GFD-I.217
NSI-WG
nsi-wg@ogf.org

John MacAuley, ESnet
Chin Guok, ESnet
Guy Roberts, GEANT
May 1, 2015

Network Service Interface Signaling and Path Finding

Status of This Document
Grid Forum Document – Informational (GFD-I)

Abstract
This document provides a discussion of the signaling and path finding models supported by the NSI protocol.

Contents

1 Introduction

The NSI Service Framework defines a Transport Plane and a Service Plane. The connectivity services are built using the resources available in the Transport Plane. NSI messages are exchanged on the Service Plane. The NSI messages follow a workflow that has been designed to be independent of the Transport Plane connectivity.

The NSI framework provides the flexibility to support workflows as simple as a chain or as complex as multi-level trees. The fundamentals of NSI that support this flexibility can be found in the OGF NSI Framework document [1]. To provide clarity on the various scenarios for signaling and path finding, the following sections outline specific workflows and their requirements.

2 Signaling and Path Finding Models

The model for signaling and path finding have been driven by the following architectural requirements that were decided upon early in the inception of the NSI working group. These requirements have been driven by the need to support a flexible Service Plane that is topologically decoupled from the Transport Plane. Recent developments in Software Defined Networking (SDN) have confirmed the importance of decoupling of the Service and Transport Planes.

1. The Network Service Agent (NSA) Service Plane topology does not need to be congruent with Transport Plane topology.
2. Not all NSAs will be directly interconnected through a Service Plane peering between NSAs. Pair-wise peering arrangements will dictate the Service Plane topology. NSA Service Plane inter-connectivity will be guided by security and administration considerations and NOT exclusively Transport Plane considerations. Therefore, a requester NSA may not have a direct Service Plane peering to the complete set of Provider Agents (PAs) involved in a reservation.
3. Users may request a reservation from an NSA (i.e. an Aggregator NSA (AG)) that is not directly managing resources in the Transport Plane. Based on item #2 above, this AG may not have direct Service Plane peering with all the NSAs involved in the reservation request.
4. Users may request reservations between endpoints that are not in their Network, or the Network of their NSA. This implies the user request may not originate from the NSA managing the source end of the Connection.

These all imply that a reservation request in a tree workflow may pass through many AGs on its way to instantiating Connection segments on the children uPAs. If the AG at the top of the request tree is to perform its duties, then it will need to understand the Service Plane topology, AGs within the Network that can route reservation requests, uPAs managing specific Networks, and of course, Transport Plane topology. Now for the most important bit: this top-level AG *can only get all this information through its direct peers*.

The following sections make a clear distinction between tree and chain workflows even though a chain workflow can be considered to be a degenerate case of a tree workflow. The reason for this is to highlight certain concessions that can be made when considering the chain workflow independently.

2.1 Tree-based signaling and path finding model

To support the NSI tree-based workflow (see Fig 1.), the following must be supported:

1. A Network Service request can be initiated at an AG anywhere in the service region, and this AG does not have to be associated with any Network resources involved in the creation of the service.
2. A Network Service request can be propagated to as many NSAs as there are sub-connection segments. The AG receiving the request does not need to fully resolve the path of the Connection, but can delegate the task further down the tree to other AGs.

