Network Service Interface – gateway for future network services

Radek Krzywania
Poznan Supercomputing and Networking Center,
ul. Noskowskiego 12/14, 61-704 Poznan, Poland
e-mail: radek.krzywania@man.poznan.pl

Joan Antoni Garcia-Espin
Distributed Applications and Networks Area, i2CAT
C/Gran Capitŕ 2--4, oficina 203, Edifici Nexus 1, 08034, Barcelona, Catalonia, Spain
e-mail: joan.antoni.garcia@i2cat.net

Chin Guok
Lawrence Berkeley National Laboratory, Energy Sciences Network
1 Cyclotron Road, Mail stop 50A-3111, Berkeley, CA 94720, USA
e-mail:chin@es.net

Jeroen van der Ham
University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands
e-mail: vdham@uva.nl

Tomohiro Kudoh
AIST, Information Technology Research Institute
Central2, 1-1-1 Umezono, Tsukuba, Ibaraki, 305-8568 Japan
e-mail: t.kudoh@aist.go.jp

John MacAuley
SURFnet
Radboudkwartier 273, 3511 CK Utrecht, The Netherlands
e-mail: john.macauley@surfnet.nl

Joe Mambretti
Northwestern University
750 North Lake Shore Drive, Suite 600, Illinois, USA
e-mail: j-mambretti@northwestern.edu

Inder Monga
Lawrence Berkeley National Laboratory, Energy Sciences Network
1 Cyclotron Road, Mail stop 50A-3111, Berkeley, CA 94720, USA
e-mail: imonga@es.net

Guy Roberts
DANTE
City House, 126-130 Hills Rd, Cambridge, CB2 1PQ, England

e-mail: guy.roberts@dante.org.uk

Jerry Sobieski
NORDUnet A/S,
Kastruplundgade 22, DK-2770 Kastrup, Denmark
e-mail: jerry@nordu.net

Alexander Willner
Institute of Computer Science 4
Friedrich-Ebert-Allee 144, 53113 Bonn, Germany
e-mail: willner@cs.uni-bonn.de

Paper type
Research Paper

Abstract
Purpose
The NSI main objective is to provide a unified communication method enabling independent single domain resource management tools to collaborate at global scale providing multi-domain services in heterogeneous environments.

Approach
The NSI research was driven by specialists experienced in dynamic provisioning system and resources allocation. The protocol design was a result of long discussions and evaluations of proposed architectures and ideas.

Findings
The NSI efforts was focused on network provisioning and resulted in specification of NSI Connection Service (CS) protocol v1.0, released in 2011, and adopted by a set of independent network provisioning tools. Successful demonstrations at the end of year 2011 have proven the potential of the protocol and estimated direction of further development.

Research limitation
The NSI CS needs to meet requirements of many users and network providers in order aggregate them and to deliver a single communication language for global resources allocation. There is a need to enhance security mechanism of NSI, topology modeling and advertisement, monitoring features, and accounting, before the protocols could be used in purely operational environment.

Practical implications
The research is mainly driven by high capacity demanding users and operators of NRENs around the world, who found present inter-networks circuit creation methods inconvenient and inefficient, as those are mostly manual processes without any automation. The NSI is giving a proposal on how to introduce multi-domain dynamic services to all interested researchers.

Value
The paper describes a new concept of multi-domain resources management, where services can go beyond single domain boundaries facing current research community requirements. The proposed NSI framework is not a tool or application, but rather a language that can be learned by provisioning tools to communicate each other. This feature in correlation with open standard implies unlimited scalability.

Keywords
Network, protocol, connections, services, global connectivity

1. Problem statement

Over the last decade, global networks have begun delivering high-performance transport services directly to applications that require performance levels or capabilities unavailable in conventional, best-effort IP networks. The ability to create connections between a fixed set of ports worldwide, with specific, predictable, and often demanding performance characteristics, enables emerging global collaborations to establish well-defined and highly customized network environments to support the end users and their applications. This has been particularly true within the Research and Higher Education environment and the Grid community.

Historically, connections across these transport networks have been reserved and provisioned in a variety of ways. The most common approach is manual provisioning – typically performed by a network engineer. More recently, some networking communities have developed tools and protocols to automate the process of network resource allocation allowing users or applications to participate directly in the path creation process. These new approaches to automating transport connection provisioning are the basis for the standardization effort behind the Network Service Interface.

Automated connection-oriented transport provisioning capabilities are currently being deployed by Research and Education (R&E) providers as well as by commercial providers, and could eventually be implemented in home/retail networks as deployment progresses. , Despite the fact that these automated provisioning systems are being developed independently by different communities, all share similar concepts and have common architecture components. They have developed software-based control agents to regulate access to the network hardware, in order to schedule and reserve resources, to trigger or control timely provisioning of the network resources, and to monitor and release resources. These controllers are deployed in two different contexts. One context is application-centric, where a network provides a resource to an application or middleware. The other context is network-centric; where network resources are collaboratively shared among networks to expand or improve network performance or reach. In the former context, a user or an application agent is requesting the service from a network provider. In the latter context, one network is interacting with one or more other networks to manage these resources and deliver a comprehensive and well-integrated service portfolio to the user community.

A new type of a network wide area architecture often referred to as “customer-controlled” or “customer-managed networks” [1] is becoming increasingly common among large enterprise networks, university research networks, and government departments. Customer-controlled and -managed networks are radically different from the traditional networks in that the institution not only manipulates its own local/campus area network, but also its own wide area optical network, assuming responsibility for direct peering and interconnection with other networks. As a consequence, traditional management and hierarchical backbone network technologies, which are premised on central provisioning for network paths to customers, are largely unsuitable for customer management of their own network.

