simplified DM topology overview as shall be used in this Plugtest.

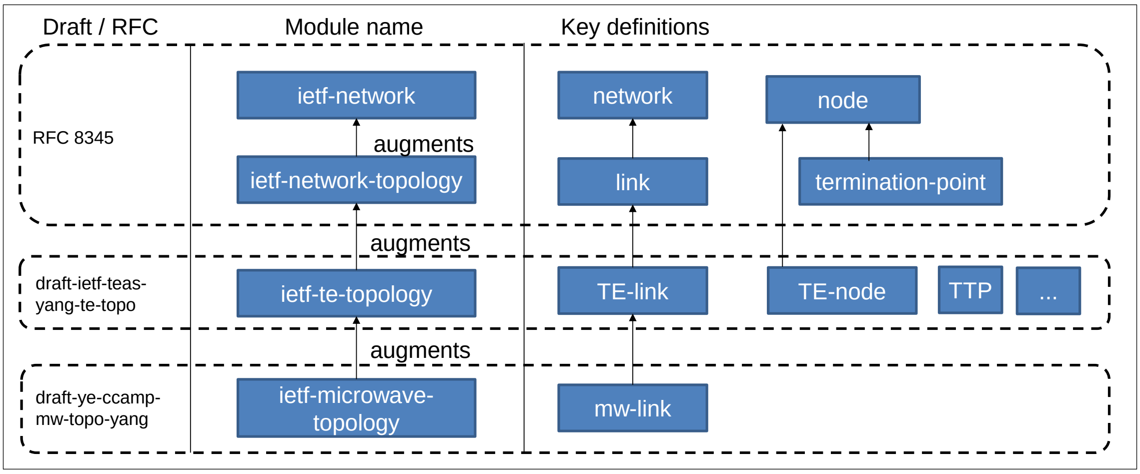


Figure 4 IETF Microwave Topology Models

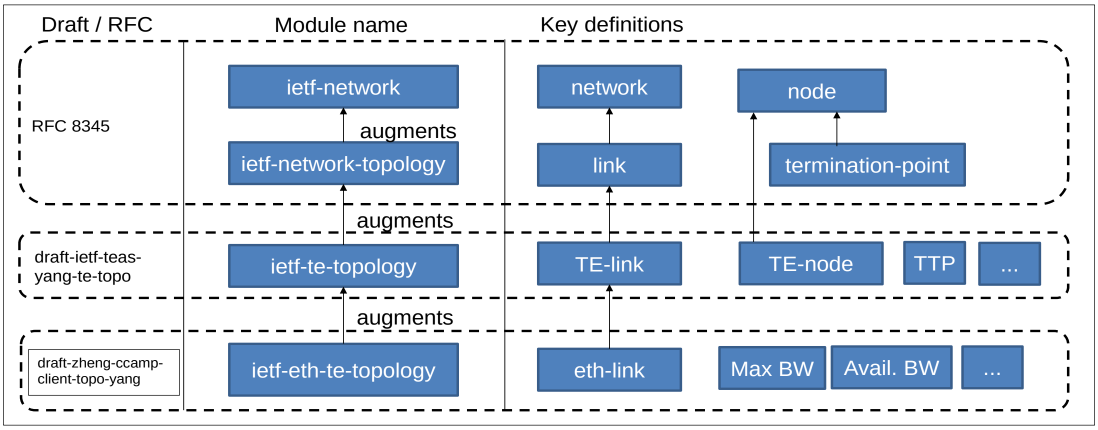


Figure 5 IETF Ethernet Topology Models

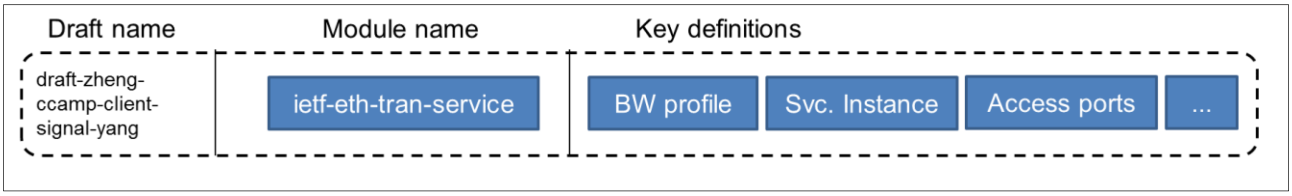


Figure 6 IETF Ethernet Service Model

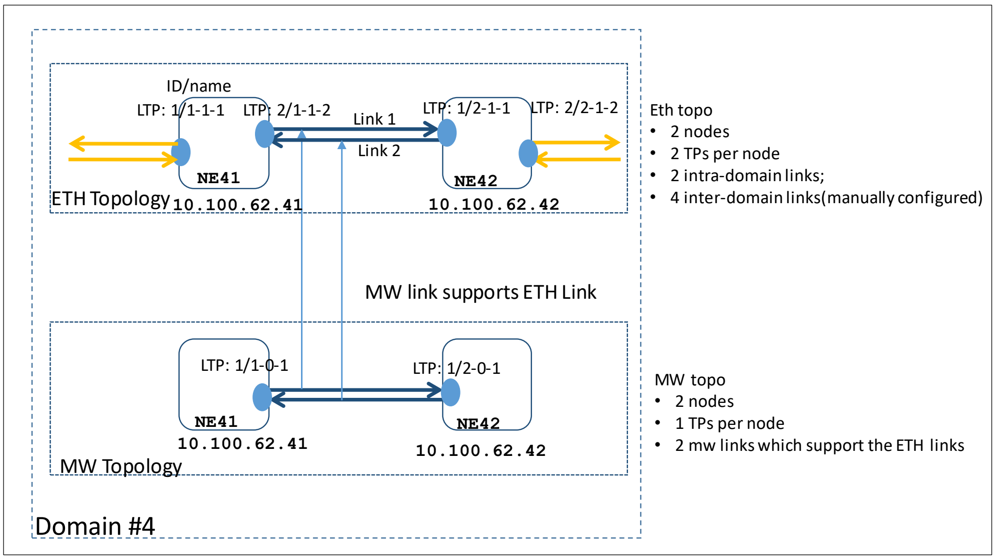


Figure 7 Single Domain Topology Exposed on the MPI

Note 1: Figure 7 describes a possible implementation, other Topologies may differ.

Note 2: The inter-domain links' information is not requested to be published across the NBI for these Plugtests.

6.4 Test LAN Structure

6.4.1 Logical Structure

The test LAN logical structure is depicted in Figure 8 Test LAN Architecture. The actual physical components may be different, but from the logical point of view the components are:

1. LAN Ethernet switch with copper GE connectivity to the NBI interface of the MPIs and Postman;
2. Internet access is available if needed through the HIVE Gateway;
3. WiFi connectivity is also available in the test area;
4. Any local networking arrangement and equipment necessary to connect the MW NEs to the corresponding PNC is out of scope of the test and shall be provided by each Participant for themselves;
5. The traffic ports of the MW links are not part of the test LAN – that is a completely separated physical network, connected only to the Optical Patch Panel (and through that to the packet test equipment).

Any other connectivity requirements for individual participants are left out of scope of the present document and may be discussed with the Plugtests organizational Team.

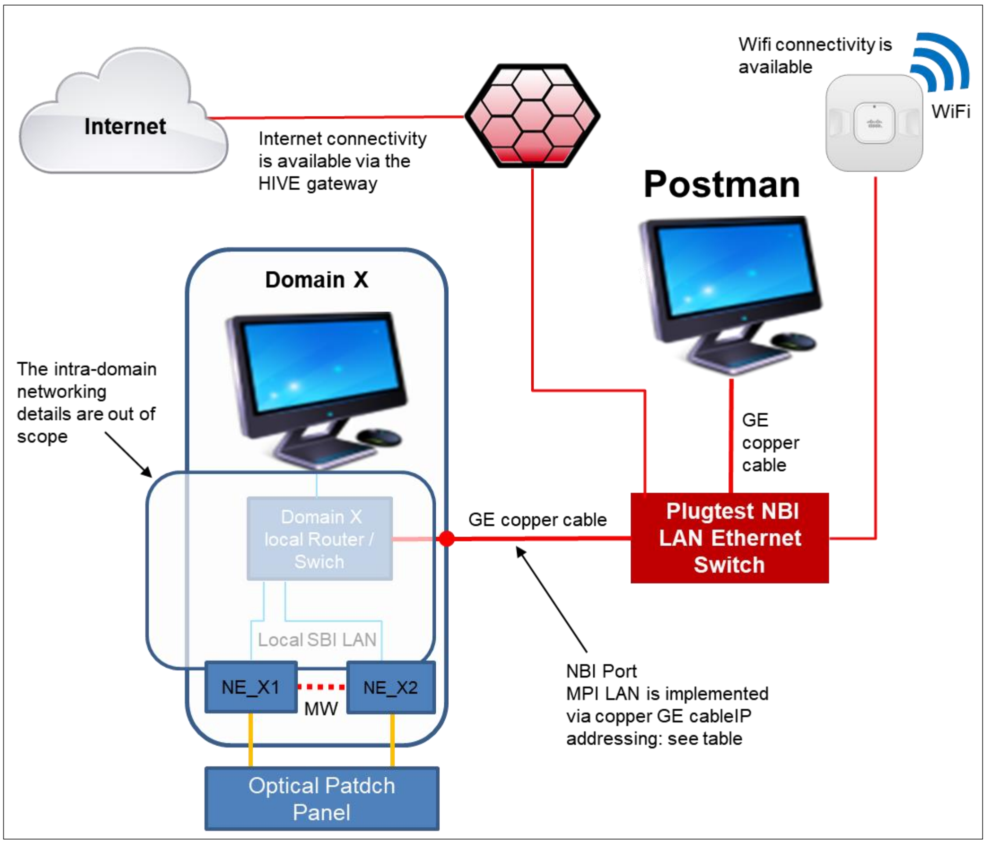


Figure 8 Test LAN Architecture

6.5 Local HTTP Server for LLDP Testing

As mentioned in Clause 5.3.1 LLDP Use for Inter-Domain Link Discovery, the LLDP test focuses on two main aspects:

1. Compliance: validation of the correctness of the plug-id value created by each Domain, depending on the other domain connected within the test session. This test is performed comparing the plug-id value returned by each domain with the reference value, produced by the reference algorithm using the LLDP packets captured on the live links under test. The result is an individual Pass/Fail test result for each individual Domain.

2. Interoperability: validation of the equivalence of the plug-id values computed by two different Domains under test, and validation of the interoperability level for plug-ids calculation and consumption. The result is a joint Pass/Fail test result for the given couple of Domains.

Step 1 requires that the usage of a test system external to the individual Domains. The test system should implement the following behavior:

1. LLDP packets for the couple of Domains under test are captured and saved in a location (URL) that is accessible by Postman,
2. Invoke the reference algorithm with the correct couple of captured LLDP packets.