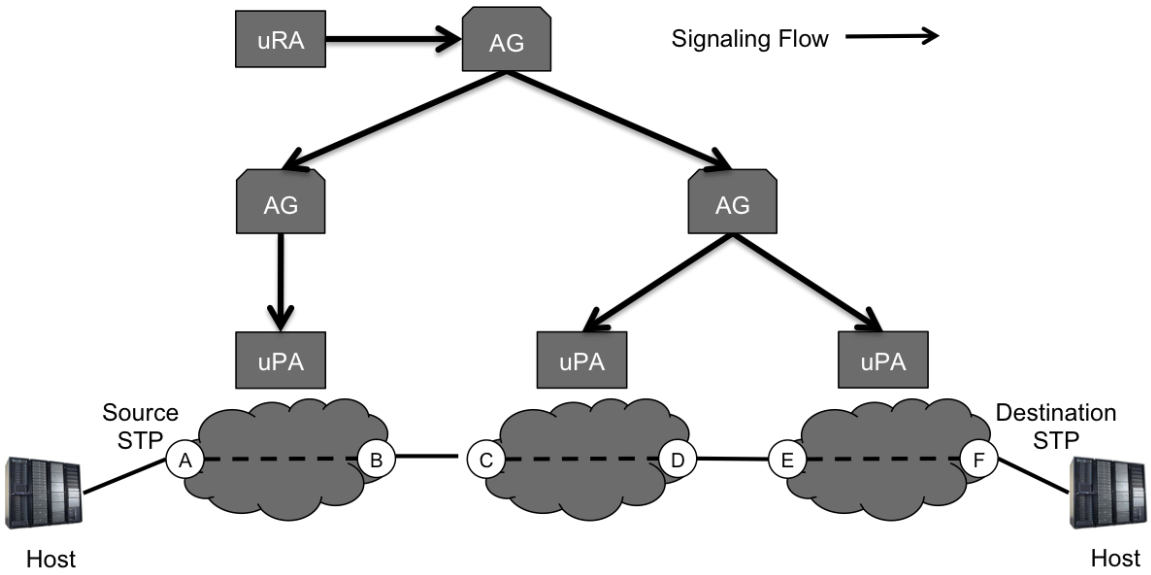

Figure 1. Tree-based signaling flow

In order to support the above two premises, the following requirements are necessary:

1. The Aggregator NSA needs a full view of the Network topology to perform advanced "intelligent" routing decisions. At a minimum, the Aggregator NSA needs all the Networks, the Transport Plane peering ports (to derive SDP), the services offered, associated service domains (Switching Services), and any adaptations within the Network.
2. An aggregator may hide Network details of child Networks if desired, and advertise a summarized NSI Topology as needed. The key with this model is that external NSAs need not know the details of these internal child NSAs.
3. An Aggregator NSA is restricted to communicating with only direct peer NSAs based on administrative policies, however, *it needs access to **all** NSA Description Documents and NSI Topology documents to perform tree based routing*. In addition to peer communications, the NSA Description Documents and NSI Topology documents are needed to determine the NSA managing Networks and the Service Plane topology to determine routing paths of requests in the connected graph. The NSI Topology Documents from each Network are then used to build NSI Topology for the service region for routing of the Connection. The security, verification, and distribution of the NSA Description Documents and NSI Topology documents are outside the scope of this document and are addressed in separate NSI working group efforts [3].

The specific requirement that an NSA would only directly peer with a subset of NSAs, and not be able to directly communicate with all NSAs is due to administrative limitations and scaling. Provisioning a full mesh of trust relationships between all NSAs is considered prohibitive, and anonymous retrieval of the NSI documents have negative security implications. This results in the need for Service Plane topology so that trust relationships between NSAs can be discovered, and NSI description information can be distributed to all NSAs within the service region.

2.1.1 Tree routing

In a tree-based source routing workflow, it is typical for the (root) AG receiving the request from the uRA to perform path finding and segmentation using NSI Topology for the whole service region the route the requests to the appropriate NSAs accordingly. Figure 2 shows an example of such a workflow with the associated Path Computation Engines (PCEs):