1.1. Related Work

Applications typically need to allocate and reserve multiple types of resources, such as computation, data, instrumentation, and networks. In 1999, Czajkowski [2] defined the co-allocation problem for computational Grids. In the same year, based on the techniques and concepts of the Globus Resource Management Architecture (GRMA) [3], a Distributed Resource Management Architecture (GARA) [4, 1] that used a Globus Resource Allocation Manager (GRAM) job scheduler to co-allocate the network with other resources in advance was proposed. By then, the mechanisms used for network provisioning were IntServ [5] and DiffServ [6].
While GARA is popular among the Grid community as a general-purpose platform allowing reservations of numerous resources, it is not specialized for networks. Its API and Resource Specification Language (RSL) do not take network specific attributes into account. Therefore, the Network Resource Scheduling Entity (NRSE) [7] was introduced in the Grid Resource Scheduling (GRS) project in 2002. Still IntServ and DiffServ mechanisms were used to deliver guaranteed throughput over packet-based networks.
Unfortunately, bulk data transfer oriented Grid computing often requires guaranteed minimum bandwidth and minimized packet loss, which are not easily achievable in packet switching networks. Moving terabytes or petabytes of data among multiple sites requires dedicated (preferably optical)

networks, e.g. based on wavelength switching, that can provide guaranteed bandwidth and performance in terms of low bit error rates.
In 2005 the Exploitation of Switched Light paths for e-Science Applications (ESLEA) project had started. It demonstrated the usefulness of circuit-switches networks for different application areas. The ESLEA Control Plane Software (CPS) was implemented as a modification of the aforementioned NRSE and was integrated into the EGEE Bandwidth Allocation and Reservation (BAR) architecture. At the same time the DARPA DWDM-RAM project addressed similar issues and a Network Resource Scheduling (NRS) service was developed to enable the efficient use of optical networks as a primary Grid resource.
In the Research and Education environment, several solutions for intra- and inter-domain path provisioning and scheduling exist. Examples for the former are ARGIA [8] (former UCLP – User Controlled Light path Provisioning); Nortel's Dynamic Resource Allocation Controller (DRAC) [9], which now is being opened to the community through openDRAC [10]; the MPLS-based Allocation and Reservation of Grid-enabled Optical Networks (ARGON) [11] system that was developed within the German research project VIOLA. With regard to the latter, examples to be taken into account are: the Automated Bandwidth Allocation across Heterogeneous Networks (AutoBAHN) system [12], originating from the GÉANT2 project; the inter-domain control plane based on OSCARS was developed as an achievement of the DANTE-Internet2-CANARIE-ESnet collaboration (DICE), where an Inter-Domain controllers (IDCs) communicate in a decentralized way to provision end-to-end multi-domain network paths; the G-lambda project, on its turn, developed an interface between Grid resource management systems and network resource management systems that also support advance reservations; the GMPLS based EnLIGHTened Computing project focuses on dynamic optical light-paths between supercomputing sites that are created and torn down in advance or on demand based upon application needs; the Dynamic Resource Allocation in GMPLS Optical Networks (DRAGON) project that aims at dynamically provisioning packet and circuit switched network resources in response to user requests for high-performance e-Science applications; and last but not least, the solutions from the EU FP6 Phosphorus project [13]: Grid-enabled GMPLS (G^2MPLS) Network Control plane (as an enhancement of the ASON/GMPLS Control Plane architecture that implements the concept of Grid Network Services (GNS)) and the Harmony Network Service Plane [14], which provides a full-fledged network services framework with advance reservation capabilities and user/Grid middleware interfaces for multi-domain network provisioning using ARGIA, ARGON, DRAC and GMPLS/G^2MPLS.

The role of large-scale science is fundamental to the study of the most complex, subtle, and elusive natural phenomena. Such studies are completely dependent on world-wide collaborations of scientists and scientific workflows that coordinate large volume of distributed data and geographically dispersed and diverse resources such as instruments, storage assets, compute nodes, visualization appliances, and networks. Below is a discussion of several scientific application workflows that require a certain level of network guarantees and predictability to function optimally.

1.2. Movement of large data sets with deadline scheduling requirements

The Large Hadron Collider (LHC) at CERN is the largest, most expensive, and most powerful high energy particle accelerator in the world. The LHC was designed to recreate the conditions similar to those just after the Big Bang in order to further our understanding of the Standard Model of particle physics by answering some questions, such as the existence of the Higgs boson, composition of dark matter and dark energy, matter/antimatter biases, and the reality of extra dimensions of space. The LHC instrument (Tier 0) supports 7 detector experiments and produces on average 15 PBytes of data a year. Because of the relatively large amounts of information that are constantly being generated by the experiments, data must be moved in a timely manner to remote Tier 1 storage sites located around the world to avoid exhaustion of the Tier 0 local resources. In addition, the LHC research community must provide for data flows among Tier 1, Tier 2, and Tier 3 sites around the world. *The use of schedulable guaranteed bandwidth circuits facilitates predictable data transfers and supports deadline scheduling.*

1.3. Co-relation of data sets generated by distributed instruments

Very Long Baseline Interferometry (VLBI) is a research method for obtaining highly detail information about the cosmos. The method uses multiple radio telescopes that are highly distributed, with each focused on the same area of the cosmos. The radio telescopes sample large volumes of information, which is sent simultaneously in real time to computational facilities for data analysis, correlation, and image construction. These images of the

observational areas and objects can be much more detailed than those provided by optical telescopes. This technique also allows for real time instrumentation adjustment. *The use of guaranteed multipoint-to-single point network paths that can be provisioned ad hoc allow for extremely large streams of high quality research data to be gathered and analysed from multiple distributed sites.*