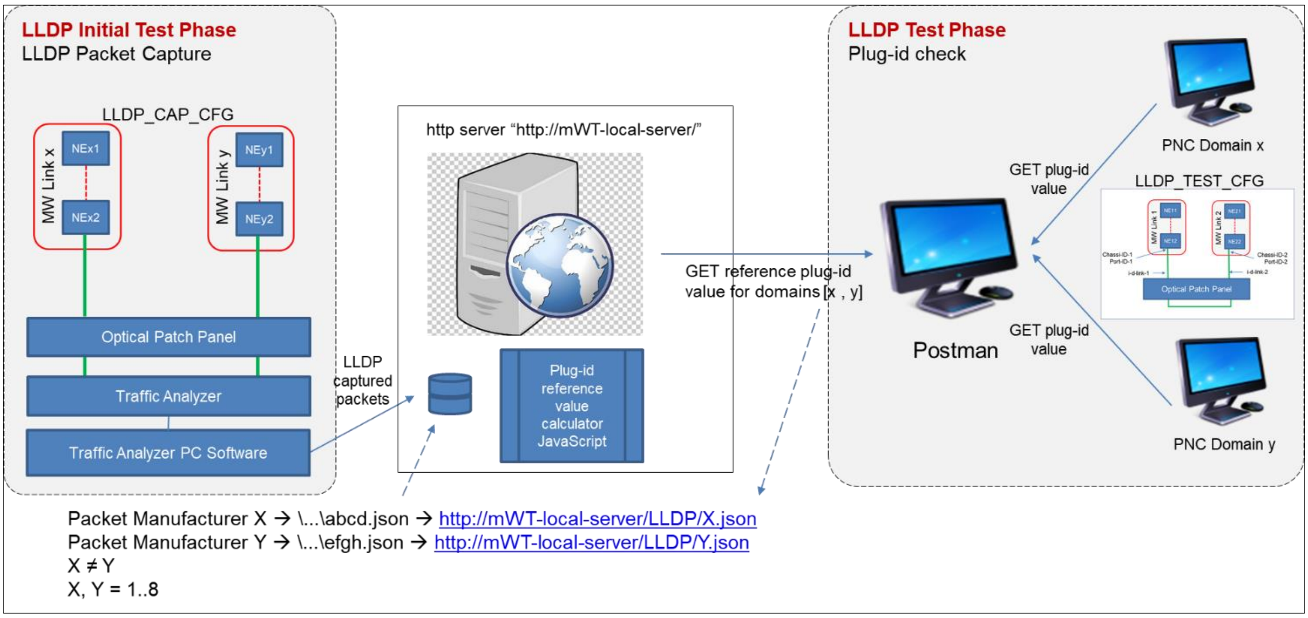


Figure 9 LLDP Test Infrastructure

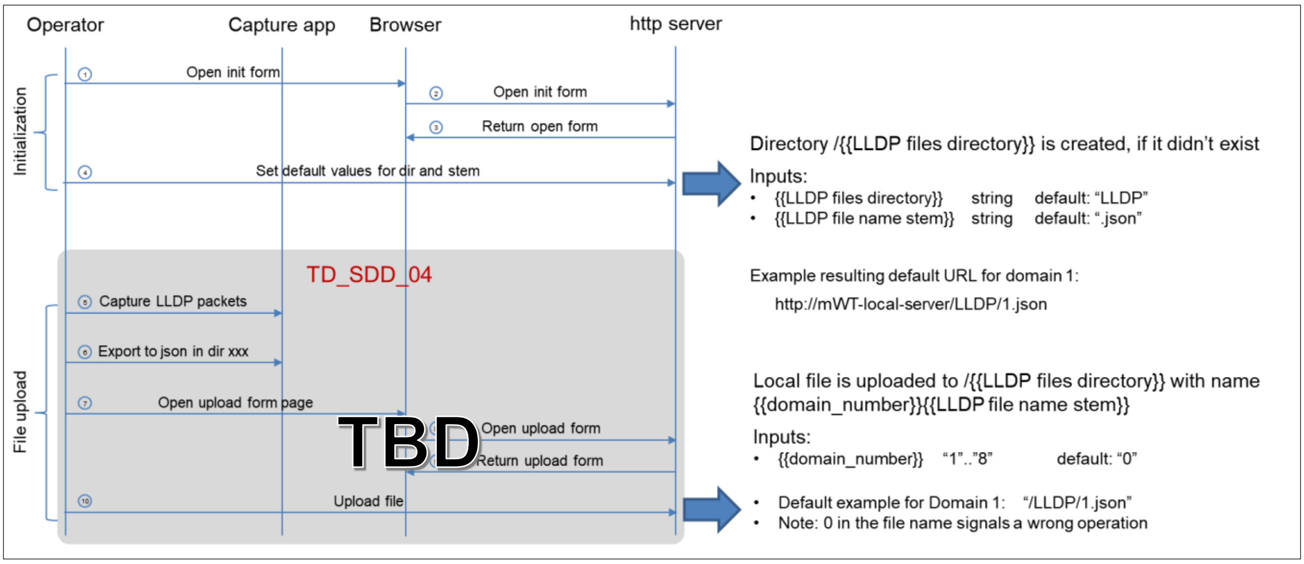


Figure 10 LLDP Test: Initialization and Packet Capture Phases

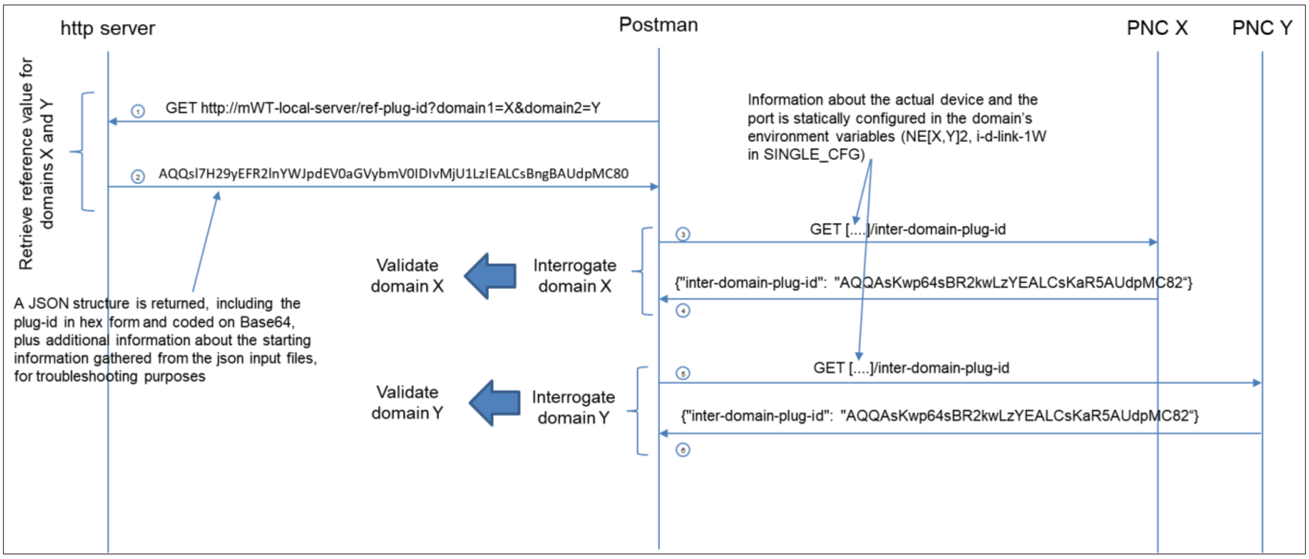


Figure 11 LLDP-based plug-id Value Validation against the Reference Algorithm

7 Configurations

Following the MEF Carrier Ethernet definitions for the L2 E-LINE services, which are commonly used in mobile backhaul, requires that each service be identified by a single S-VLAN tag.

Therefore, for the Plugtest we specify only one service to be terminated on a local port, since this is the general case.

This is a change from the original proposal, where be two services sharing the same S-VLAN were supposed to be terminated on different ports, based on the different values of the C-VLAN. An example is shown in Figure 12

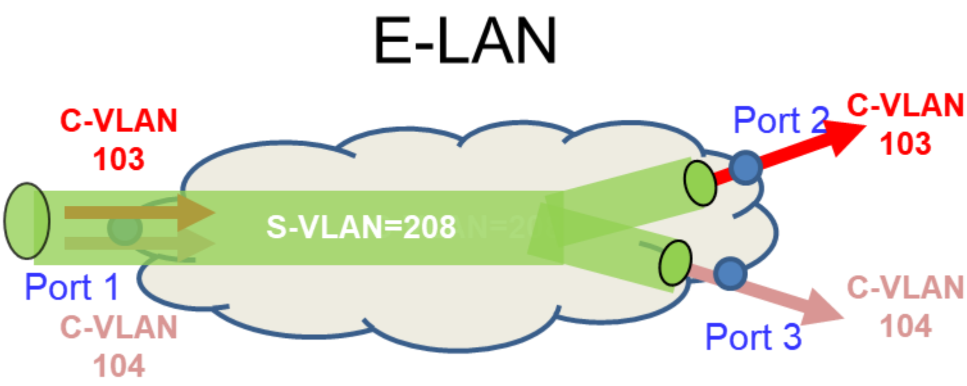


Figure 12 Example of MEF E-LAN service

Only E-LINE services are going to be used in this Plugtest.

7.1 Data Plane Connectivity

The Data Plane connectivity is required to validate the actual creation, correct functioning and deletion of the L2 Ethernet Services.

Each link shall provide optical GE interfaces (multimode fiber, LC connectors, 850nm) to be connected to the Optical Patch Panel.

In principle, to support all tests without re-wiring, at least 3 optical ports are required on a MW terminal**Error! Reference source not found.**. In case the equipment does not support enough optical ports, the test sequence can be arranged such that services are tested in sequence instead of all at once – with no loss of generality.

All ports from all Domains are connected to the back of the Optical Patch Panel in their orderly, fixed positions for the duration of the test *see for example Figure 15 SINGLE_CFG Physical Connections and Figure 21.

The connections between Domains and to the TGA, as required by each Configuration (see below), are all implemented at test time by using optical patch cords on the front on the Patch Panel. Examples are pictured where appropriate.

The equipment from Intracom have two optical ports that are 10GE only, and must therefore be connected directly to the 10GE ports available on the TGA.

7.2 Single Domain Test Configuration: SINGLE_CFG

Each Domain is first put through the Discovery and the L2 Service Provisioning tests (but not the LLDP tests) to ensure it individually complies with all the specifications, as a baseline.

7.2.1 SINGLE_CFG Logical Connections

One MW terminal provides a connection for the aggregated traffic of the services that will be provisioned in the “Root to Leaves” test section.

The other terminal provides two ports for inter-domain traffic (named East and West) and one port for traffic locally terminated in the Domain (Local Port).