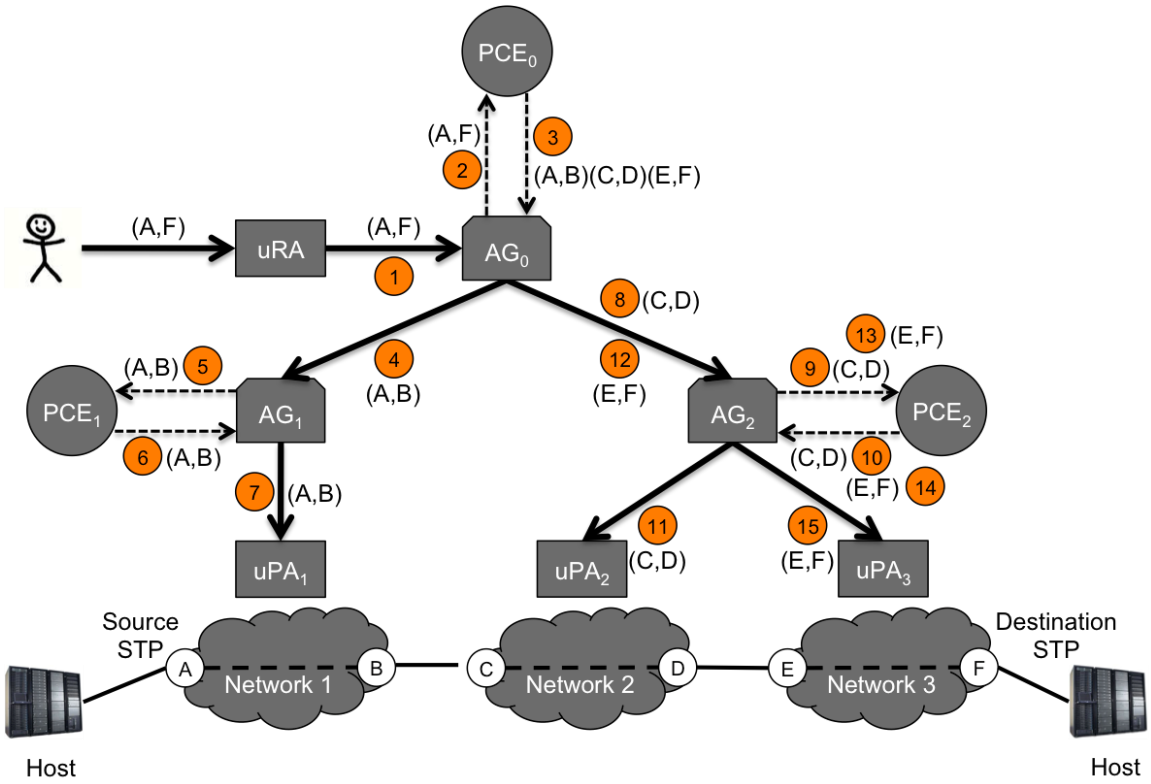

Figure 2. Example of a tree-based source routing workflow

1. The uRA makes a reservation request to the root Aggregator AG_0 on behalf of the user to Connect STP A to STP F.
2. AG_0 receives the request (A,F), and after validation, performs path finding to determine an end-to-end path through the various networks.
3. PCE_0 uses NSI Topology over the whole service region to determine an explicit Transport Plane path for the request, returning three Connection segments (A,B)(C,D)(E,F).
4. AG_0 then sends the Connection request for segment (A,B) to AG_1 which represents the only signaling path to uPA_1.
5. AG_1 receives the request (A,B), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation.
6. PCE_1 determines that both the source and destination STP are managed by uPA_1 and returns the Connection segment (A,B).
7. AG_1 then makes a reservation request to the uPA_1 for the local Connection segment (A,B).
8. With uPA_2 having only a trust relationship with AG_2, AG_0 sends the request to uPA_2 for Connection segment (C,D) via AG_2.
9. AG_2 receives the request (C,D), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation.
10. PCE_2 determines that both the source and destination STP are managed by uPA_2 and returns the Connection segment (C,D).
11. AG_2 then makes a reservation request to the uPA_2 for the local Connection segment (C,D).
12. To complete the end-to-end Connection, AG_0 sends the request for Connection segment (E,F) to AG_2 as it is the only signaling path to uPA_3.
13. AG_2 receives the request (E,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation.
14. PCE_2 determines that both the source and destination STP are managed by uPA_3 and returns the Connection segment (E,F).