1.4. Storage and Retrieval of Data from Distributed Depots

The Earth System Grid Federation (ESGF) was designed as a data distribution portal for the Coupled Model Intercomparison Project (CMIP) as part of the World Climate Research Programme (WCRP) to support climate model diagnosis, validation, intercomparison, documentation and data access. To date, the CMIP federated archive includes simulation data from 109 experiments performed with 44 different climate models from 25 modelling centres around the world. The 41,420 data sets consist of 2,468,940 files in 1,018.62 TBytes replicated and stored in 18 data nodes internationally. *The use of guaranteed point-to-multipoint network overlays can support efficient replication of information by reliably multicasting data from one source to multiple depots.*

1.5. Time Sensitive Data Transfers as part of an Execution Workflow

The Magnetic Fusion Research D-IIID program in San Diego, California was established to understand and optimize the production of fusion energy using tokamaks, and subsequently shaped the design of the International Thermonuclear Experimental Reactor (ITER). The D-IIID experiments operate in a pulsed mode to produce plasmas of up to 10 seconds in duration every 15 to 20 minutes, generating several gigabytes of data per pulse. In order to adaptively change the parameters for the next plasma pulse, the data from the previous pulse must be transferred to a remote compute cluster for analysis and assimilated in near real-time by a geographically dispersed research team within the 15-20 minute window. *The ability to co-schedule network bandwidth along with compute and storage resources is essential to support time sensitive distributed workflows.*

1.6. Remote Control of Experiments/Instruments

Multiple advanced light instruments are been implemented to explore the fundamental properties of matter. For example, the Advance Light Source (ALS) in Berkeley, California was built to study the atomic and electronic structure of matter. The ALS, with its 39+ beam lines, produce x rays that are one billion times brighter than the sun, allowing researchers to observe structures and understand biological processes that are inscrutable to visible light. At any point in time, between 50-100 researchers in various science disciplines from around the world use the ALS to conduct experiments that last anywhere from an hour to three weeks. *The ability to schedule low latency, near-zero jitter, guaranteed bandwidth circuits is necessary to support remote control applications.*

2. Network Service Interface

As the answer for raising demand for dynamic creation of global connections, a working group termed Network Service Interface (NSI) was created within Open Grid Forum community, to unify the protocols and procedures for network provisioning. The main objective of this group was to provide a common interface for global reservations and resources negotiations, which can be easily adopted by already existing provisioning tools and enable automated inter-domain communication of peering control planes. The individuals involved in that efforts were representing knowledge and experience of NRENs around the world, provisioning tools developers, as well as the end user communities, which gives the momentum to make NSI a global standard for resources negotiations in heterogeneous environments at global scale.

2.1. NSI architecture

In the proposed architecture of the NSI [15], a dedicated Network Service Agent (NSA) manages each provider's network. These agents interact to realize the delivery of a Network Service supported by the network infrastructure. The NSI is the service interface between NSAs and is used to request and provide the network services. An NSA can take on the role of a requester, a provider, or both. As a requester, the NSA requests network resources and as a provider it delivers network resources to create a service. The NSA acts as both when it is a requester over one interface while acting as a provider at a different interface.

The Network Services Framework allows multiple services to be delivered across a chain of multiple participating networks. Thus, multiple NSAs can form a recursive framework of requesters and providers, only restricted by the trust relationships pre-established between the various network administrative domains. NSI requests can be propagated through this framework of NSAs using a tree or chain workflow.

The NSI protocol describes an exchange of messages between the requester and provider. Each NSI Message includes a set of attributes that provides the specific details of the service being requested. The first specification of NSI protocol is the Connection Service (CS), which allows requestors to setup a point-to-point circuit across multiple network domains. In order to build a connection service a circuit end point is designated with a Service Termination Point (STP) identifier. STPs are conceptual entities, and by combining the concepts of NSAs and STPs, a mapping between a "service topology" and an abstract representation of multi-layer physical topology can be realized. When two STPs in adjacent networks with matching capabilities are paired, the resulting pairing forms a Service Demarcation Point (SDP). SDPs are used by NSA path-finding algorithms that help build segments across the recursive tree that connect with each other to satisfy the connection service request.

Using the NSI Connection Service, the reservation request for an end-to-end connection is shown in Figure 2.1. The User Agent may request a connection at the Provider Agent, who acts as an Aggregator taking care of global resources negotiation phase. The Requestor agents delegates resources management to local Provider Agents responsible for controlling particular domain along reservation path (Networks A, B, and D). Those agents uses Network Resources Mangers (NRMs) within a domain to implement connection segments in their network domains. More details on the process are described in chapters 2.2 Messaging primitives, and 2.3 Messaging framework The NSI-CS state machine supports the implementation of an auto-start (automated and independent provisioning of the circuit due to a time-based trigger) and manual-start (provisioning of the circuit that is triggered by an explicit message by the requestor). In either case, an explicit provisioning message is required. The behaviour of auto-start or manual-start is dependent correspondingly on whether the provisioning message is received before or after the start time of the reservation. The amount of information exchanged between NSAs is limited to minimum in order to provide the simplest possible solution, easy to implement in deploy in production. Since the protocol was a proof of concept, a work on further extensions is in progress.