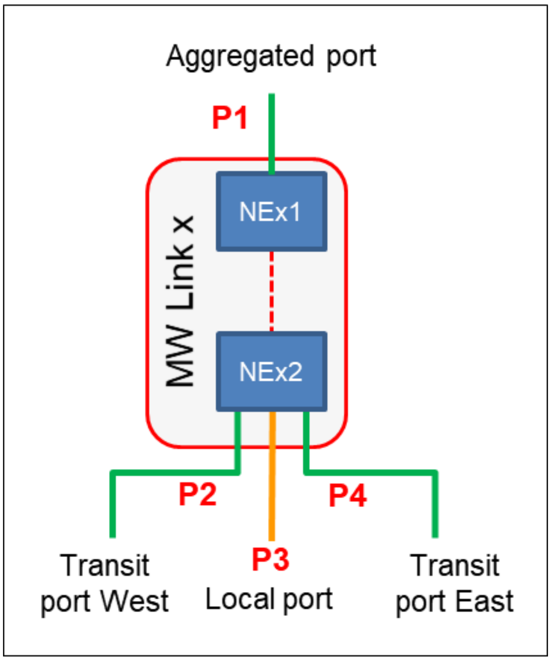


Figure 13 Data Plane Connectivity of each Domain in SINGLE_CFG

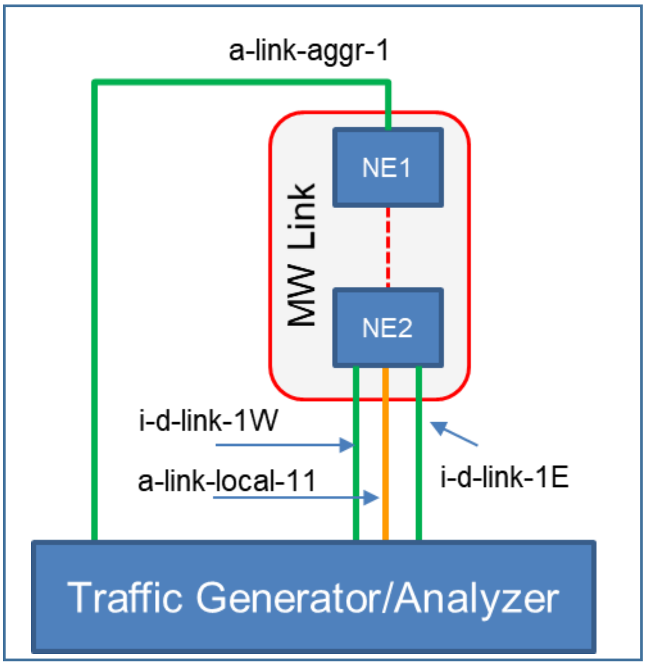


Figure 14 Logical Connection in the SINGLE_CFG Configuration

7.2.2 SINGLE_CFG Physical Connections

The optical ports on the equipment are connected 1:1 to the test instrument, through the optical patch panel.

In case the MW equipment and/or the TGA do not have enough optical ports, the test sequence can be suitably modified so that the services are not tested all at once.

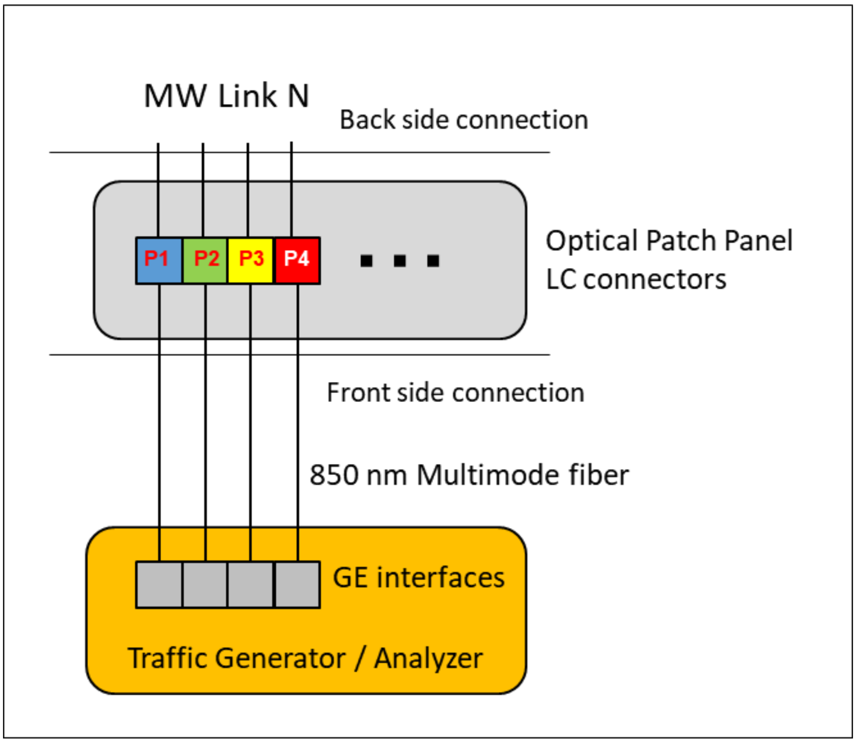


Figure 15 SINGLE_CFG Physical Connections

Depending on the number of ports available on the TGA, and on the number of TGAs available for the test, more than on SINGLE_CFG test suite can be executed at the same time.

The equipment from Intracom will have the 10GE ports P1 and P2 connected to the 10GE ports of the TGZ.

7.2.3 SINGLE_CFG LLDP Capture

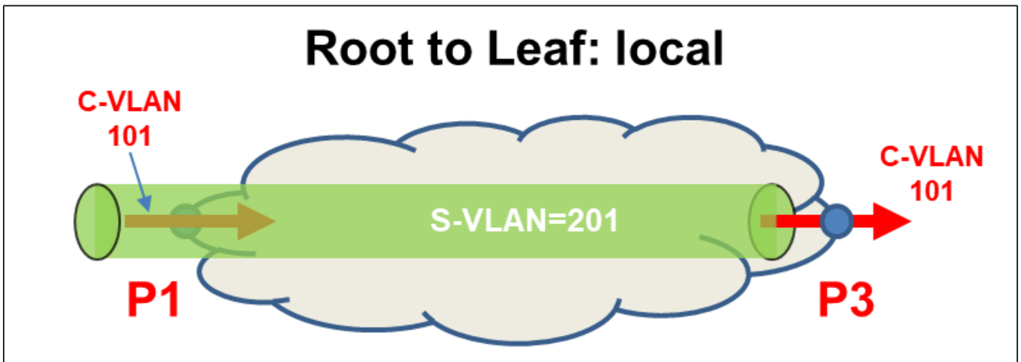
During the SINGLE_CFG testing, the LLDP packets are captured from P2/i-d-link-1W, and saved as explained in the LLDP description paragraph.

7.2.4 SINGLE_CFG Data Connections

All Data Connections are bi-directional.

Even if CIR and EIR are defined on a port without further specification, it is observed that equipment may not implement the PIR (CIR+EIR) control in the egress direction. The CIR enforcement is done on both directions.

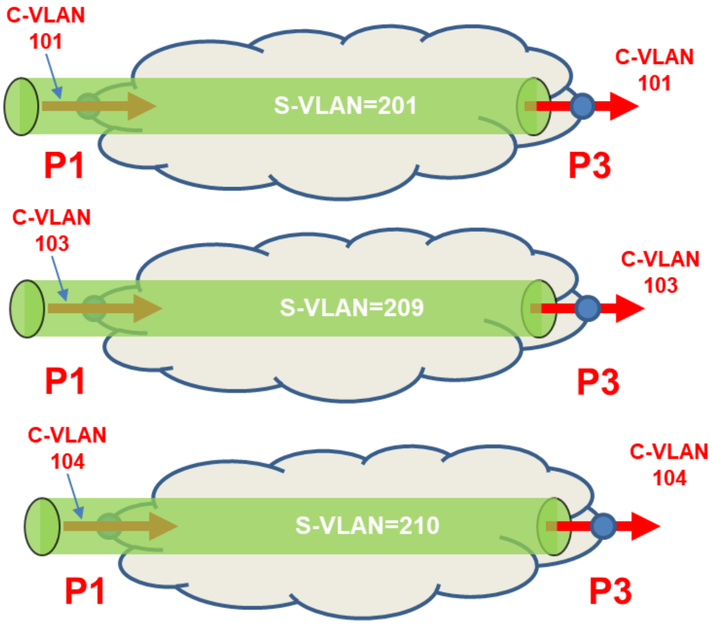
7.2.4.1. SINGLE_CFG Root to Leaf Service: Local



Service	Port	Outer Tag
S1-local	a-link-aggr-1	S-VLAN=201
	a-link-local-11	C-VLAN=101

Figure 16 SINGLE_CFG Configuration Data Connections – Local Service

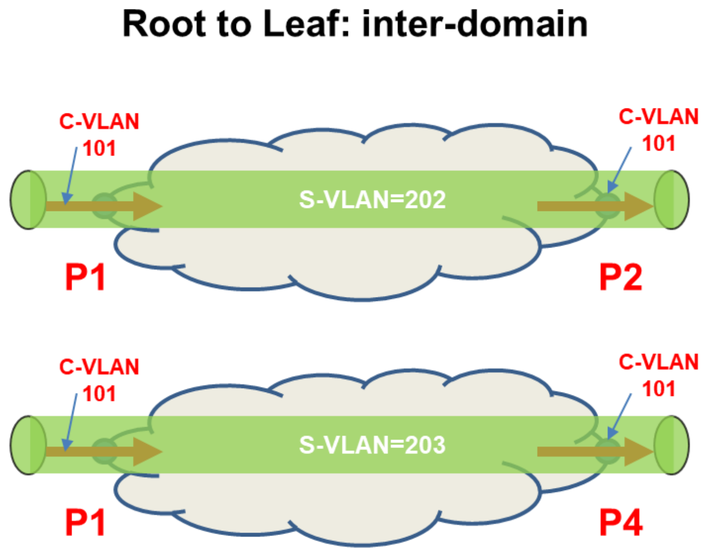
7.2.4.2.SINGLE_CFG Root to Leaf Service: Three Local Services



Service	Port	Outer Tag
S1-local	a-link-aggr-1	S-VLAN=2081
	a-link-local-11	C-VLAN=101
S2-local	a-link-aggr-1	S-VLAN=209
	a-link-local-11	C-VLAN=103
S3-local	a-link-aggr-1	S-VLAN=210
	a-link-local-11	C-VLAN=104

Figure 17 SINGLE_CFG Configuration Data Connections – Three Local Services

7.2.4.3. SINGLE_CFG Root to Leaf Service: Inter-Domain



Service	Port	Outer Tag
S1-inter-d	a-link-aggr-1	S-VLAN=202
	i-d-link-1W	S-VLAN=202
S2-inter-d	a-link-aggr-1	S-VLAN=203
	i-d-link-1E	S-VLAN=203

Figure 18 SINGLE_CFG Root to Leaf Services: Inter-Domain

Domain	Port	Service	S-VLAN	C-VLAN	BWP CIR/EIR	Total PIR	Instrument
Domain under test	P1	S1-local	S-VLAN=201	C-VLAN=101	50/50 Mbps	500 Mbps	G1, G5[1]
		S2-local	S-VLAN=209	C-VLAN=103	50/50 Mbps		
		S3-local	S-VLAN=210	C-VLAN=104	50/50 Mbps		
		S1-inter-d	S-VLAN=202				
		S2-inter-d	S-VLAN=203				
	P2	S1-inter-d	S-VLAN=202			100 Mbps	G2, G6
	P3	S1-local	S-VLAN=201	C-VLAN=101	50/50 Mbps	300 Mbps	G3, G7[1]
		S2-local	S-VLAN=209	C-VLAN=103	50/50 Mbps		
		S3-local	S-VLAN=210	C-VLAN=104	50/50 Mbps		
	P4	S2-inter-d	S-VLAN=203			100 Mbps	G4, G8

Table 1 Summary of Services on Domain 1 – SINGLE_CFG

Service	S-VLAN	C-VLAN	Port A	Port B
S1-local	S-VLAN=201	C-VLAN=101	G1/P1[1]	G3/P3[1]
S2-local	S-VLAN=209	C-VLAN=103		
S3-local	S-VLAN=210	C-VLAN=104		
S1-inter-d	S-VLAN=202	C-VLAN=101	G1/P1[1]	G2/P2
S2-inter-d	S-VLAN=203	C-VLAN=101	G1/P1[1]	G4/P4

Table 2 Summary of Service Termination Points

[1] The 10GE ports P1 and P3 of Intracom Equipment are connected to the 10GE interfaces of the TGA

7.3 3-Domain Test Configuration: TREE_CFG

The basic MW topology structure found in MBH networks is a tree (may collapse in a chain).

