

# Control Plane Architecture and Design Considerations for Multi-Service, Multi-Layer, Multi- Domain Hybrid Networks

Tom Lehman<sup>1</sup>, Xi Yang<sup>1</sup>, Chin P. Guok<sup>2</sup>, Nageswara S. V. Rao<sup>3</sup>, Andy Lake<sup>4</sup>, John Vollbrecht<sup>4</sup>, Nasir Ghani<sup>5</sup>

<sup>1</sup>Information Sciences Institute East, University of Southern California, Arlington, VA 22203, USA,  
Email: {tlehman,xyang}@isi.edu

<sup>2</sup>Network Engineering Services Group, ESnet, Berkeley, CA 94720, USA, Email: chin@es.net

<sup>3</sup>Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA,  
Email: raons@ornl.edu

<sup>4</sup>University Corporation for Advanced Internet Development, Internet2, Ann Arbor, MI 48104, USA,  
Email: {jrv,alake}@internet2.edu

<sup>5</sup>Department of Electrical and Computer Engineering, Tennessee Technological University, Cookeville, TN 38505, USA,  
Email: nghani@tntech.edu

## I. INTRODUCTION

The hybrid network architecture promises the combined advantages of both the current best-effort Internet Protocol (IP) service and dedicated deterministic end-to-end network services. While the details of "deterministic services" are under active discussion and development at this time, they being are provisioned fundamentally as circuits. The vision for these hybrid networks is to enable flexible and dynamic provisioning of these services to empower e-Science and other large-scale networked applications to carry out tasks such as massive data transfers, remote interactive visualizations, and monitoring and steering of computations on supercomputers. Such tasks require hybrid network capabilities that can only be achieved by innovating and advancing the network services in a manner not possible on current network infrastructures.

A critical enabling technology to realize this vision is a control plane which allows for provisioning of services in this multi-service, multi-layer, multi-domain hybrid network environment. The multi-service aspect refers to the capability to provide a variety of connection modalities such as Ethernet, SONET, and InfiniBand. The multi-layer aspect refers to the fact end-to-end service may be instantiated via a data plane path which traverses multiple different network elements that belong to different technology layers. The multi-domain aspect refers to establishing services across multiple administrative domains to provide the largest value to end users and applications. While current packet-switched networks are uniform in that routers are key elements, the connection-oriented networks continue to be disparate. For, example, Energy Sciences Network (ESnet) [1] provides tunnels over an IP routed network using Multiple Protocol Label Switching (MPLS), and UltraScience Net (USN) [2] and CHEETAH [3] provide Synchronous Optical Network (SONET) switched networks using TL1/CLI and Generalized MPLS (GMPLS), respectively. The Internet2 Networks [4], HOPI and Dynamic Circuit Services (DCS), provides Ethernet-switched and SONET services, respectively, using the DRAGON [5] GMPLS control plane. The key observation is that the emerging hybrid network infrastructure will be built

out of best practices from various current networks, and consequently will likely be extremely heterogeneous in nature at both the data plane and control plane levels. We therefore, propose to integrate the various control planes into a "service plane", which allows heterogeneous administrative domains and technology regions to understand and accommodate one another's service requirements.

In this paper we discuss key architecture and design considerations associated with the development of a control plane capable of dynamic provisioning in this heterogeneous multi-domain, multi-layer, multi-service hybrid network environment. We present a framework for addressing the heterogeneous nature of the hybrid networks via the development of a flexible set of mechanisms which address the key control plane functions of routing, path computation and signaling. An interoperable set of constructs are proposed based on GMPLS and Web Service for seamless provisioning across heterogeneous data and control planes. This paper also includes a discussion of our recent design and implementation efforts to instantiate these concepts on ESnet [1], USN [2], and the Internet2 Networks [4].

## II. HYBRID NETWORK CONTROL PLANES - ISSUES AND SOLUTION APPROACH

In a heterogeneous hybrid network, a given end-to-end service may be provisioned using one or more of the following data plane technology layers: i) IP router based MPLS tunnels, ii) Ethernet VLAN based circuits, iii) Synchronous Optical Network / Synchronous Digital Hierarchy (SONET/SDH) circuits, iv) Wavelength Division Multiplexing (WDM) connections. At the control plane level, administrative domain specific control planes may be based on a variety of technologies including GMPLS (as being defined in the IETF CCAMP [6] and OIF [7] communities), Centralized Management systems, and native Web Services based systems.