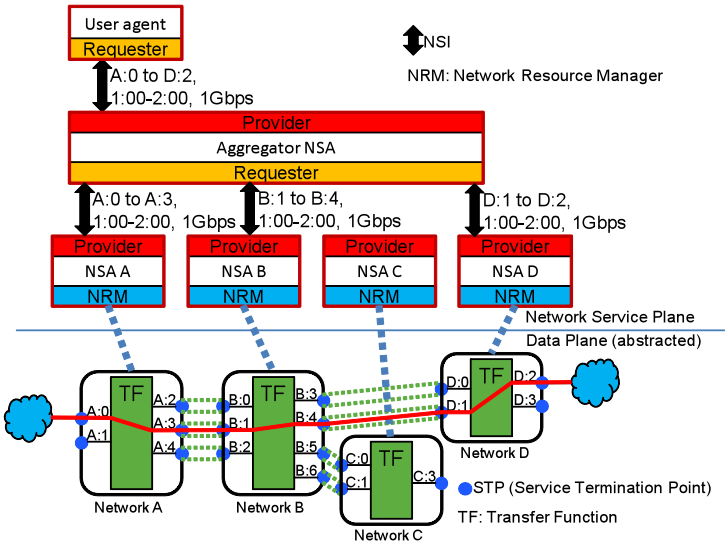

Figure 2.1 The NSI-CS architecture and reservation overview

2.2. Messaging primitives

The NSI Connection Service (CS) protocol is a message based request/response protocol that operates between a Requester NSA (RA) and a Provider NSA (PA). The protocol defines the following set of five primitives providing the control necessary to manage connections within the network:

reserve

The RA requests the PA to reserve network resources for a connection between two STP's constrained by the provided service parameters.

provision

The RA requests the PA to provision a reservation associated with a previous reservation message. If the reservation start time has passed then this will initiate an activation of the data plane resources within the networking equipment.

release

The RA requests the PA to de-provision resources from the network without removing the reservation. All resources part of the reservation will continue to be reserved.

terminate

The RA requests the PA de-provision the provisioned resources from the network and terminate the reservation. All resources part of the reservation will be freed for use in other reservations.

query

A mechanism for either of the RA or PA to query the other NSA for a set of connection service instances between the RA-PA pair. The requester can also ask for a recursive query to be performed that will return all connection information from all NSA involved in the reservation. This message can be used as a reservation status polling mechanism.

Each of these five NSI CS primitives is implemented using three protocol messages supporting the request/response interaction:

Request

The RA sends a request to the PA containing the desired operation, for example *reserveRequest* is issued to request a reservation from a PA. Only the *queryRequest* message can be issued by both the RA (RA to PA) and by the PA (PA to RA).

Confirm

The PA sends this positive operation response message (such as *reserveConfirm*) to the RA that issued the original request message (*reserveRequest*) if the operation requested is successful (Figure 2.2).

Failed

The PA sends this negative operation response message (such as *reserveFailed*) to the RA that issued the original request message (*reserveRequest*) (Figure 2.3).

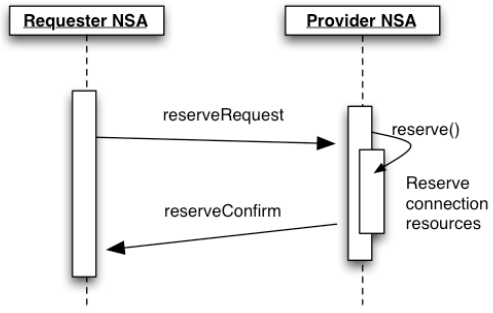

Figure 2.2 – Request/confirm exchange.

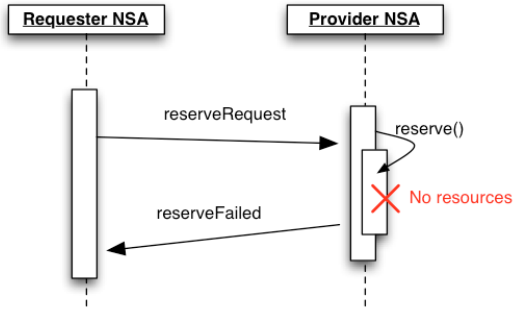

Figure 2.3 – Request/failed exchange.

It is import to note that a significant amount of time can occur between an RA issuing a request message to the PA, and the PA returning a corresponding confirm or failed back to the RA. This is the result of the behavioural definition of the operation primitives, and the duration needed to perform the operations within a distributed NSA environment. This had a direct impact on the underlying transport implementation as described in Chapter 2.3 Messaging framework.

In addition to these five protocol operations, the NSI CS utilizes a notification model to send autonomous events from the PA to RA as they are generated from children NSA. At the moment, only the following notification message is supported:

forcedEnd

This is reported by the PA to the RA to notify that the PA has forced (administratively) a termination of the reservation. The forcedEnd is issued as a Request message without a paired Confirm message (Figure 2.4).

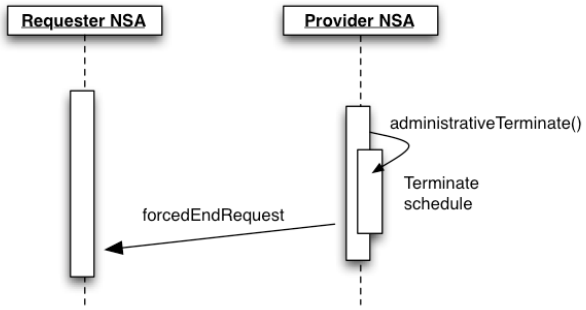

Figure 2.4 Notification exchange

These five operation primitives, combined with the autonomous notifications, are the basic building blocks for the NSI CS protocol.