7.3.1 TREE_CFG Logical Connections

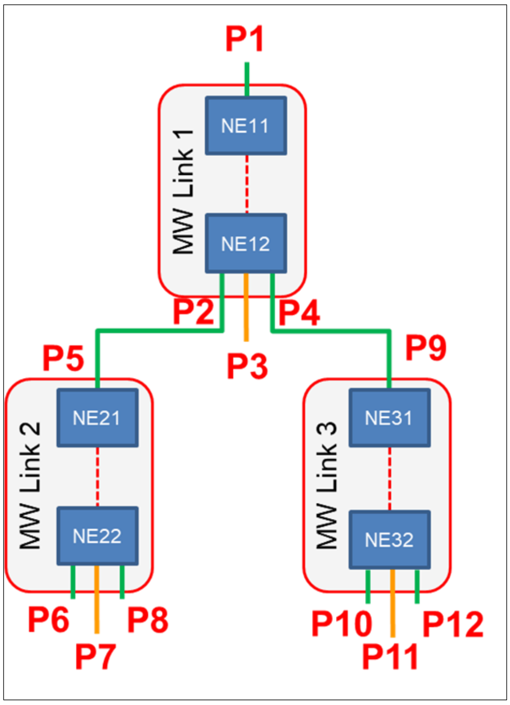


Figure 19 Port Designations of TREE_CFG

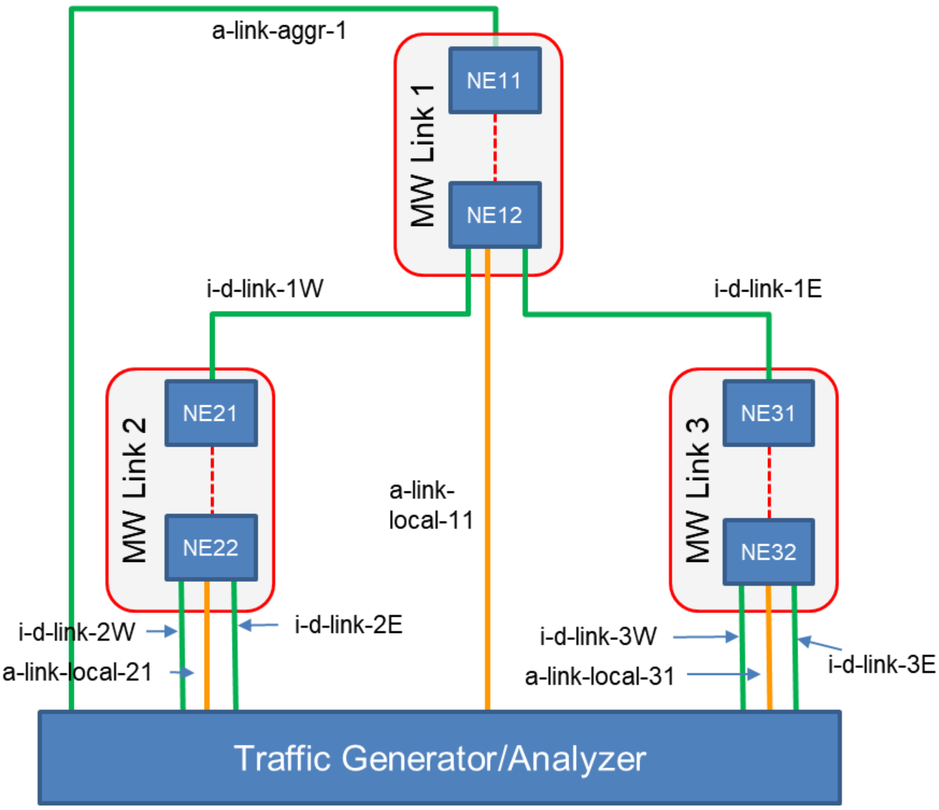


Figure 20 Logical Connection in the TREE_CFG Configuration

7.3.2 TREE_CFG Physical Connections

In case the MW equipment and/or the TGA do not have enough optical ports, the test sequence can be suitably modified so that the services are not tested all at once.

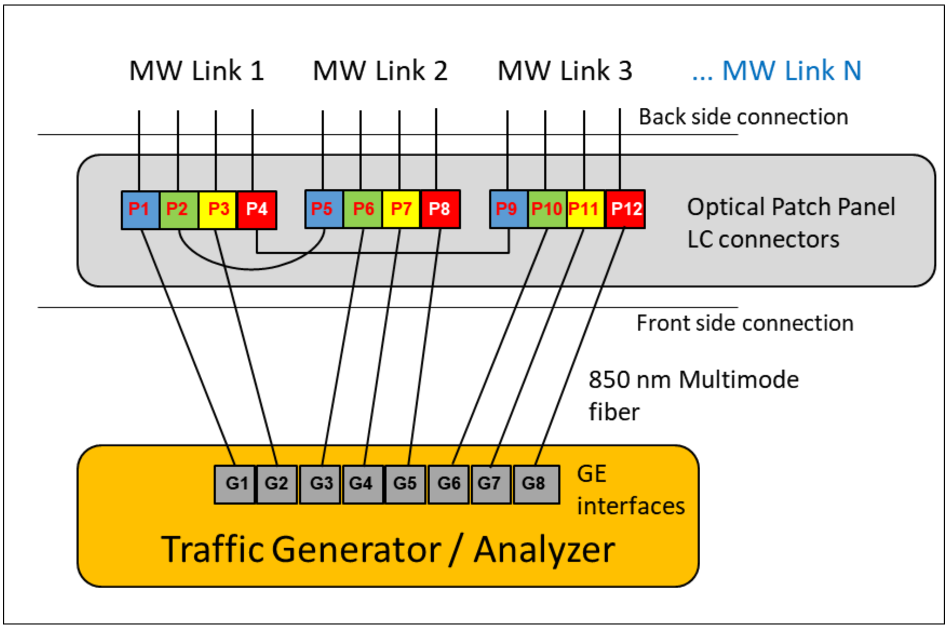


Figure 21 Physical Connections to Build one 3-Domain TREE_CFG Configuration

Depending on the number of ports available on the TGA, and on the number of TGAs available for the test, more than on TREE_CFG test suite can be executed at the same time.

Due to the 10GE interfaces available on the Intracom equipment, that domain must be in position “MW Link 1” and the 10GE ports P1 and nP3 are connected to the 10GE ports of the TGA,

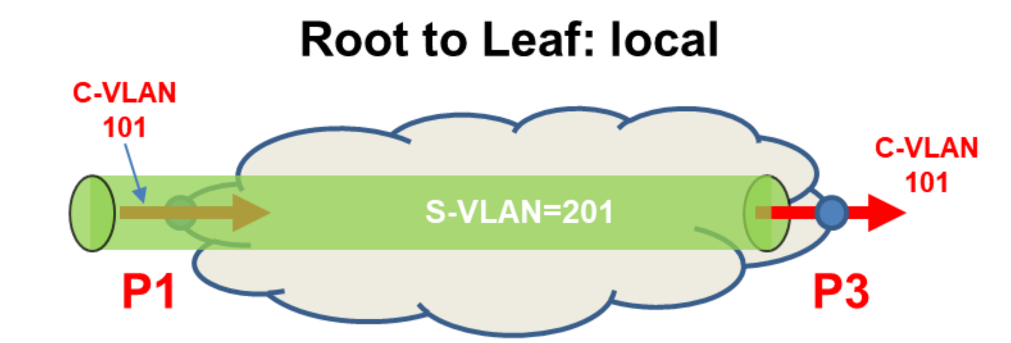
7.3.3 TREE_CFG Data Connections

All services are bi-directional and can be tested all at once or sequentially, depending on the available ports on the equipment and the test instrument(s). In actual networks all these services run at the same time, but this has no relevance for the test itself.

7.3.3.1 Root to Leaf Services: Local

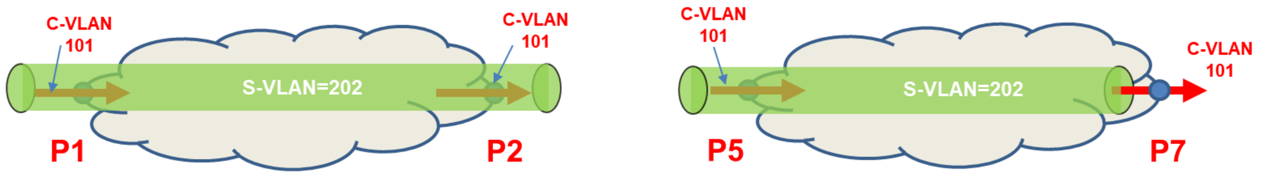
These services are originated in the TGA and flow all through the a-link-aggr-1 link.

Local Services are terminated on the local link on NE12, NE22 and NE32 (see Figure 20 Logical Connection in the TREE_CFG Configuration).



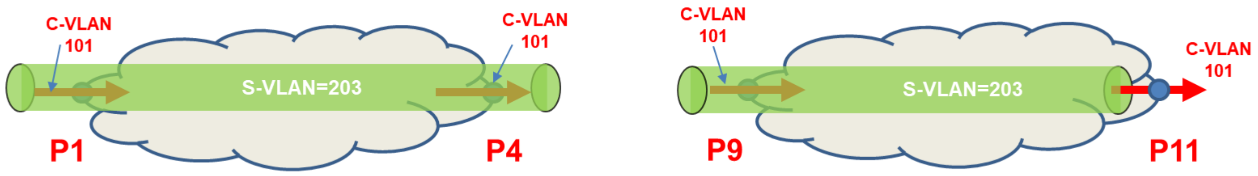
Service	Port	Outer Tag
S1-local	a-link-aggr-1	S-VLAN=201
	a-link-local-11	C-VLAN=101

Figure 22 TREE_CFG Root to Leaf Service: Local, Domain #1



Service	Port	Outer Tag
S2-local-1	a-link-aggr-1	S-VLAN=202
	i-d-link-1W	S-VLAN=202
S2-local-2	i-d-link-1W	S-VLAN=202
	a-link-local-21	C-VLAN=101

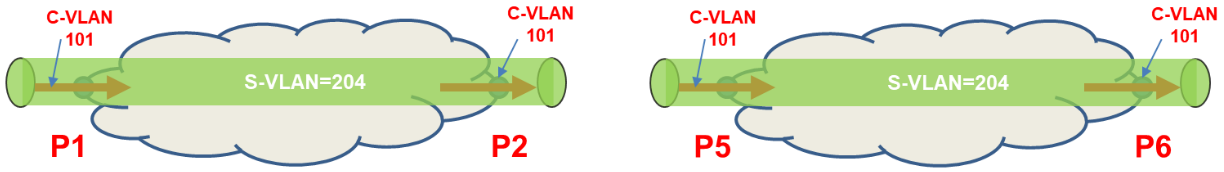
Figure 23 TREE_CFG Root to Leaf Service: Local, Domain #2



Service	Port	Outer Tag
S3-local-1	a-link-aggr-1	S-VLAN=203
	i-d-link-1E	S-VLAN=203
S3-local-2	i-d-link-1E	S-VLAN=203
	a-link-local-31	C-VLAN=101

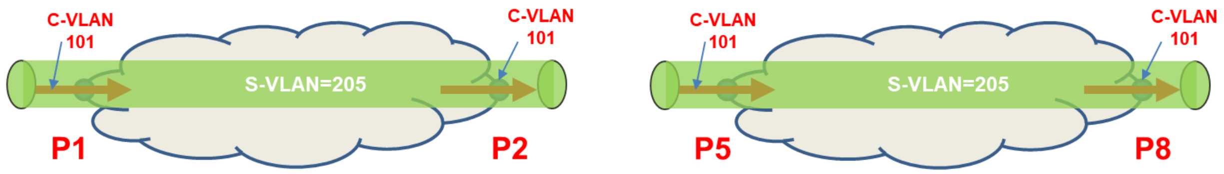
Figure 24 TREE_CFG Root to Leaf Service: Local, Domain #3