Our solution approach is rooted in the realization that different networks and administrative domains will implement different network data plane and control plane technologies

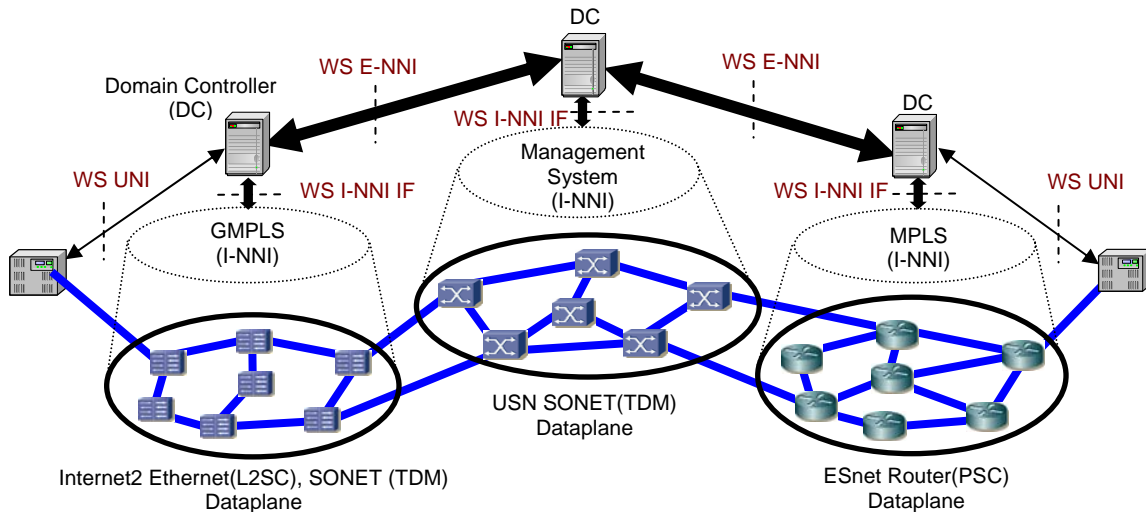


Figure 1 Multi-Domain Web Service Architecture

that best suite their situation based on factors such as performance, cost, available physical resources (such a fiber plants), current equipment, vendor relationships and user requirements. To effectively peer and interoperate such diverse networks, the key capability of the control plane is the definition of Inter-Domain Communications (IDC). This is in line with the Automatically Switched Optical Network (ASON) [8] model that focus on communications between domains as opposed to intra-network operations.

Figure 1 shows an architectural model which combines the ASON model of domain autonomy with Web Services to provide a uniform IDC mechanism for the currently available and evolving multi-domain control plane protocols. This approach allows us to take advantage of the current and future work in the standards bodies and also focus on the critical issues of scalability, security, flexible application of policy, general AAA, and scheduling.

The key control plane architectural components in our solution approach are the Web Service User to Network Interface (WS UNI), External-Network Network Interface (WS E-NNI), and Internal-Network Network Interface (WS I-NNI IF). The WS-UNI defines client service request structures. The WS E-NNI defines web service functions for inter-network routing/topology exchange, path computation/scheduling, and signaling/path setup. While these capabilities are inspired by the standards on control plane protocols, they do more than simply recreating that work at the web service level. They create a service plane by composing various individual control planes. Each domain will be served by one or more web services that present a uniform IDC mechanism described in Web Services Description Language (WSDL). The WS I-NNI IF provides the mechanism for the domain controller to interact with the domain internal resource management and provisioning systems. The details of these internal mechanisms are strictly up to the individual domains.

As noted in Figure 1, we are working in an environment where each of three networks utilizes different mechanisms. The Internet2 Networks utilized a GMPLS model for multilevel provisioning across Ethernet and SONET regions, the USN utilizes a centralized management system for the provisioning across Ethernet and SONET regions, and the ESnet utilizes MPLS for mapping of Ethernet edge ports into layer 3 label switched paths. While the details of these internal mechanisms and associated interface to the domain controller are specific to the domain, the WS UNI and WS E-NNI will naturally frame what is needed in terms of a WS I-NNI IF. For this reason, some broad outlines of the WS I-NNI IF are also discussed in this architecture.