2.3. Messaging framework

To help expand the abstract NSI CS primitives into a full protocol specification, the NSI-WG utilized the standard XML schema and Web Services Description Language for message definition. This permitted the team to specify data types associated with the content of the messaging primitives, as well as specify the message structure in a concrete protocol definition. In addition, by utilizing XML and WSDL for the specification, standard tools could be utilized for design, and working implementation prototypes could be rapidly developed. This was extremely valuable to help prove out the protocol and get continuous feedback for improvement.

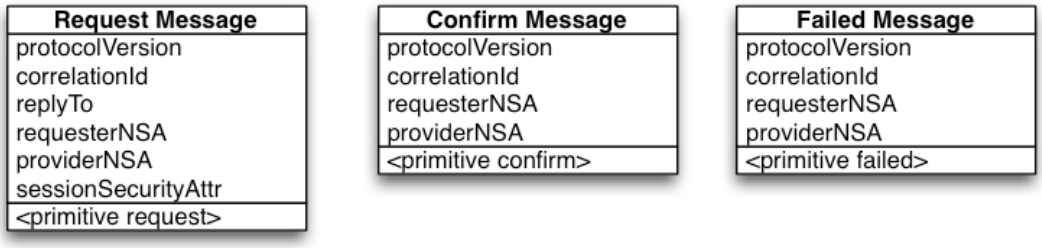

Figure 2.5 Generalized message envelope.

Figure 2.5 provides an abstract representation of the NSI framework's request, confirm, and failed message envelopes. Each message contains a common set of header attributes, followed by data associated with the specific operation primitive.

protocolVersion

This abstract attribute is implemented using the XML schema namespace URI for the specific message primitive. This value is updated for each new version of the protocol or change in the XML schema representation.

correlationId

The *correlationId* attribute is of type UUID and is a unique value identifying the request. The *correlationId* may be used to associate a response (confirmed or failed) with the instance of the request that triggered the response.

replyTo

The replyTo attribute is of type any *URI*, and is used as a protocol endpoint address for the destination RA of any response messages associated with the request. The protocol endpoint to deliver the initial request message to the PA is a published address. Only the response endpoint address is passed in the message.

requesterNSA

The *requesterNSA* attribute identifies the name of the source RA of the original request message.

providerNSA

The *providerNSA* attribute identifies the name of the destination PA targeted by the original request message.

sessionSecurityAttr

The *sessionSecurityAttr* is the security attribute associated with the NSI connection services session. This attribute is an opaque element that contains information that may be used to authenticate the end user who made the request and authorize the associated operation. NSA-to-NSA security is not implemented using this mechanism, but instead use security provided by the transport mechanism (in our case HTTPS and SOAP).

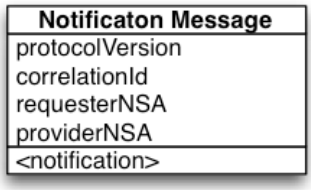

Figure 2.6 – Generalize notification envelope.

Figure 2.6 provides an abstract representation of the NSI framework's notification message envelope. Each header attribute is used similar to the request header, however, there is no reply expected for this message, and therefore, no *replyTo* field is present. In addition, there are no *sessionSecurityAttr* since these notifications are not user initiated as with the request messages.

As described in the previous section, a significant amount of time can occur between an RA issuing a request message to the PA, and the PA returning a corresponding confirm or failed back to the RA. To implement this type message interaction behaviour using web services, an asynchronous messaging model was introduced requiring both the RA and PA to implement SOAP endpoints for receiving message primitives. Figure 2.7 illustrates this model.

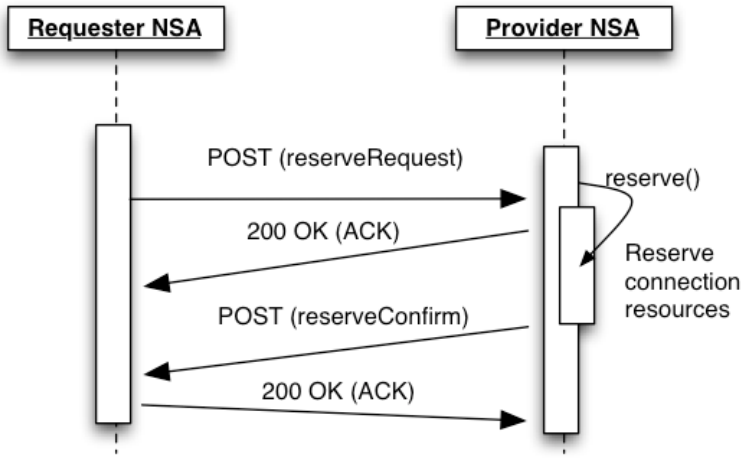

Figure 2.7 – Transport messaging interactions.

The original *reserveRequest* is sent from the RA to the PA's SOAP endpoint using a standard SOAP HTTP POST operation. The PA immediately responds with an HTTP *200 OK* response and a simple acknowledgement identifying the *correlationId* of the *reserveRequest* indicating it has accepted the message for processing. At some later time when the network resources have been successfully reserved, the PA will send a *reserveConfirm* to the RA's SOAP endpoint (identified in the request replyTo field) using a standard SOAP HTTP POST operation. The RA immediately responds with an HTTP *200 OK* response and simple acknowledgement identifying the *correlationId* of the *reserveConfirm*.

If for some reason the PA cannot accept the reserveRequest for processing, it must reply to the SOAP POST operation with an HTTP *500 Internal Server Error* containing the NSI *ServiceException* message describing the reason for the error. It should be noted that even if a 200 OK acknowledgment is returned for the *reserveRequest,* the PA still may fail the reservation request and return a *reserveFailed* message.