7.3.3.2 Root to Leaf Services: Inter-domain Links



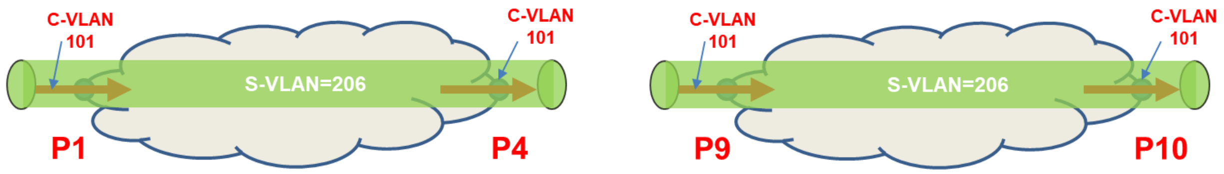
Service	Port	Outer Tag
S1-inter-d-1	a-link-aggr-1	S-VLAN=204
	i-d-link-1W	S-VLAN=204
S1-inter-d-2	i-d-link-1W	S-VLAN=204
	i-d-link-2W	S-VLAN=204

Figure 25 TREE_CFG Root to Leaf Services: Inter-domain Links, to Domain#2 West



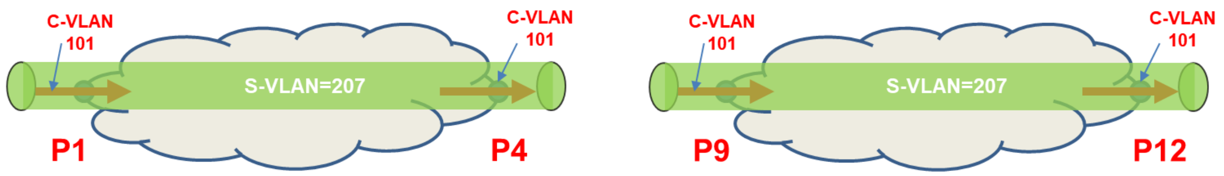
Service	Port	Outer Tag
S2-inter-d-1	a-link-aggr-1	S-VLAN=205
	i-d-link-1W	S-VLAN=205
S2-inter-d-2	i-d-link-1W	S-VLAN=205
	i-d-link-2E	S-VLAN=205

Figure 26 TREE_CFG Root to Leaf Services: Inter-domain Links, to Domain #2 East



Service	Port	Outer Tag
S3-inter-d-1	a-link-aggr-1	S-VLAN=206
	i-d-link-1E	S-VLAN=206
S3-inter-d-2	i-d-link-1E	S-VLAN=206
	i-d-link-3W	S-VLAN=206

Figure 27 TREE_CFG Root to Leaf Services: Inter-domain Links, to Domain #3 West



Service	Port	Outer Tag
S4-inter-d-1	a-link-aggr-1	S-VLAN=207
	i-d-link-1E	S-VLAN=207
S4-inter-d-2	i-d-link-1E	S-VLAN=207
	i-d-link-3E	S-VLAN=207

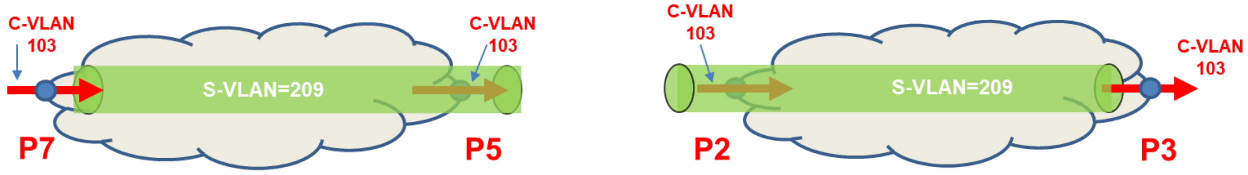
Figure 28 TREE_CFG Root to Leaf Services: Inter-domain Links, to Domain #3 East

7.3.3.3 Leaf to Leaf Services Test

These services are originated in the TGA and flow from one local link in one Domain to a local link in another Domain.

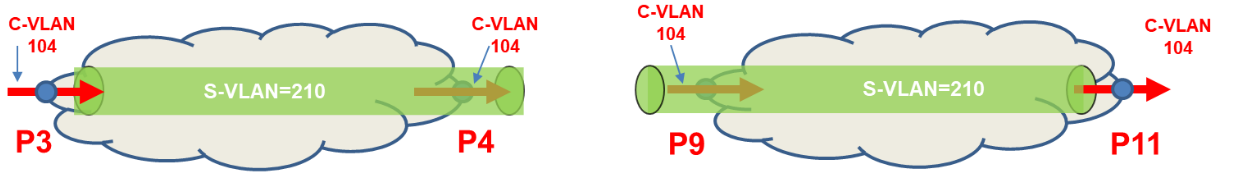
Service	Port	Outer Tag
S1-leaf-1	a-link-local-21	C-VLAN=102
	i-d-link-1W	S-VLAN=208
S1-leaf-2	i-d-link-1W	S-VLAN=208
	i-d-link-1E	S-VLAN=208
S1-leaf-3	i-d-link-1E	S-VLAN=208
	a-link-local-31	C-VLAN=102

Figure 29 TREE_CFG Leaf to Leaf Services: Domain #2 – Domain #3



Service	Port	Outer Tag
S2-leaf-1	a-link-local-21	C-VLAN=103
	i-d-link-1W	S-VLAN=209
S2-leaf-2	i-d-link-1W	S-VLAN=209
	a-link-local-11	C-VLAN=103

Figure 30 TREE_CFG Leaf to Leaf Services: Domain #1 – Domain #2



Service	Port	Outer Tag
S3-leaf-1	a-link-local-11	C-VLAN=104
	i-d-link-1E	S-VLAN=210
S3-leaf-2	i-d-link-1E	S-VLAN=210
	a-link-local-31	C-VLAN=104

Figure 31 TREE_CFG Leaf to Leaf Services: Domain #1 – Domain #3

Domain	Port	Service	S-VLAN	C-VLAN	BWP CIR/EIR	Total	Connection
MW Link 1	P1	S1-local	S-VLAN=201			700 Mbps	Instrument G1
		S2-local-1	S-VLAN=202				
		S3-local-1	S-VLAN=203				
		S1-inter-d-1	S-VLAN=204				
		S2-inter-d-1	S-VLAN=205				
		S3-inter-d-1	S-VLAN=206				
		S4-inter-d-1	S-VLAN=207				
	P2	S2-local-1	S-VLAN=202			500 Mbps	P5
		S1-inter-d-1	S-VLAN=204				
		S2-inter-d-1	S-VLAN=205				
		S1-leaf-2	S-VLAN=208				
		S2-leaf-2	S-VLAN=209				
	P3	S1-local	S-VLAN=201	C-VLAN=101	50/50 Mbps	300 Mbps	Instrument G2
		S2-leaf-2	S-VLAN=209	C-VLAN=103	50/50 Mbps		
		S3-leaf-1	S-VLAN=210	C-VLAN=104	50/50 Mbps		
	P4	S3-local-1	S-VLAN=203			500 Mbps	P9
		S3-inter-d-1	S-VLAN=206				
		S4-inter-d-1	S-VLAN=207				
		S1-leaf-2	S-VLAN=208				
		S3-leaf-1	S-VLAN=210				

Table 3 Summary of Services on Domain 1 – TREE_CFG

Domain	Port	Service	S-VLAN	C-VLAN	BWP CIR/EIR	Total PIR	Connection
MW Link 2	P5	S2-local-2	S-VLAN=202			500 Mbps	P2
		S1-inter-d-2	S-VLAN=204				
		S2-inter-d-2	S-VLAN=205				
		S1-leaf-1	S-VLAN=208				
		S2-leaf-1	S-VLAN=209				
	P6	S1-inter-d-2	S-VLAN=204			100 Mbps	Instrument G3
	P7	S2-local-2	S-VLAN=202	C-VLAN=101	50/50 Mbps	300 Mbps	Instrument G4
		S1-leaf-1	S-VLAN=208	C-VLAN=102	50/50 Mbps		
		S2-leaf-1	S-VLAN=209	C-VLAN=103	50/50 Mbps		
	P8	S2-inter-d-2	S-VLAN=205			100 Mbps	Instrument G5

Table 4 Summary of Services on Domain 2 – TREE_CFG

Domain	Port	Service	S-VLAN	C-VLAN	BWP CIR/EIR	Total PIR	Connection
MW Link 3	P9	S3-local-2	S-VLAN=203			500 Mbps	P4
		S3-inter-d-2	S-VLAN=206				
		S4-inter-d-2	S-VLAN=207				
		S1-leaf-3	S-VLAN=208				
		S3-leaf-2	S-VLAN=210				
	P10	S3-inter-d-2	S-VLAN=206			100 Mbps	Instrument G6
	P11	S3-local-2	S-VLAN=203	C-VLAN=101	50/50 Mbps	300 Mbps	Instrument G7
		S1-leaf-3	S-VLAN=208	C-VLAN=102	50/50 Mbps		
		S3-leaf-2	S-VLAN=210	C-VLAN=104	50/50 Mbps		
	P12	S4-inter-d-2	S-VLAN=207			100 Mbps	Instrument G8

Table 5 Summary of Services on Domain 3 – TREE_CFG

Port	Service	S-VLAN	C-VLAN	BWP CIR/EIR	Total PIR	Connection
G1	S1-local	S-VLAN=201	C-VLAN=101	50/50 Mbps	700 Mbps	P1
	S2-local	S-VLAN=202	C-VLAN=101	50/50 Mbps		
	S3-local	S-VLAN=203	C-VLAN=101	50/50 Mbps		
	S1-inter-d	S-VLAN=204	C-VLAN=101	50/50 Mbps		
	S2-inter-d	S-VLAN=205	C-VLAN=101	50/50 Mbps		
	S3-inter-d	S-VLAN=206	C-VLAN=101	50/50 Mbps		
	S4-inter-d	S-VLAN=207	C-VLAN=101	50/50 Mbps		
G2	S1-local	S-VLAN=201	C-VLAN=101	50/50 Mbps	300 Mbps	P3
	S2-leaf	S-VLAN=209	C-VLAN=103	50/50 Mbps		
	S3-leaf	S-VLAN=210	C-VLAN=104	50/50 Mbps		
G3	S1-inter-d	S-VLAN=204	C-VLAN=101	50/50 Mbps	100 Mbps	P6
G4	S2-local	S-VLAN=202	C-VLAN=101	50/50 Mbps	300 Mbps	P7
	S1-leaf	S-VLAN=208	C-VLAN=102	50/50 Mbps		
	S2-leaf	S-VLAN=209	C-VLAN=103	50/50 Mbps		
G5	S2-inter-d	S-VLAN=205	C-VLAN=101	50/50 Mbps	100 Mbps	P8
G6	S3-inter-d	S-VLAN=206	C-VLAN=101	50/50 Mbps	100 Mbps	P10
G7	S3-local	S-VLAN=203	C-VLAN=101	50/50 Mbps	300 Mbps	P11
	S1-leaf	S-VLAN=208	C-VLAN=102	50/50 Mbps		
	S3-leaf	S-VLAN=210	C-VLAN=104	50/50 Mbps		
G8	S4-inter-d	S-VLAN=207	C-VLAN=101	50/50 Mbps	100 Mbps	P12

Table 6 Summary of Services the Traffic Generator/Analyzer

Service	S-VLAN	C-VLAN	Port A	Port B
S1-local	S-VLAN=201	C-VLAN=101	G1/P1	G2/P3
S2-local	S-VLAN=202	C-VLAN=101	G1/P1	G4/P7
S3-local	S-VLAN=203	C-VLAN=101	G1/P1	G7/P11
S1-inter-d	S-VLAN=204	C-VLAN=101	G1/P1	G3/P6
S2-inter-d	S-VLAN=205	C-VLAN=101	G1/P1	G5/P8
S3-inter-d	S-VLAN=206	C-VLAN=101	G1/P1	G6/P10
S4-inter-d	S-VLAN=207	C-VLAN=101	G1/P1	G8/P12
S1-leaf	S-VLAN=208	C-VLAN=102	G4/P7	G7/P11
S2-leaf	S-VLAN=209	C-VLAN=103	G2/P3	G4/P7
S3-leaf	S-VLAN=210	C-VLAN=104	G2/P3	G7/P11

Table 7 Summary of Service Termination Points

7.3.4 TREE_CFG Domain Test Assignments

In addition to testing the compliance to the MPI specification by checking the PNC's responses and the model database entries, part of the test plan is to also check the configured L2 services do actually work as expected.