15. AG_2 then makes a reservation request to the uPA_3 for the local Connection segment (E,F).

NB: It should be noted that since Connection segments (C,D) and (E,F) are distinct, steps 8-11 and 12-15 can be done in parallel. This can be generalized to anywhere an AG has multiple children

2.2 Chain-based signaling and path finding model

Chain signaling with the NSI CS protocol requires every NSA to be an AG capable of propagating a reservation request to the local uPA component (associated with local Network resources) and at most one adjacent (child) NSA associated with the next Connection segment in the data path. In general, a Connection request issued by a uRA should be to the AG associated with the head-end STP of the service, as the service request will only be signaled in one direction, from the NSA associated with source STP through to the NSA associated with the destination STP. If a uRA issues a Connection request to an arbitrary AG, the AG must first route the request to the AG associated with the head-end STP of the service. It is important to note that if the request was initially issued to an intermediate AG (i.e. an AG in the middle of the chain), the intermediate AG will maintain two records of the reservation, the first to track the routing of the request message to the head-end AG, and the second to manage the request for resources and the forwarding of the message to the downstream AG in the chain.

All NSAs in the chain request must contain a Transport Plane Connection segment associated with the reservation request. An NSA in the chain must have a peering relationship with all NSAs that it is connected to at the data plane. This implies that the Service Plane MUST be congruent with the Transport Plane. Figure 3.3 shows this basic chain signaling flow and NSA components involved in the deployment.

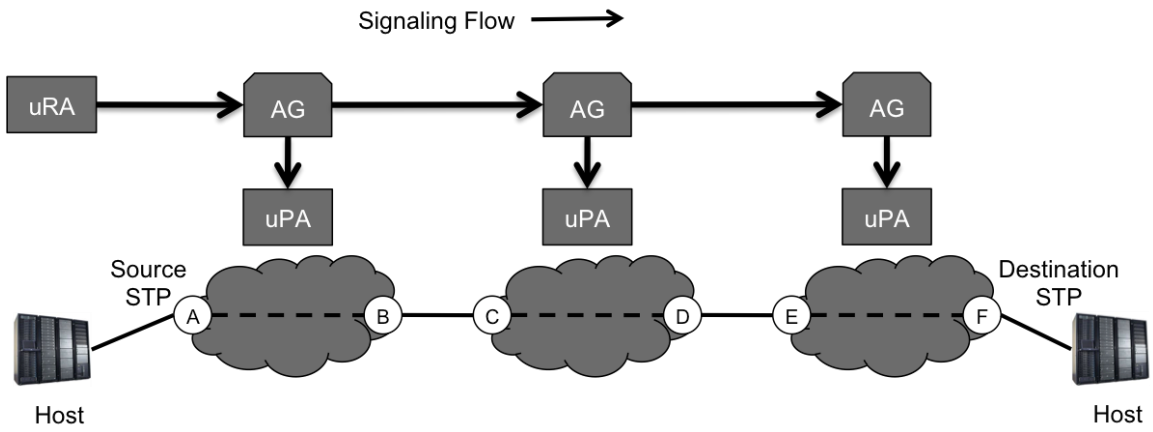

Figure 3. Chain-based signaling flow

The NSI CS 2.0 architecture currently supports two path-finding models for chain-based deployments: Hop-by-hop routing, and source routing. These are discussed in the next two sections.

2.2.1 Hop-by-hop routing

Hop-by-hop routing is a chain-based solution that makes a localized routing decision at each NSA along the service path while using the NSI Topology to guide next hop decisions. Figure 4 shows the following hop-by-hop routing workflow:

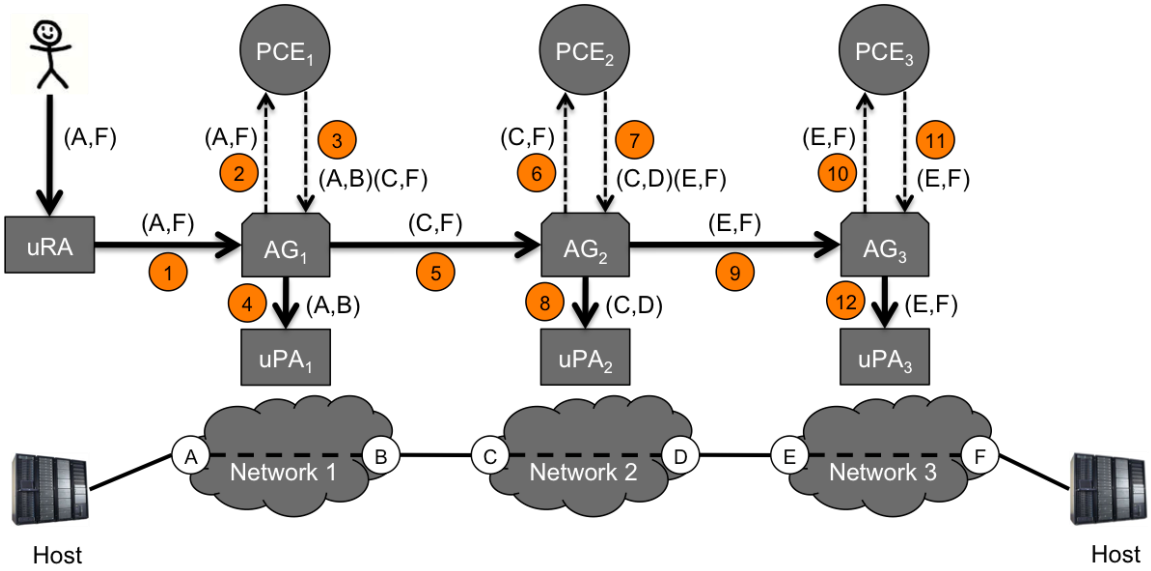

Figure 4. Example of a chain signaling workflow using hop-by-hop routing

1. The uRA make a reservation request to the head-end Aggregator AG_1 on behalf of the user to interconnect STP A to STP F. This request is made to the head-end Aggregator NSA associated with the source STP in Network 1.
2. AG_1 receives the request (A,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation, and therefore, the peer NSA that will be next in the chain.
3. The PCE_1 uses NSI Topology over the whole service region to determine a loose path for the request, returning the local Connection segment (A,B), and the remaining path segment in the form of two STPs: the ingress STP in the adjacent Network 2 and the original destination STP (C,F).
4. AG_1 makes a reservation request to the local uPA_1 for the local Connection segment (A,B).
5. AG_1 then sends a new reservation request to the next NSA (Network 2) in the chain for the remaining Connection segment (C,F).
6. AG_2 receives the request (C,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation, and therefore, the peer NSA that will be next in the chain.
7. PCE_2 also uses the NSI Topology over the whole service region to determine a loose path for the request, returning the local Connection segment (C,D), and the remaining path segment in the form of two STPs: the ingress STP in adjacent Network 3 and the original destination STP (E,F).
8. AG_2 makes a reservation request to the local uPA_2 for the local Connection segment (C,D).
9. AG_2 then sends a new reservation request to the next NSA (Network 3) in the chain for the remaining Connection segment (E,F).
10. AG_3 receives the request (E,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation.
11. PCE_3 determines that both the source and destination STP are within the local Network and returns the local Connection segment (E,F). The chain is complete and no further segments are returned.
12. AG_3 makes a reservation request to the local uPA_3 for the local Connection segment (E,F).

2.2.2 Source routing

Source routing is a chain-based solution where the head-end Aggregator NSA (or uRA) uses The NSI topology for the whole service region to partially or completely specify a path the Connection will take through the network. This detailed path information is passed from NSA to NSA along signaling path using an Explicit Route Object (ERO) within the service request. Each NSA is bound by the ERO to follow the path segments specified during its own path finding activities. If an NSA along the path cannot meet the constraints specified in the ERO, the reservation request is rejected and an error is returned to the head-end Aggregator (or uRA) that can attempt an alternative path. Figure 5 shows the following source routing workflow:

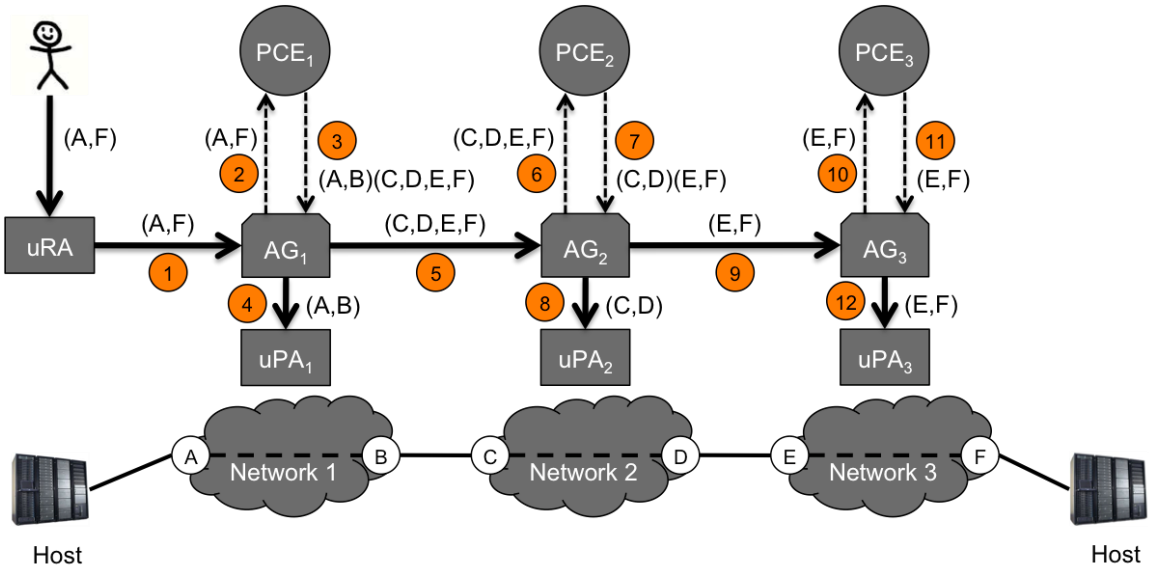

Figure 5. Example of a chain signaling workflow using source routing

1. The uRA make a reservation request to the head-end Aggregator AG_1 on behalf of the user to interconnect STP A to STP F. This request is made to the head-end Aggregator NSA associated with the source STP in Network 1.
2. AG_1 receives the request (A,F), and after validation, performs path finding to determine an end-to-end path through the network.
3. The PCE_1 uses NSI Topology over the whole service region to determine an explicit path for the request, returning the local Connection segment (A,B), and the remaining path segment in the form of a set of STP in other Network to include in the service (C,D,E,F). The egress STP B from Network 1 is connected to the ingress STP C in the adjacent Network 2, and therefore, the peer NSA that will be next in the chain.
4. AG_1 makes a reservation request to the local uPA_1 for the local Connection segment (A,B).
5. AG_1 then sends a new reservation request to AG_2 in the chain for the remaining Connection segments (C,D,E,F).
6. AG_2 receives the request (C,D,E,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation, and therefore, the peer NSA that will be next in the chain.
7. PCE_2 uses NSI Topology over the whole service region to determine path for the request, returning the local Connection segment (C,D), and the remaining path segment (E,F).
8. AG_2 makes a reservation request to the local uPA_2 for the local Connection segment (C,D).
9. AG_2 then sends a new reservation request to AG_3 in the chain for the remaining Connection segment (E,F).

10. AG_3 receives the request (E,F), and after validation, performs path finding to determine which local STP should be involved in the Connection reservation.
11. PCE_3 determines that both the source and destination STP are within the local domain, so returns the local Connection segment (E,F). The chain is complete and no further segments are returned.
12. AG_3 makes a reservation request to the local uPA_3 for the local Connection segment (E,F).

3 Conclusion

The above sections have outlined several rudimentary examples of how an end-to-end path can be segmented and corresponding requests routed to the appropriate NSAs. It does not however dictate the algorithm that the path finder uses to determine the "optimal" path and segments, which is beyond the scope of this document. It should be noted however that in all the schemes discussed above, it is necessary to apply the path computation against the NSI Topology over the whole service region topology and not simply partial fragments of topology.

4 Security Considerations

Security considerations are dealt with in Open Grid forum GWD-R draft-trompert-gwdi-nsi-aa-v04, NSI Authentication and Authorization [3].

No additional security issues have been raised.