The following four main Web Service functions areas are identified for further discussion:

- WS UNI Services
  - User Request
- WS E-NNI Services
  - Topology Exchange
  - Resource Scheduling
  - Signaling

One of the benefits, and driving motivations, to follow a Web Service framework is the built in security mechanisms available. A robust and flexible set of capabilities to incorporate encryption, message integrity, authentication, and authorization is essential to hope for any degree of multi-domain adoption. This includes an ability for a client to authenticate a server (i.e. Domain Controller), as well as for the Domain Controller to be able to authenticate AND apply policy based processing to user requests. Basic SSL over HTTP will be utilized for the former. However, the latter is a complex space and the Web Service Security (WSS) specification by OASIS is being utilized as the basis to solve these issues. The WSS framework provides a method to

include message signatures, authentication tokens, and authorization credentials in the header of a SOAP message. This includes the use of X.509 certificates and Security Assertion Markup Language (SAML) assertions as tokens in the SOAP header to provide the required mechanisms to allow policy based user request processing and provisioning. SAML assertions contain information that can be used to make user access and provisioning decisions. Due to space limitations, the details of the certificate formats, user attribute definitions, and associated SAML assertions are not reviewed here. In summary, the benefits offered by Web Services include:

- standardized mechanisms for user authentication and policy management
- flexible features for interfacing with a diverse set of I-NNI mechanisms
- allowing focus on several issues that current control plane work has not addressed in a robust manner: scalability, stability, security, flexible application of policy, AAA, and scheduling

While there is some overlap and parallels with the ASON, OIF, IETF-CCAMP UNI/I-NNI/E-NNI work there are also some key distinctions. The standards bodies:

- have not converged on Inter-AS interdomain E-NNI routing or signaling protocols
- completed very little work on application of an Authentication, Authorization, Accounting (AAA) model for the control plane
- completed very little work on scheduling of provisioned services
- are not addressing scalability and security to the degree required for the R&E community

In addition, use of a Web Service technique between some domains will still allow for other peering domains to utilize non web service E-NNI (i.e. GMPLS based) as desired. This heterogeneous style control plane interconnect is anticipated to be the result of regional networks preferring to use a GMPLS based E-NNI when connecting with a trusted wide area network while the interdomain connections between peering international wide area networks is likely to be based on Web Services. The previously identified WS UNI and WS E-NNI functions are now described in more detail below.

#### *User Request*

The User Request services provides a mechanism for a client to provide the details of a desired multi-domain path instantiation. The intent is for user information to be very simple. The details, and complexities, of instantiating a multi-domain path is the responsibility of the network control plane. In this manner, the user views this as a "network service" for which it only needs to inform its local domain controller of its request. A createReservation WSDL operation is defined which contains a message that includes the following information:

- X.509 User Certificate
- source IP(circuit ingress node)
- destination IP (circuit egress node)
- bandwidth Requested
- start time
- duration

The createReservationResponse returns a set of information to the user for utilization in a subsequent Signaling phase. This includes the following:

- reservation id
- per signaling instance token
- unconfirmed loose hop path

There are also similar createPath and createPathResponse WSDL operations identified for the signaling of a path after a reservation is confirmed. The ability to provide the information in the above response messages relies on the network to network WS E-NNI functions. These are described in the following sections.

#### *Topology Exchange*

A starting point for multi-domain provisioning is for each network to provide a summarized or abstracted view of their topology and make it available on a global basis. The purpose of this Topology Exchange is for each domain to provide sufficient information such that a service initiating domain can determine possible multi-domain paths for a requested circuit. The output of this process is referred to as an "unconfirmed loose hop" path and is the basis for initiation of the subsequent Resource Scheduling phase.

The default abstracted topology is a network view which includes all the inter-domain links and associated border network elements. The interconnectivity between domain border network elements is then represented via a set of virtual links. This allows individual networks to hide their real topology as well as control what level of resources are offered for dynamic provisioning. It is envisioned that in the default case little or no dynamic information will be included in the topology exchange. The motivation for limiting the exchange of dynamic data is due to scalability concerns. The type of dynamic information we refer to here is instantaneous bandwidth availability on a link, or real-time VLAN tag availability for Ethernet based services. The topology will contain "static" information such as total capacity of the link, type of link (PSC, L2SC, TDM, LSC, FSC), range of VLAN tags being offered for dynamic provisioning, and other items which do not change on frequent basis. The result of this approach is that the Resource Scheduling phase has the burden of identifying and reserving exact resources for specific service instantiations. This is similar in concept to the work in the IETF Path Computation Element (PCE) Working Group. There may be some situations, where dynamic information exchange is desired and this is a topic for further evaluation and test.

The following WSDL operations are identified for the Topology phase:

- `getNetworkTopology` - results in an XML file which defines the (abstract) network topology in a network graph (nodes and links) format
- `getNetworkRouterList` - returns the full or partial set of router IDs in the network
- `getInterNetworkLinkList` - returns the inter-network link descriptions
- `getAdjacencyNetworkList` - returns a list of links that point to the Topology Description services in the adjacent domains or networks. The operation can be used to recursively explore the entire inter-network topology
- `getNetworkTopologyDelta` - returns only the updated portion of a network topology. The `getNetworkTopology` operation returns a full description of the topology. This topology will likely be an abstracted topology based on local domain policy and configuration.