This is done by using a packet data generator/analyzer (TGA) and the corresponding management software.

There is more than one possible way to proceed for this test, the following proposal is to be discussed by all Participants. An alternative to the proposed test procedure would be e.g. to build one big tree with all domains connected together at the same time. There are several pros and cons for each possible way of proceeding.

1. The domains are arranged in the TREE_CFG configuration
    a. This allows to have multi-domain interoperability testing
    b. The specific combination of domains assigned to each TREE_CFG should not be relevant:
        i. The MPI is compliant
        ii. The L0, L1 and L2 interoperability of the MW equipment are out of question
        iii. If any couple of compliant domains supports correctly a multi-domain service, there is no reason to expect it would not interoperate with any other domain tested in the same way. Therefore, it is **not** required to test multiple combinations of domains so that any possible combination is met.
        iv. The combinations could be simply by alphabetical order, or randomized if so desired
    c. This minimizes the number of traffic ports on the TGA and allows parallelizing the tests, if more than one test instrument is available
2. If not enough ports are available on the test instrument, the available ports can be connected to the TREE_CFG in sequence

a. The Root-to-Leaf services would need 11 ports at once
b. The Leaf-to-Leaf services would need 3 ports
c. In the worst case, the TGA may provide two ports for the inter-domain services (S-VLAN and C-VLAN: root access link and East or West inter-domain link in the leaf), and one for the leaf local connection (C-VLAN)
d. Testing the services in sequence does not reduce the test significance in any way

Example test sequence in alphabetical order:

Test Assignment	Domains under Test
TREE_1	MW Link 1: Aviat MW Link 2: Ceragon MW Link 3: Ericsson
TREE_2	MW Link 1: Intracom MW Link 2: Huawei MW Link 3: NEC
TREE_3	MW Link 1: Nokia MW Link 2: SIAE MW Link 3: Voluntary Domain

Table 8 Example TREE_CFG Domain Test Assignments

7.4 LLDP Test Configuration: LLDP_TEST_CFG

This configuration is required to test the interoperability between the two Domains under test.

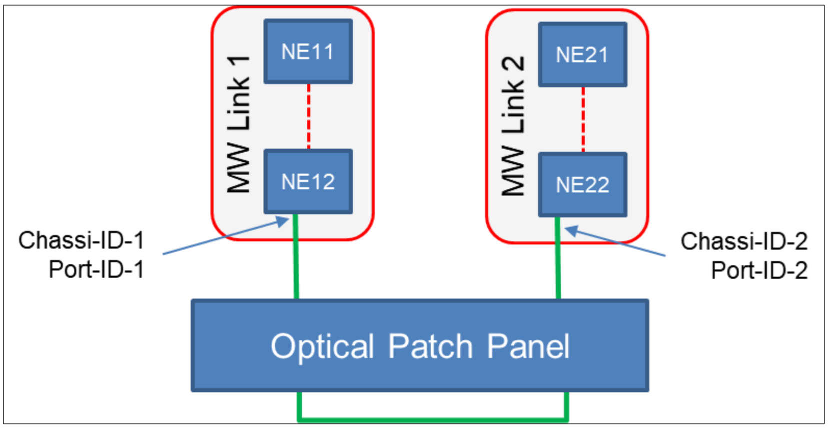


Figure 32 LLDP_TEST_CFG LLDP Packets Capture Phase Configuration

To simplify the automation of the test scripts, the **LLDP packets are captured during the single-domain testing phase**, and are named and stored according to the following scheme:

Domain Name	Domain Number	LLDP Captured Packet URL
Aviat	1	http://mWT-local-server/LLDP/1.json
Ceragon	2	http://mWT-local-server/LLDP/2.json
Ericsson	3	http://mWT-local-server/LLDP/3.json
Huawei	4	http://mWT-local-server/LLDP/4.json
Intracom	5	http://mWT-local-server/LLDP/5.json
NEC	6	http://mWT-local-server/LLDP/6.json
Nokia	7	http://mWT-local-server/LLDP/7.json
SIAE	8	http://mWT-local-server/LLDP/8.json

Table 9 LLDP Captured Packets URL

7.4.1 LLDP_TEST_CFG Domain Test Assignments

All possible combinations of domains must be tested. With 8 Participants, up to 4 tests can be run in parallel.

For sake of simplicity, let's enumerate the domains in alphabetical order:

1. Aviat
2. Ceragon
3. Ericsson
4. Huawei
5. Intracom
6. NEC
7. Nokia
8. SIAE

One possible combination is:

Test Assignment	Domains under Test			
LLDP_1	1-2	3-4	5-6	7-8
LLDP_2	1-3	2-4	5-7	6-8
LLDP_3	1-4	2-3	6-7	5-8
LLDP_4	1-5	2-6	3-7	4-8
LLDP_5	1-6	2-5	3-8	4-7
LLDP_6	1-7	2-8	3-5	4-6
LLDP_7	1-8	2-7	3-6	4-5

Table 10 Example LLDP Domain Test Assignments

8 Test Summary

8.1 Postman Environment Initialization Procedure

8.1.1 Applicable configurations

Not relevant.

8.1.2 List of objectives

Test ID	Objective
TD_POSTMAN_INIT	Create and initialize the POSTMAN_SETTINGS object in the runtime environment of Postman. This needs to be done only once at the beginning of a test session with Postman.

Table 11 Initialization Test Objectives

8.2 Single Domain Discovery Tests

Note: the Service discovery is only performed in the L2 Service Provisioning tests.

8.2.1 Applicable configurations

- SINGLE_CFG

8.2.2 List of objectives

Test ID	Objective
TD_SDD_01	Issue a request via Postman to one individual domain controller, to check it's reachability and basic functionality.
TD_SDD_02	The microwave topology information are requested from a single DC. The received information is compared to a template and checked for compliance.
TD_SDD_03	The Ethernet topology information are requested from a single DC. The received information is compared to a template and checked for compliance.
TD_SDD_04	The LLDP packets of i-d-link-1W are captured and stored in the corresponding URL in the local http server, as described in Table 9 LLDP Captured Packets URL for later use.

Table 12 Single Domain Discovery Test Objectives

8.3 Multi-Domain Discovery Tests

8.3.1 Applicable configurations

- TREE_CFG

8.3.2 List of objectives

Test ID	Objective
TD_MDD_01	Issue a request via Postman to all domain controllers, to check the overall viability of the test network.
TD_MDD_02	The microwave topology information are requested from all DCs connected to the NBI LAN at the same time for the same information as TD_SDD_02. This allows to check that the connectivity to all DC is fully functional. A comparison of the answers received may be performed to check consistency and compliance.
TD_MDD_03	The Ethernet topology information are requested from all DCs connected to the NBI LAN at the same time for the same information as TD_SDD_03. This allows to check that the connectivity to all DC is fully functional. A comparison of the answers received may be performed to check consistency and compliance.

Table 13 Multi-Domain Discovery Test Objectives

8.4 Single-Domain Service Provisioning Tests

8.4.1 Applicable configurations

- SINGLE_CFG

8.4.2 List of objectives

Test ID	Objective
TD_SSP_01	Create the specified L2 data services over a single domain. The TGA confirms that data start flowing.
TD_SSP_02	The Ethernet service information is requested from the single DC under test in TD_SSP_01. The received information is checked to correctly list the newly created services.
TD_SSP_03	Delete the specified L2 data services over a single domain. The TGA confirms that data stops flowing.
TD_SSP_04	The Ethernet service information is requested from the single DC under test in TD_SSP_03. The received information is checked to correctly not list the newly deleted services anymore.

Table 14 Single Domain Service Provisioning Test Objectives

8.5 Multi-Domain Service Provisioning Tests

8.5.1 Applicable configurations

- TREE_CFG

8.5.2 List of objectives

Test ID	Objective
TD_MSP_01	Create the specified L2 data services over all available domains. The TGA confirms that data start flowing. The traffic on the inter-domain links is classified based on S-VLAN only.
TD_MSP_02	The Ethernet service information is requested from all the DCs under test in TD_MSP_01. The received information is checked to correctly list the newly created services.
TD_MSP_03	Delete the specified L2 data service over all available domains. The TGA confirms that data stops flowing.
TD_MSP_04	The Ethernet service information is requested from all the DCs under test in TD_MSP_03. The received information is checked to correctly not list the newly deleted services anymore.

Table 15 Single Domain Service Provisioning Test Objectives

8.6 LLDP Tests

8.6.1 Applicable configurations

- LLDP_TEST_CFG

8.6.2 List of objectives

Test ID	Objective
TD_LLDP_01	The inter-domain-plug-id value associated to the port used to capture the LLDP packets in TD_SDD_04 is requested from the two DCs under test. For each DC, this value is compared to the reference value calculated by the reference algorithm, available on the local http server, based on the given couple of DCs under test. Each DC is hereby tested for compliance with the reference algorithm.