2.4. Reservation state diagram and example

Figure 2.8 shows a simplified v1.0SC state transition diagram of an NSA. A state machine (SM) is generated for each connection reservation. This chapter shows an example successful case only, and does not show termination of a connection. There are three kinds of events which will cause a state transition, which are RA to PA messages, PA to RA messages and timer events (startTime and endTime). When new reservation is requested a state machine is created with the Initial state. The Figure 2.8 shows a successful reservation states transition at the NSA originating the reservation request. The state transitions will occur according to the following schema:

1. A reserveRequest sent from an RA to a PA, transits from the Initial state to the Reserving state. The PA attempts to make a reservation according to parameters specified in the reserveRequest message in this state.
2. When a reservation is successfully made, a reserveConfirm is sent from a PA to an RA. The SM transits to the Reserved state.
3. Depending on the events order:
    a. If a provisionRequest is sent/received before the startTime, which is designated by a reserveRequest message, the SM transits to the Auto-Provision state.
    b. At the startTime, an activation process of a connection is started, and the SM transits to the Provisioning state.
4. Depending on state after step 3:
    a. If the startTime event occurs before a provisionRequest message is sent, the SM transits from the Reserved to the Scheduled state.
    b. If a provisionRequest is sent/received when a SM is in the Scheduled state, an activation process of a connection is started, and the SM transits to the Provisioning state.
5. When a connection is activated, a provisionConfirm message is sent from a PA to an RA, and the SM transits to the Provisioned state.
6. If a releaseRequest message is sent/received when the SM is in the Provisioned state, a de-activation process of a connection is started, and the SM transits to the Releasing state.
7. When a connection is de-activated, a releaseConfirm message is sent from a PA to an RA, and the SM transits to the Scheduled state. After this transition, the SM can transit to the Provisioning state again by a provisionConfirm message.

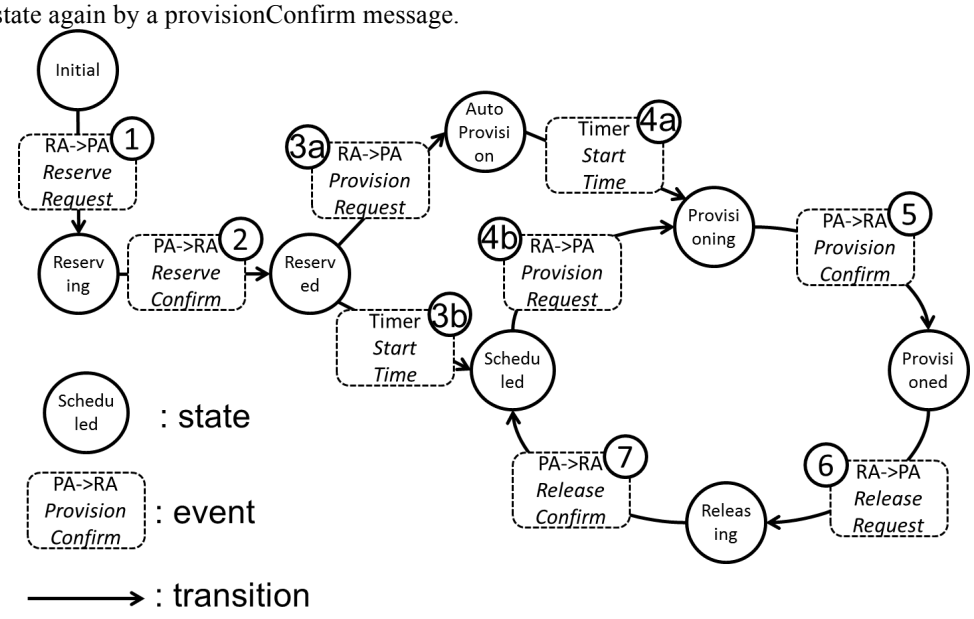

Figure 2.8 NSI-CS v1.0SC state machine diagram

The SM can transit either (2)->(3a)->(4a)->(5) or (2)->(3b)->(4b)->5 path, depending on which of a provisionRequest message or a startTime event comes first. This allows skew of message propagations.

The current state machine allowed releasing the v1.0 of NSI-CS and performing a successful demonstrations as a proof of concept for global dynamic provisioning in heterogeneous environment. A new state machine for the v2.0 NSI-CS protocol is now under discussion, which will separate message delivery confirmation in the control plane and data plane activation/de-activation notifications. This will result in simpler state machine diagram and processing, which has direct transition into implementation efforts needed for deployment.

3. Demonstrations and lessons learned

At three recent events, the GLIF meeting in Rio de Janeiro (Sep 2011), Future Internet Week in Poznan (Oct 2011), and SuperComputing11 in Seattle (Nov 2011), many parties collaborated to create the first interoperable automated inter-domain circuit provisioning demonstration, based on NSI. The demonstration showed that six different implementations could successfully communicate to exchange path reservations, queries and requests.

The implementations taking part in the demonstration are:

- AutoBAHN
- OpenDRAC
- OpenNSA
- G-Lambda-A
- G-Lambda-K
- DynamicKL

The current testbed interconnecting topology is shown in Figure 3.1. The network testbed has four VLANs permanently available for testing, and a scheduler is running continuously on one of these VLANs. The scheduler randomly selects a set of endpoint pairs from a set, sends off a bandwidth reservation for three minutes, every four minutes. This not only shows that inter-domain dynamic lightpath reservations are now possible, but also in timescale orders of magnitude smaller than ever before. Just a few years ago an inter-domain lightpath reservation took weeks to implement, instead of minutes.