#### *Resource Scheduling*

The purpose of Resource Scheduling phase is to enable a multi-domain, multi-stage path computation process which results in a "confirmed loose hop explicit route object". A "confirmed loose hop explicit route object" is defined as a loose hop object which has been reviewed and approved by all the domains in the path. This will involve the generation of a per "confirmed loose hop explicit route object" instance security token which can be utilized later in the signaling phase. This process is initiated by using an "unconfirmed loose hop" which can be generated locally via the information available as a result of the Topology Exchange services.

The following WSDL operations are identified for the Resource Scheduling phase:

- `createReservation` - includes an "unconfirmed loose hop" object as the basis for a multi-domain resource scheduling process. The "unconfirmed" means that resources have not been held across the various domains. It also means that resource parameters like specific VLAN tags have not been identified at this point. This operation in general relies on the above `getNetworkTopology` web service in order to run a path computation algorithm across that network graph.
- `createReservationResponse` - returns a "confirmed loose hop" and a per reservation instance security token. Both of these are utilized at signaling time to associate with the policy decisions made during the resource scheduling phase with actual network provisioning.
- `getConfirmedPath` - returns a "confirmed" loose hop explicit route object in XML format. The "confirmed" means that resources have been held across the various domains. It also means that resource parameters like

specific VLAN tags have been identified and are being held for future provisioning.

- `getConfirmedPathStatus` - check on status of a previously `getConfirmedPath` result
- `revokeConfirmedPath` - revokes a previously requested `getConfirmedPath` computation request

#### *Signaling*

The purpose of Signaling phase is to instantiate a circuit based on the results of the Resource Scheduling phase. The Signaling phase will utilize the "confirmed loose hop" (which includes specific VLAN information for ethernet circuits) produced in the Resource Scheduling phase.

The following WSDL operations are identified for the Signaling phase:

- `createPath` - this utilizes the "confirmed loose hop explicit route object" to instantiate the multi-domain path
- `createPathResponse` - provides success or failure status of path set up attempt
- `getPathStatus` - returns a status of a specifically indicated path
- `tearDownPath` - requests tear down of a specifically indicated path
- `refreshPath` - periodic refresh information for a path already instantiated, similar in function to in a way analogous to RSVP-TE operations.

A flow diagram is shown in Figure 2 which describes a sequence (or work flow) using these four Web Service to instantiate a multi-domain circuit. In general, the WS E-NNI functions listed above are intended to occur in sequential manner. That is, Topology Exchange is followed by Resource Scheduling which is followed by Signaling. These are intended to be standard Web Service type processes, i.e., each domain offer a set of available Web Services (via WSDL), and then applications (or work flows) will utilize these Web Services. It is envisioned that the work flow shown in Figure 2 would be accomplished via a set of "recursive Web Service" actions. That is, one Web Service would invoke another Web Service (often from another domain), which might invoke a third Web Service, and continuing as necessary to completion. Every Web Service would return a success or failure indication (with detailed status information) such that the state of a service can be known in a deterministic fashion.

While the four areas are intended to work together to provision a multi-domain circuit, they can in fact be used independently. For instance, if the information obtained via Topology Exchange was available via other means, then one could proceed directly to Resource Scheduling without going thru a Topology Exchange phase. In a similar manner, if the information obtained via Resource Scheduling was available thru other means then, one could proceed directly to Signaling without going thru the prior phases.