Table 16 LLDP Test Objectives

9 Test Descriptions

9.1 Initialization Test Descriptions

Interoperability Test Description			
Identifier	TD_POSTMAN_INIT		
Test Objective	Create and initialize the POSTMAN_SETTINGS object in the runtime environment of Postman. This needs to be done only once at the beginning of a test session with Postman. In case implementation-specific parameters contained in the TD_POSTMAN_INIT source code have been changed, it should be run again, followed by TD_SDD_02 and TD_SDD_03 (or TD_MDD_02 and TD_MDD_03, depending on the case).		
Configuration	Not relevant		
References			
Applicability	Not relevant		
Pre-test conditions	• Postman is running		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Launch the* TD_POSTMAN_INIT *script from Postman's GUI*
	2	***Validation***	*TBD*

9.2 Network and Service Discovery Test Descriptions

Interoperability Test Description			
Identifier	TD_SDD_01		
Test Objective	Issue a request via Postman to one individual domain controller, to check its reachability and basic functionality.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.)		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to one individual domain controller by executing Collection TD_SDD_01*
	2	***Validation***	*Check the response body of the above request and confirm if the Restconf server is serviceable.*
	3	***Validation***	*The response body of the request should contain a list of all YANG modules and submodules used by the Restconf server along with information about name and revision for each module.*
	4	***Validation***	*The response body of each query should contain the specified YANG module along with its name and revision.*

Interoperability Test Description				
Identifier	TD_MDD_01			
Test Objective	Issue a request via Postman to all domain controllers, to check the overall viability of the test network.			
Configuration	TREE_CFG			
References				
Applicability	MW_8040, MW_8345			
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • All the Domain Controller instances are up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.)			
Test Sequence	**Step**	**Type**	**Description**	
	1	***Request***	*Send GET request via Postman to all domain controllers by executing Collection TD_MDD_01*	
	2	***Validation***	*Check the response body of each request and confirm if all the Restconf servers are serviceable.*	
	3	***Validation***	*The response body of the request should contain a list of all YANG modules and submodules used by the Restconf server along with information about name and revision for each module.*	
	4	***Validation***	*The response body of each query should contain the specified YANG module along with its name and revision.*	

Interoperability Test Description			
Identifier	TD_SDD_02		
Test Objective	The microwave topology information are requested from a single DC. The received information is compared to a template and checked for compliance.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT IT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to one individual domain controller by executing Collection TD_SDD_02.*
	2	***Validation***	*The response body should contain information about the microwave topology in JSON format as specified in a separate document.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_SDD_03		
Test Objective	The Ethernet topology information are requested from a single DC. The received information is compared to a template and checked for compliance.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to one individual domain controller by executing Collection TD_SDD_03_S.*
	2	***Validation***	*The response body should contain information about the Ethernet topology in JSON format as specified in a separate document.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_SDD_04		
Test Objective	The LLDP packets of i-d-link-1W are captured and stored in the corresponding URL in the local http server, as described in Paragraph 6.4 (Figure 8) for later use.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_LLDP		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable • The LLDP protocol has been enabled in bi-directional mode on the inter-domain port West • The packet analyzer instrument is connected to inter-domain port West and ready to capture LLDP packets		
Test Sequence	**Step**	**Type**	**Description**
	1	***Action***	*Capture one or more LLDP packets from the Ethernet port*
	2	***Validation***	*It is verified that at least one correctly formed LLDP packet has been captured*
	3	***Action***	*Select one LLDP packet and save it in JSON format at the URL described in* Table 10 Example LLDP Domain Test Assignments
	4	***Validation***	*The JSON file is available at the correct URL and is correctly formatted*

Interoperability Test Description			
Identifier	TD_MDD_02		
Test Objective	The microwave topology information are requested from all DCs connected to the NBI LAN at the same time for the same information as TD_SDD_02. This allows to check that the connectivity to all DC is fully functional. A comparison of the answers received may be performed to check consistency and compliance.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • All the Domain Controller instances are up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to all domain controllers by executing Collection TD_MDD_02*
	2	***Validation***	*The response body of each request should contain information about the microwave topology in JSON format as specified in a separate document.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_MDD_03		
Test Objective	The Ethernet topology information are requested from all DCs connected to the NBI LAN at the same time for the same information as TD_SDD_03_S. This allows to check that the connectivity to all DC is fully functional. A comparison of the answers received may be performed to check consistency and compliance.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • All the Domain Controller instances are up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to all domain controllers by executing Collection TD_MDD_03*
	2	***Validation***	*The response body of each request should contain information about the Ethernet topology in JSON format as specified in a separate document.*

9.3 Service Provisioning Test Descriptions

Interoperability Test Description			
Identifier	TD_SSP_01		
Test Objective	Create the specified L2 data services over a single domain. The TGA confirms that data start flowing.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send POST request via Postman to one individual domain controller by executing Collection TD_ SSP_01.*
	2	***Validation***	*Check the TGA if the data start flowing properly.*

Interoperability Test Description			
Identifier	TD_SSP_02		
Test Objective	The Ethernet service information is requested from the single DC under test in TD_SSP_01. The received information is checked to correctly list the newly created services.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to one individual domain controller by executing Collection TD_ SSP_02*
	2	***Validation***	*The response body should contain information about the created L2 service in JSON format referring to a separate document.*

Interoperability Test Description			
Identifier	TD_SSP_03		
Test Objective	Delete the specified L2 data services over a single domain. The TGA confirms that data stops flowing.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send DELETE request via Postman to one individual domain controller by executing Collection TD_ SSP_03*
	2	***Validation***	*Check the TGA if the data stop flowing.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_SSP_04		
Test Objective	The Ethernet service information is requested from the single DC under test in TD_SSP_03. The received information is checked to correctly not list the newly deleted services anymore.		
Configuration	SINGLE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • The Restconf server is serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to one individual domain controller by executing Collection TD_ SSP_04*
	2	***Validation***	*The response body should no longer contain information about the L2 service deleted in TD_ SSP_03.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_MSP_01		
Test Objective	Create the specified L2 data services over all available domains. The TGA confirms that data start flowing.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable. • If TD_SSP_01 has been run before TD_ MSP_01, all Domain Controllers and microwave units should be reset to the state they were before executing TD_SSP_01		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send POST request via Postman to all domain controllers by executing Collection TD_ SSP_01*
	2	***Validation***	*Check the TGA if the data start flowing properly.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_ MSP_02		
Test Objective	The Ethernet service information is requested from all the DCs under test in TD_MSP_01_S. The received information is checked to correctly list the newly created services.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • All the Domain Controller instances are up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable. • All L2 data services are successfully created.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to all domain controllers by executing Collection TD_SSP_02*
	2	***Validation***	*The response body of each request should contain information about the created L2 service in JSON format referring to a separate document.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_MSP_03		
Test Objective	Delete the specified L2 data service over all available domains. The TGA confirms that data stops flowing.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • The Domain Controller instance is up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send DELETE request via Postman to all domain controllers by executing Collection TD_ SDN_ ISP_03*
	2	***Validation***	*Check the TGA if the data stop flowing.*
	3		
	4		

Interoperability Test Description			
Identifier	TD_ MSP_04		
Test Objective	The Ethernet service information is requested from all the DCs under test in TD_MSP_03_S. The received information is checked to correctly not list the newly deleted services anymore.		
Configuration	TREE_CFG		
References			
Applicability	MW_8040, MW_8345, MW_TETOPO, MW_MWTOPO, MW_ETHSVC		
Pre-test conditions	• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT • All the Domain Controller instances are up and running normally • All the devices are upgraded to correct versions • All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.) • All Restconf servers are serviceable. • All L2 data services are successfully created.		
Test Sequence	**Step**	**Type**	**Description**
	1	***Request***	*Send GET request via Postman to all domain controllers by executing Collection TD_MSP_04*
	2	***Validation***	*The response body should no longer contain information about the L2 service deleted in TD_MSP_03*
	3		
	4		

9.4 LLDP Test Descriptions

<table>
<tr><th colspan="4">Interoperability Test Description</th></tr>
<tr><td>Identifier</td><td colspan="3">TD_ LLDP_01</td></tr>
<tr><td>Test Objective</td><td colspan="3">The inter-domain-plug-id value associated to the port used to capture the LLDP packets in TD_SDD_04 is requested from the two DCs under test.

For each DC, this value is compared to the reference value calculated by the reference algorithm, available on the local http server, based on the given couple of DCs under test.

Each DC is hereby tested for compliance with the reference algorithm.</td></tr>
<tr><td>Configuration</td><td colspan="3">LLDP_TEST_CFG</td></tr>
<tr><td>References</td><td colspan="3"></td></tr>
<tr><td>Applicability</td><td colspan="3">MW_LLDP</td></tr>
<tr><td colspan="4"></td></tr>
<tr><td>Pre-test conditions</td><td colspan="3">• Postman has been correctly initialized earlier, by executing TD_POSTMAN_INIT
• The two Domain Controller instances to be tested are up and running normally
• All the devices are upgraded to correct versions
• All basic configurations are completed (e.g., NE_id, OSPF, PCEP, etc.)
• The Restconf servers are serviceable
• The local http server is up and running
• The LLDP packets have been captured in are available at the URLs specified in Table 9 LLDP Captured Packets URL</td></tr>
<tr><td colspan="4"></td></tr>
<tr><td rowspan="7">Test Sequence</td><td>Step</td><td>Type</td><td>Description</td></tr>
<tr><td>1</td><td>Setup</td><td>The two domains (called x and y here) are connected to each other on their inter-domain port West</td></tr>
<tr><td>2</td><td>Request</td><td>The reference inter-domain-plug-id value is calculated by the reference algorithm on http://mWT-local-server/ based on the captured LLDP packets for Domains x and y</td></tr>
<tr><td>3</td><td>Request</td><td>The inter-domain-plug-id for inter-domain port West is requested from Domain x via the RESTconf interface</td></tr>
<tr><td></td><td>Validation</td><td>The retrieved value is compared to the reference one. Test result is Pass for Domain x if they coincide</td></tr>
<tr><td>5</td><td>Request</td><td>The inter-domain-plug-id for inter-domain port West is requested from Domain y via the RESTconf interface</td></tr>
<tr><td>6</td><td>Validation</td><td>The retrieved value is compared to the reference one. Test result is Pass for Domain y if they coincide</td></tr>
</table>

Annex A Interoperability Feature Statement

A.1 Entities

Item	Which entity do you support?	Status	Support
E1	MW SDN Domain Controller	Available	Mandatory

Table 17 Entities

A.2 MW Domain Controller Features

Item	Feature	ID	Ref	Support
1	DC supports Restconf (RFC 8040) on the NBI	MW_8040	[7]	Mandatory
2	DC supports the I2RS DM (RFC 8345) on the NBI as per Annex C	MW_8345	[2]	Mandatory
3	DC supports the TE Topology DM (draft-ietf-teas-yang-te-topo) on the as per Annex C	MW_TETOPO	[3]	Mandatory
4	DC supports the MW Topology DM (draft-ye-ccamp-mw-topo-yang) on the as per Annex C	MW_MWTOPO	[4]	Mandatory
5	DC supports the Ethernet Topology DM (draft-zheng-ccamp-client-topo-yang) on the NBI as per Annex C	MW_ETHTOPO	[5]	Mandatory
6	DC supports the Ethernet Service DM (draft-zheng-ccamp- client-signal-yang) on the NBI as per Annex C	MW_ETHSVC	[6]	Mandatory
7	DC supports calculation of the inter-domain-plug-id from the LLDP information as per algorithm in	MW_LLDP	[11]	Mandatory

Table 18 MW Domain Controller Features

Notes:

1. For the precise reference to the model drafts to be used, please refer to Annex C.1

Annex B IETF Data Model Selection

The subset of IETF Data models, and the subset of parameters thereof, to be used in this Plugtest are specified here.

For convenience, the relevant information is listed here too.

B.1 IETF Data Models Version

The IETF data models are defined in different IETF documents (Internet-Drafts or RFCs) which contain one or more YANG modules each as listed below:

- ietf-microwave-topology@2018-10-22.yang (draft-ye-ccamp-mw-topo-yang-02)
- ietf-eth-te-topology@2018-03-01.yang (draft-zheng-ccamp-client-topo-yang-03)
- ietf-eth-tran-service@2018-08-30.yang (draft-zheng-ccamp-client-signal-yang-02)
- ietf-network@2018-02-26.yang (RFC8345)
- ietf-network-topology@2018-02-26.yang (RFC8345)
- ietf-te-topology@2018-06-15.yang (draft-ietf-teas-yang-te-topo-18) – RFC Queue

- ietf-te-types@2018-06-12.yang (draft-ietf-teas-yang-te-15)
- ietf-eth-tran-types@2018-08-30.yang (draft-zheng-ccamp-client-signal-yang-02)
- ietf-yang-types@2013-07-15.yang (RFC6991)
- ietf-inet-types@2013-07-15.yang (RFC6991)
- ietf-routing-types@2017-12-04.yang (RFC8294)
- ietf-yang-schema-mount@2018-10-16.yang (draft-ietf-netmod-schema-mount-12) – RFC Queue (See Note below)

A reference copy of these files can be found here.