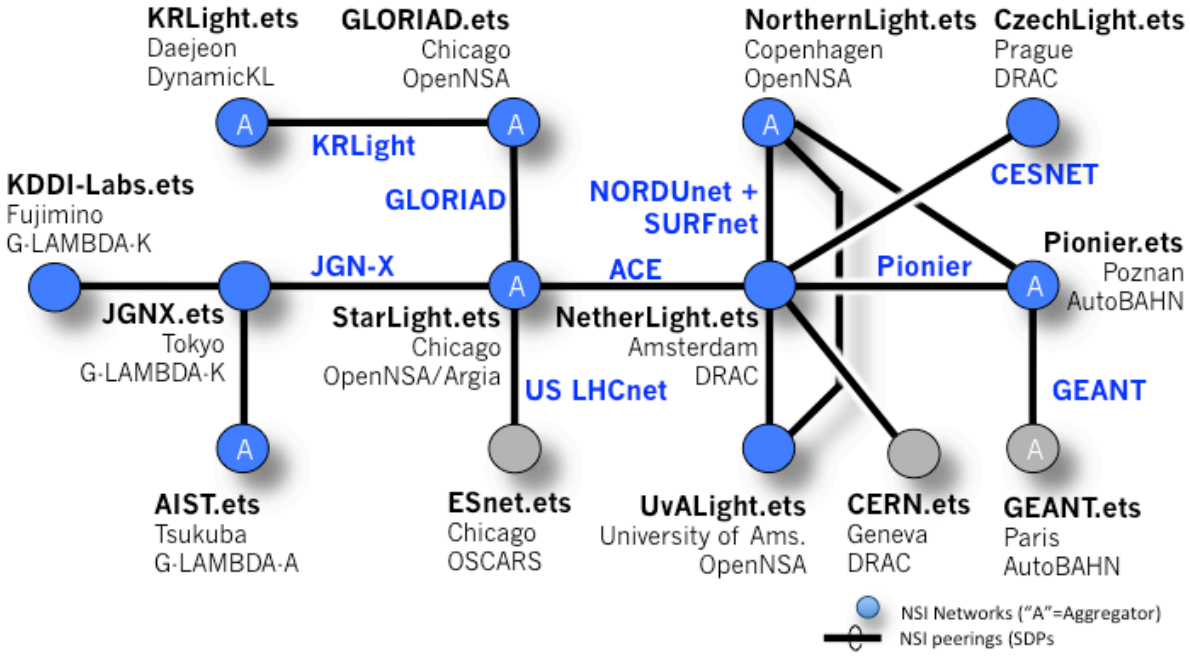

Figure 3.1 The topology of the Automated GOLE testbed in April 2012

In the demonstrations at GLIF, FIW, and SC11 we have used a static centrally managed global topology description for a very small and very simple network. This description was expressed using a simple OWL [16] ontology. This ontology allowed domains to describe themselves, and express their edge ports, (the service destinations inside each network). The domain description also contained a description of the Network Service Agent responsible for that network domain, including its address, and perhaps most importantly – the

adjacencies that existed between the networks. This information was minimally sufficient, but we learned – even from this very small pilot project – that a more powerful and automated topology model is critically important.
Over ten different networks provided resources for this Automated GOLE testbed. The topology for this testbed was provided before the demonstration and remained static throughout.

3.1. Topology Distribution

The central, static topology description for the fulfilled the needs of the demonstration, however in a practical application the topology will need to be distributed in a different manner. Domains must be able to define and advertise their own topology, which may be somewhat different than their actual physical infrastructure. The difference between the actual physical infrastructure inside a network and the publicly advertised topology is important for scaling and summarizing purposes, and often perceived to be a security or privacy issue. The global scope and complexity dictates a distributed approach to topology discovery and exchange – which introduces concerns about coherency and convergence of topology information. Further, the form in which the topology information is passed around - the ontology and representation – is critical for common exchange and shared interpretation of the information contained in the topology.

3.2. Security and Trust

Creating a globally available network for circuit provisioning also raises security considerations. This has been a key requirement of NSI from day one and has been designed into the NSI service model and the CS protocol. Each and every service request is authorized at each network boundary, and communication between each network service agent is authenticated and similarly authorized. However, while the mechanisms are in place to pass security credentials among cooperating agents, the specifics of those credentials – how they apply to the service requested is more complex. This is the "security profile". The roles of various entities requesting or providing services, the value or class of the information that may be exchanged in a service request or the function performed itself, and the local policies of each network must be considered and some minimal set of security profile agreed to in order for efficient inter-domain authorization to work reliably.
These security concerns apply equally to topology exchange. It should not be possible, for instance, for attackers to disrupt operations by injecting malicious information into the system, while at the same time allowing legitimate agents to provide appropriate A comprehensive topology architecture will carry a wide array of information besides simple data plane connectivity...it may contain peering relations at the service plane, it may carry policy descriptions, it may contain varying amounts of state information. So the topology exchange process must be provably secure so that participating agents are authenticated, and the actual content of the expressed topology should be verifiable and "valid" in context of other known topology.
Topology is a key to pathfinding, and so being able to rely on the veracity and timeliness of the topology information – trust- is critical to a secure and reliable global inter-operability. These are just a few of the topics that must be considered for a comprehensive Distributed Topology Exchange architecture.