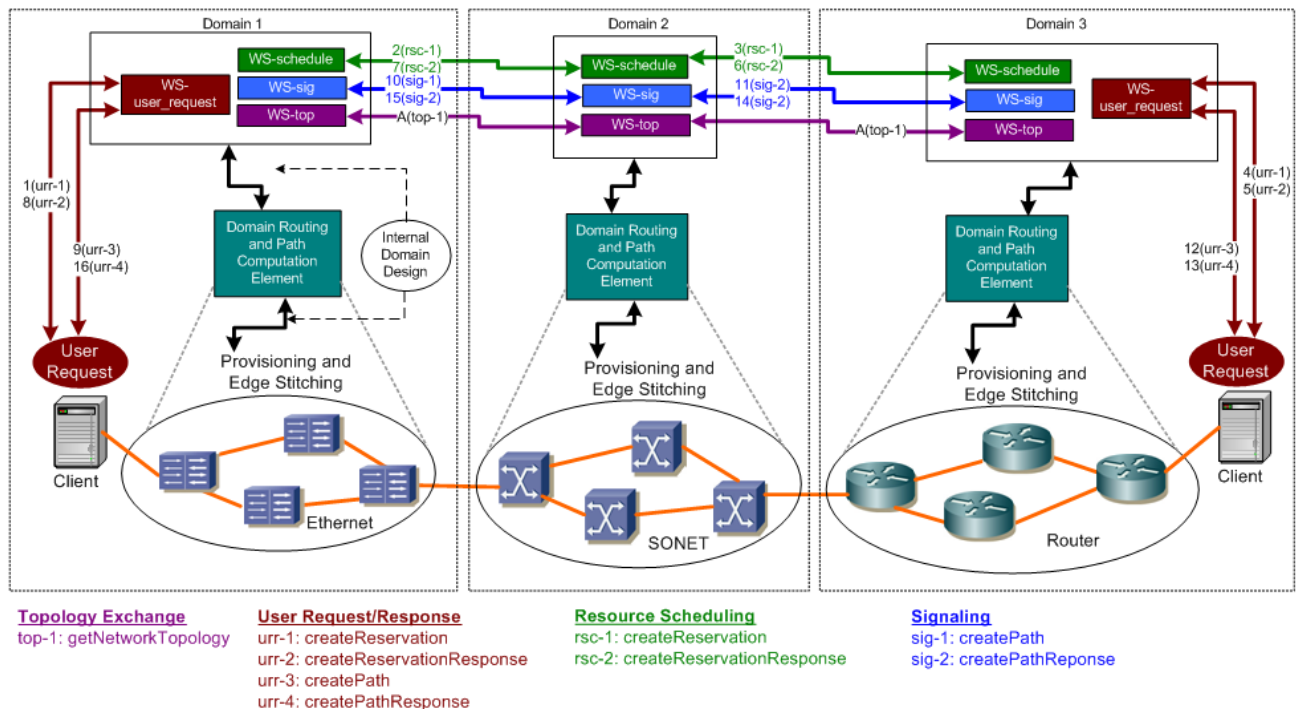


Figure 2 Multi-Domain Web Service Based Provisioning Flow

An alternative technique for the Signaling phase has also been defined which allows for "token based RSVP" to be utilized for Signaling, instead of web services. In this mode Web Services are still utilized for the Topology Exchange and Resource Scheduling phase. Additional details on this mode will be presented in future papers.

### III. DEPLOYMENT SUMMARY AND STATUS

We have begun initial implementation and testing of this control plane architecture on ESnet, USN, and Internet2 Network. This work is based on augmentation and adaptation of the OSCARS[9] and DRAGON[5] software to include these new multi-domain web service features. As shown in Figure 1, this represents a heterogeneous dataplane and control plane environment on which to evaluate this service plane architecture. We have completed initial testing of web service based signaling/path setup between ESnet and Internet2 HOPI. This has verified an ability to provision interoperable end-to-end services spanning both a MPLS/PSC and GMPLS/L2SC region.

### IV. CONCLUSION

In this paper we propose a "service plane" approach to addressing dynamic and deterministic service provisioning in multi-service, multi-layer, multi-domain hybrid network environments. The results of this work indicate that Web Service, GMPLS, and Management style based provisioning systems can be made to interoperate for the efficient provisioning of multi-layer, multi-domain hybrid network

resources. Additional architecture, design, and implementation work will continue to further evaluate this approach.

### Acknowledgements

This research is sponsored by the High Performance Networking Program of the Office of Science, U.S. Department of Energy.

### REFERENCES

- [1] DOE Energy Sciences Net (ESNet), <http://www.es.net/>.
- [2] DOE UltraScience Network (USN), <http://www.csm.ornl.gov/ultranet>
- [3] CHEETAH: Circuit-switched high-speed end-to-end transport architecture testbed. *IEEE Communications Magazine*, 2005.
- [4] Internet2 Network, <http://www.internet2.edu/network>.
- [5] DRAGON (Dynamic Resource Allocation via GMPLS Optical Networks), <http://dragon.east.isi.edu>
- [6] The IETF Common Control and Measurement Plane (ccamp), <http://www.ietf.org/html.charters/ccamp-charter.html>
- [7] The Optical Internetworking Forum(OIF), <http://www.oiforum.com>
- [8] ITU-T Recommendation G.8080/Y.1304 (2003), Architecture for the Automatically Switched Optical Network (ASON)
- [9] DOE ESNet, "OSCARS: On-demand Secure Circuits and Advance Reservation System", <http://www.es.net/oscars/>.