**Addressing questions raised on Service Termination Points and their abstraction/use within the NSI CS protocol**

We have been having a debate in the NSI group lately on what STP’s stand for and how they should be used within the NSI CS protocol.

STP background:

STP is a topological abstraction that represents a topological point to which a connection can be setup. Only the network that owns the STP can resolve it into the physical topological end-point. Also STP string may not be parsed to get physical topological information for path-finding or CS purposes.

The question that has been raised repeatedly is what abstraction do STP’s represent and for examples on how they would be used in a network, fully well knowing that corresponding topology and path-finding service have yet to be defined. The proposal below addresses the above question and addresses how the multi-layer technology specific parameters can be abstracted and addressed with STPs. Even though we have had many discussions on the STP concepts and what they represent, the exact mechanics and specifics have not been agreed upon. One reference is the following document from Tomohiro:

<http://forge.gridforum.org/sf/go/doc16046?nav=1>

Under-specification of STP

The connection service protocol is a generic method to setup an end-to-end connection between two (or “n” in the future versions) points on the network. The Requestor agent (RA) makes a service request to the Provider Agent (PA) and specifies the two Service Termination points – point A where the service connection starts and point Z where the service connection ends. Each network can have different technologies and end-points that are truly multi-layer i.e. optical/SONET/OTN, Ethernet, MPLS, or IP service. The reason we call this a connection service because it has certain characteristics guaranteed between point A and Z of the connection that could include bandwidth, latency, specific technology parameters, path, etc. How the network actually leverages encapsulations and various network technologies to provide that service is not of concern to the connection services protocol.

The question remains on what the abstract representation of STP should map to in the physical topology domain.

**Proposal**

We propose that there should be a capability for the network to under-specify the STP or another way is that it represents multiple slices of the topology to allow for the multiplexing of multiple connections over the same physical port. The technology specific parameters will be specified in detail in the Service Request/Service Definition.

For example:

RAa 🡨-🡪 PAa 🡨 🡪 PAz 🡨 🡪 RAz

Lets assume that RAa wants to talk with RAz. RAa sends the service request to PAa.

Case A – partially specified to fully specified.

STPa maps to switchA/portA

STPz maps to switchZ/portZ/vlan1000

RAa sends a service request that asks for SD {2 Gs of bandwidth with Ethernet service, {optional VLAN: 100 – 2000} } between STPa and STPz.

***Tree Model***

The pathfinder of PAa first resolves that STPz belongs to NSA PAz.

It then forms two service requests:

1. SD {2G, Ethernet, VLAN600} STPa to STPa’{VLAN600}
2. SD {2G, Ethernet} STPz’ to STPz

STPa’ = DomainA/switchA’/portA’

STPz’ = DomainZ/switchZ’/portZ’/VLAN600

In this case the Z network also has to deploy at TransferFunction to translate the VLAN before delivering it to STPz.

The reason why this is needed is because, the network chose a VLAN that suited its needs based on the range of STP’s possible at point A. The customer also did not have to specify a certain specific STP as he has flexibility and no direct knowledge of VLANs through the STP. All he needs to know which VLAN was picked for that connection.

If we did not do it this way, the user would be given upto 2^24 (if you use IEEE SPB) arbitrary STP strings and would be told to choose on STP. The advertisement of those possible STPs to networks and size of the topoDB will be unmanageable.

This same technique of under-specification, with optional connection specific parameters works well for all scenarios, technologies (SONET timeslots) and even the new concepts of “flows” that work on 11-tuple specifications of the connection.

Given the wide variation of tunneling techniques and possibilities for connection services, having a flexible STP mechanism assures that each network can choose what they want, to fully specify or under-specify and does not place any restrictions.

Note:

For the STP’s between domains, network administrators may choose to fully qualify their STPs or have exact range matches i.e. between a’ and z’ to avoid complexity. The pathfinding and topology capability at NSA PAa above narrows down the VLAN to 600 based on the availability of VLANs between a’ and z’. In case there is no overlap, the network will have to configure or route the connection through a transfer function that does VLAN translation or reject the connection or choose another encapsulation.

**Chain Model**

The pathfinder of PAa first resolves that STPz belongs to NSA PAz.

It then forms two service requests:

1. SD {2G, Ethernet, VLAN200} STPa to STPa’{VLAN200}
2. SD {2G, Ethernet} STPb{VLAN200} to STPb’{VLAN200}
3. SD {2G, Ethernet} STPz’ to STPz

STPa’ = DomainA/switchA’/portA’

STPb = DomainB/SwitchB/portB

STPb’= DomainB/SwitchB’/portB’

STPz’ = DomainZ/switchZ’/portZ’/VLAN600