NOTE – The ietf-yang-schema-mount module is needed just to compile the ietf-microwave-topology but it is not to be implemented for the Plugtest (so any version can be used as long as it compiles). The latest version is reported here.

Different YANG modules have different levels of maturity in the standardization process:

- YANG modules are officially released once published in an RFC
- YANG modules defined by Internet-Drafts in RFC Editors' Queue are stable
- YANG modules defined by WG Internet-Drafts (with draft names starting with "draft-ietf") are quite stable
- YANG modules defined by individual Internet-Drafts (with draft names not starting with "draft-ietf") are just individual proposals to IETF and subject to changes during IETF development process

B.2 Tree Diagrams

B.2.1 Microwave Topology Sub-tree

```
+--rw ietf-network:networks
  +--rw ietf-network:network* [network-id]
    +--rw ietf-network:network-id          network-id
    +--rw ietf-network:network-types
    |  +--rw ietf-te-topology:te-topology!
    |  |  +--rw ietf-microwave-topology:mw-topology!
    +--rw ietf-te-topology:provider-id?      te-types:te-global-id
    +--rw ietf-te-topology:client-id?        te-types:te-global-id
    +--rw ietf-te-topology:te-topology-id?   te-types:te-topology-id
    +--rw ietf-te-topology:te!
    | +--rw ietf-te-topology:name?               string
    +--rw ietf-network:node* [node-id]
```

```
   | +--rw ietf-network:node-id            node-id
   | +--rw ietf-te-topology:te-node-id?   te-types:te-node-id
   | +--rw ietf-te-topology:te!
   | | +--rw ietf-te-topology:te-node-attributes
   | | | +--rw ietf-te-topology:name?                  string
   | | +--ro ietf-te-topology:oper-status?               te-types:te-oper-status
   | +--rw ietf-network-topology:termination-point* [tp-id]
   |    +--rw ietf-network-topology:tp-id                          tp-id
   |    +--rw ietf-te-topology:te-tp-id?   te-types:te-tp-id
   |    +--rw ietf-te-topology:te!
   |       +--rw ietf-te-topology:admin-status?
   |       |       te-types:te-admin- status
   |       +--rw ietf-te-topology:name?                            string
   |       +--ro ietf-te-topology:oper-status?
   |       |       te-types:te-oper-status
   +--rw ietf-network-topology:link* [link-id]
      +--rw ietf-network-topology:link-id            link-id
      +--rw ietf-network-topology:source
      | +--rw ietf-network-topology:source-node?   -> ../../../nw:node/node-id
      | +--rw ietf-network-topology:source-tp?     leafref
      +--rw ietf-network-topology:destination
      | +--rw ietf-network-topology:dest-node?   -> ../../../nw:node/node-id
      | +--rw ietf-network-topology:dest-tp?     leafref
      +--rw ietf-te-topology:te-link-attributes
      | +--rw ietf-te-topology:admin-status?
      | |       te-types:te-admin-status
      | +--rw ietf-microwave-topology:mw-link-frequency?          uint32
      | +--rw ietf-microwave-topology:mw-link-channel-separation? uint32
      | +--ro ietf-microwave-topology:mw-link-nominal-bandwidth?  rt-types:bandwidth-ieee-float32
      | +--ro ietf-microwave-topology:mw-link-current-bandwidth?  rt-types:bandwidth-ieee-float32
      | +--rw ietf-microwave-topology:mw-unreserved-bandwidth    rt-types:bandwidth-ieee-float32
      | +--ro ietf-microwave-topology:mw-link-availability*  [availability]
      |    +--ro ietf-microwave-topology:mw-link-availability     rt-types:percentage
      |    +--ro ietf-microwave-topology:mw-link-bandwidth        rt-types:bandwidth-ieee-float32
      +--ro ietf-te-topology:oper-status?               te-types:te-oper-status
```

B.2.2 Ethernet Topology Sub-tree

```
+--rw ietf-network:networks
  +--rw ietf-network:network* [network-id]
    +--rw ietf-network:network-id            network-id
    +--rw ietf-network:network-types
    |  +--rw ietf-te-topology:te-topology!
    |  |  +--rw ietf-eth-te-topology:eth-tran-topology!
    +--rw ietf-te-topology:provider-id?      te-types:te-global-id
    +--rw ietf-te-topology:client-id?        te-types:te-global-id
    +--rw ietf-te-topology:te-topology-id?   te-types:te-topology-id
    +--rw ietf-te-topology:te!
    | +--rw ietf-te-topology:name?                   string
    +--rw ietf-network:node* [node-id]
    | +--rw ietf-network:node-id            node-id
    | +--rw ietf-te-topology:te-node-id?   te-types:te-node-id
    | +--rw ietf-te-topology:te!
    | | +--rw ietf-te-topology:te-node-attributes
    | | | +--rw ietf-te-topology:name?                  string
    | | +--ro ietf-te-topology:oper-status?               te-types:te-oper-status
    | +--rw ietf-network-topology:termination-point* [tp-id]
    |    +--rw ietf-network-topology:tp-id                          tp-id
    |    +--rw ietf-te-topology:te-tp-id?   te-types:te-tp-id
    |    +--rw ietf-te-topology:te!
    |    | +--rw ietf-te-topology:admin-status?
```

```
|     | |      te-types:te-admin- status
|     | +--rw ietf-te-topology:name?                    string
|     | +--rw inter-domain-plug-id?          binary
|     | +--ro ietf-te-topology:oper-status?
|     | |      te-types:te-oper-status
|     +--rw ietf-eth-te-topology:svc!
|        +--rw ietf-eth-te-topology:client-facing?           boolean
+--rw ietf-network-topology:link* [link-id]
| +--rw ietf-network-topology:link-id          link-id
| +--rw ietf-network-topology:source
| | +--rw ietf-network-topology:source-node?   -> ../../../nw:node/node-id
| | +--rw ietf-network-topology:source-tp?     leafref
| +--rw ietf-network-topology:destination
| | +--rw ietf-network-topology:dest-node?   -> ../../../nw:node/node-id
| | +--rw ietf-network-topology:dest-tp?     leafref
| +--rw ietf-te-topology:te-link-attributes
| | +--rw ietf-te-topology:underlay {te-topology-hierarchy}?
| | | +--rw ietf-te-topology:enabled?                  boolean
| | | +--rw ietf-te-topology:primary-path
| | | | +--rw ietf-te-topology:network-ref?    ->
/nw:networks/network/network-id
| | | | +--rw ietf-te-topology:path-element* [path-element-id]
| | | |    +--rw ietf-te-topology:path-element-id    uint32

| | | |    +--rw ietf-te-topology:index?           uint32
| | | |    +--rw (ietf-te-topology:type)?
| | | |       +--:(ietf-te-topology:num-unnum-hop )
| | | |       | +--rw ietf-te-topology:num-unnum-hop
| | | |       |    +--rw ietf-te-topology:node-id?      te-types:te-node-id
| | | |       |    +--rw ietf-te-topology:link-tp-id?   te-types:te-tp-id
| | | |       |    +--rw ietf-te-topology:hop-type?     te-hop-type
| | +--rw ietf-te-topology:admin-status?
| | |      te-types:te-admin-status
| | +--rw ietf-eth-te-topology:max-bandwidth?         uint64
| | +--rw ietf-eth-te-topology:available-bandwidth?    uint64
| +--ro ietf-te-topology:oper-status?              te-types:te-oper-status
```

B.2.3 Ethernet Service Sub-tree

```
module: ietf-eth-tran-service
   +--rw etht-svc
      +--rw globals
      | +--rw etht-svc-bandwidth-profiles* [bandwidth-profile-name]
      |    +--rw bandwidth-profile-name    string
      |    +--rw bandwidth-profile-type?   etht-types:bandwidth-profile-type
      |    +--rw CIR?                   uint64
      |    +--rw EIR?                   uint64
      |    +--rw color-aware?             boolean
      |    +--rw coupling-flag?           boolean
      +--rw etht-svc-instances* [etht-svc-name]
         +--rw etht-svc-name           string
         +--rw etht-svc-type?          etht-types:service-type
         +--rw access-provider-id?      te-types:te-global-id
         +--rw access-client-id?       te-types:te-global-id
         +--rw access-topology-id?      te-types:te-topology-id
         +--rw etht-svc-access-ports* [access-port-id]
         | +--rw access-port-id                    uint16
         | +--rw access-node-id?                    te-types:te-node-id
         | +--rw access-ltp-id?                    te-types:te-tp-id
         | +--rw service-classification-type?           identityref
         | +--rw (service-classification)?
         | | +--:(vlan-classification)
```

```
| |     +--rw outer-tag!
| |     | +--rw tag-type?     etht-types:eth-tag-classify
| |     | +--rw (individual-bundling-vlan)?
| |     |    +--:(individual-vlan)
| |     |    | +--rw vlan-value?   etht-types:vlanid
| |     +--rw second-tag!
| |        +--rw tag-type?     etht-types:eth-tag-classify
| |        +--rw (individual-bundling-vlan)?
| |           +--:(individual-vlan)
| |           | +--rw vlan-value?   etht-types:vlanid
| +--rw (direction)?
| | +--:(symmetrical)
| | | +--rw ingress-egress-bandwidth-profile-name?   string
+--rw admin-status?           identityref
+--ro state
  +--ro operational-state?         identityref
   +--ro provisioning-state?        Identityref
```

Annex C JSON Code

C.1 Microwave Topology

Please refer to this file.

C.2 Ethernet Topology

Please refer to this file.

C.3 Ethernet Service

Please refer to this file.

Annex D Test Calendar

This section is preliminary

Monday 9th	Tuesday 10th	Wednesday 11th	Thursday 12th	Friday 13th
Setup	SINGLE tests LLDP tests	TREE tests Repeats	Repeats Wrap up	(WI 24) (WI 25) Pack
Setup	SINGLE tests LLDP tests	TREE tests Repeats	Repeats Wrap up De-brief	Report Pack
Setup Free tests	SINGLE tests LLDP tests	TREE tests Repeats	De-brief Report	Pack
Setup Free tests	SINGLE tests LLDP tests	TREE tests Repeats	Report	Close down

Table 19 High Level Test Calendar

Change History

Document history		
0.1	05.11.2019	First draft
0.2	26.11.2019	Changed chapter structure, modified Scope, Architecture, Configurations, edited Annex C, added Annex F
0.3	27.11.2019	Renamed the Test IDs and updated accordingly. Added skeletons for domain rotation and LLDP
0.5	06.12.2019	Added details about LLDP testing, other minor corrections and improvements
0.6	16.12.2019	Moved appendix B and F to different documents
0.7		Internal step
0.8	14.01.2020	Introduced the two possible L2 service representation on aggregated links. Correspondently updated the feature list, the test summary and the test descriptions chapters.
0.8.1	15-01-2020	Corrected Table 6
0.9	24-01-2020	New Service representation and test connectivity – remade chapters 7, 8 sand 9
0.10	31-01-2020	Updated according to Confcall #4
0.11	05-02-2020	Added Annex D, other minor corrections
0.12	07-01-2020	Added tables 1 to 7, modified table 8, rationalized C-VLAN and S-VLAN numbering
0.13	14-02-2020	Interim version
0.14	14-02-2020	Added 3 local services in SINGLE_CFG, corrected the wording CIR/PIR to CIR/EIR, other minor

End of Document