5 Contributors

Chin Guok, ESnet
Tomohiro Kudoh, AIST
John MacAuley, SURFnet
Guy Roberts, GEANT
Henrik Thostrup Jensen, NORDUnet
Hans Trompert, SURFnet

6 Intellectual Property Statement

7 Disclaimer

of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

8 Full Copyright Notice

9 Glossary

The following terms are defined in NSI:

Aggregator (AG)	The Aggregator is an NSA that has more than one child NSA, and has the responsibility of aggregating the responses from each child NSA.
Connection	A Connection is an NSI construct that identifies the physical instance of a circuit in the Transport Plane. A Connection has a set of properties (for instance, Connection identifier, ingress and egress STPs, capacity, or start time). Connections can be either unidirectional or bidirectional.
Connection Service (CS)	The NSI Connection Service is a service that allows an RA to request and manage a Connection from a PA.
Connection Service Protocol	The Connection Service Protocol is the protocol that describes the messages and associated attributes that are exchanged between RA and PA.
NSA Description Document	The NSA Description Document is a document that provides information about the services and features supported by an NSA.
ERO	An Explicit Routing Object (ERO) is a parameter in a Connection request. It is an ordered list of STP constraints to be used by the inter-Network pathfinder.
Network	A Network is an Inter-Network topology object that describes a set of STPs with a Transfer Function between STPs.
Network Services	Network Services are the full set of services offered by an NSA. Each NSA will support one or more Network Services.

Network Service Agent (NSA)	The Network Service Agent is a concrete piece of software that sends and receives NSI Messages. The NSA includes a set of capabilities that allow Network Services to be delivered.
Network Services Framework (NSF)	The Network Services framework describes an NSI message-based platform capable of supporting a suite of Network Services such as the Connection Service and the Topology Service.
Network Service Interface (NSI)	The NSI is the interface between RAs and PAs. The NSI defines a set of interactions or transactions between these NSAs to realize a Network Service.
NSI Message	An NSI Message is a structured unit of data sent between an RA and a PA.
NSI Topology	The NSI Topology defines a standard ontology and a schema to describe network resources that are managed to create the NSI service. The NSI Topology as used by the NSI CS (and in future other NSI services) is described in: GWD-R-P: Network Service Interface Topology Representation [3].
Requester/Provider Agent (RA/PA)	An NSA acts in one of two possible roles relative to a particular instance of an NSI. When an NSA requests a service, it is called a Requester Agent (RA). When an NSA realizes a service, it is called a Provider Agent (PA). A particular NSA may act in different roles at different interfaces.
Service Demarcation Point (SDP)	Service Demarcation Points (SDPs) are NSI topology objects that identify a grouping of two Edge Points at the boundary between two Networks.
Service Termination Point (STP)	Service Termination Points (STPs) are NSI topology objects that identify the Edge Points of a Network in the intra-network topology.
Service Plane	The Service Plane is a plane in which services are requested and managed; these services include the Network Service. The Service Plane contains a set of Network Service Agents communicating using Network Service Interfaces.
Topology Distribution Description ServiceDocument	The NSI Topology distribution Service description document contains a description of the Network topology. This document is available to be distributed as a way of sharing topology information between trusted NSAs. is a service that allows the NSI topology to be exchanged between NSAs.
Transport Plane	The Transport Plane refers to the infrastructure that carries the physical instance of the Connection, e.g. the Ethernet switches that deliver the Connection.
Ultimate PA (uPA)	The ultimate PA is a Provider Agent that has an associated NRM.
Ultimate RA (uRA)	The Ultimate RA is a Requester Agent is the originator of a service request.

10 References

1. OGF GFD-R-212, Network Service Interface Connection Service, v2.0 http://www.ogf.org/documents/GFD.212.pdf
2. Open Grid forum GFD-I-213, Network Services Framework v2.0, http://www.ogf.org/documents/GFD.213.pdf
3. Open Grid forum GWD-R draft-trompert-gwdi-nsi-aa-v04, NSI Authentication and Authorization, https://redmine.ogf.org/dmsf_files/13398?download=