4. Summary

The NSI framework is meeting user expectations for high bandwidth requirements delivered in easy and dynamic way. Users need the service to be provided at specific time in future, in order to assure their experiments will have sufficient data transfer background, or need fast and immediate connection with distant resources for file transfer or streaming. Current global network model requires them to contact multiple providers with phone or emails and attempt to orchestrate the connection segments through several networks. Experience shows that the process is inefficient and takes much time, which is wasted for slow human-to-human communication. Setting up a global transatlantic connection may take days, weeks or in some cases even months before users will be able to use assigned resources. This model is not applicable to current research needs, especially when terabytes, or even petabytes, are needed to be sent immediately from one point to another. The NSI provides an architecture that allows for tools enhancing the communication between networks, replacing or limiting a need for a human interaction. The recent NSI demonstrations showed, that a transatlantic connection from a server in pan-European GÉANT network to AIST in Japan can be set up in about couple of minutes, ready to use by end users. This of course requires some pre-configuration, policy and security agreements, and preparation of local infrastructures, but as the result, the network provisioning model will change from static and heavy into easily accessible by end users. The interest around the NSI framework is continuously growing, involving both scientist communities and network providers, which expresses the demand for such solution. The NSI is at the very beginning of its development, and there are still some unresolved issues, like topology

distribution, security models, monitoring, and accounting. Nevertheless, it already has the potential and support to globally unite network providers into one bandwidth provisioning service cloud in the future.

References

[1] I. Foster, N. Jennings, C. Kesselman: "Brain Meets Brawn: Why Grid and Agents Need Each Other." In: International Conference on Autonomous Agents: Proceed- ings of the Third International Joint Conference on Autonomous Agents and Mul- tiagent Systems. Volume 1. (2004) 8–15

[2] K. Czajkowski, I. Foster, C. Kesselman: "Resource Co-Allocation in Computational Grids." In: Proceedings of the Eighth IEEE International Symposium on High Performance Distributed Computing (HPDC-8). (1999) 219–228

[3] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, S. Tuecke: "A Resource Management Architecture for Metacomputing Systems.", LECTURE NOTES IN COMPUTER SCIENCE (1998) 62–82

[4] I. Foster, C. Kesselman, C. Lee, B. Lindell, K. Nahrstedt, A. Roy: "A distributed resource management architecture that supports advance reservations and co-allocation." In: Quality of Service, 1999. IWQoS'99. 1999 Seventh International Workshop on. (1999) 27–36

[5] R. Braden, D. Clark, S. Shenker, et al.: "Integrated Services in the Internet Architecture: an Overview.", IETF RFC 1633 (Proposed Standard) (1994)

[6] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss: "An Architecture for Differentiated Service.", IETF RFC 2475 (Proposed Standard) (1998)

[7] S. Bhatti, S. Sorensen, P. Clark, J. Crowcroft: "Network QoS for Grid Systems.", International Journal of High Performance Computing Applications 17 (2003) 219

[8] E. Grasa, G. Junyent, S. Figuerola, A. Lopez, M. Savoie: "UCLPv2: a network virtualization framework built on web services [web services in telecommunications, part II].", Communications Magazine, IEEE 46 (2008) 126–134

[9] F. Travostino, R. Keates, T. Lavian, I. Monga, B. Schofield: "Project DRAC: creating an applications-aware network", (2005)

[10] https://www.opendrac.org

[11] C. Barz, U. Bornhauser, P. Martini, M. Pilz, C. de Waal, A. Willner: "ARGON: Reservation in Grid-enabled Networks.", In: Proceedings of the 1. DFN-Forum on Communication Technologies. (2008)

[12] M. Büchli, M. Campanella, G. Ivánszky, R. Krzywania, B. Peeters, D. Regvart, V. Rejis, L. Serrano, A. Sevasti, K. Stamos, C. Tziouvaras, D. Wilson: "Deliverable DJ.3.3.1:GÉANT2 Bandwidth on Demand Framework and General Architecture", GÉANT, 2005

[13] S. Figuerola, N. Ciulli, M. de Leenheer, Y. Demchenko, W. Ziegler, A. Binczewski, et al.: "PHOSPHORUS: single-step on-demand services across multi-domain networks for e-science.", In: Network Architectures, Management, and Applications V. Edited by Wang, Jianli; Chang, Gee-Kung; Itaya, Yoshio; Zech, Herwig. Pro- ceedings of the SPIE. Volume 6784. (2007) 67842X

[14] A. Willner, C. Barz, J. A. García-Espín, J. Ferrer, S. Figuerola, P. Martini. "Work in progress: Harmony – Advance Reservations in Heterogeneous Multi-domain Environments", in IFIP Networking, 2009, Aachen, Germany.

[15] G. Roberts, T. Kudoh, I. Monga, J. Sobieski, J. Vollbrecht: "Network Services Framework v1.0", OGF, 2010

[16] http://www.w3.org/TR/2004/REC-owl-features-20040210/

Biographies

Radosław Krzywania received the M.Sc. degree in Computer Science – Software Engineering from the Poznan University of Technology in 2003. He is working in Poznan Supercomputing and Networking Center as a network engineer. He is responsible for research and implementation tasks of GÉANT3 for develop AutoBAHN Bandwidth on Demand system. He is responsible for building and managing of network infrastructure for Future

Internet Engineering polish national project and FP7 FEDERICA project. He is also interested in resources virtualization, efficient network utilization and management.

MSc. Joan Antoni Garcia-Espin is a research project manager at the Distributed Applications and Networks Area of the i2CAT Foundation in Barcelona, Spain. He received his MSc degree from the Technical University of Catalonia (UPC) in 2007 for a thesis on design and implementation of TE-enabled, DiffServ-aware MPLS networks providing end-to-end QoS. He is currently the work package leader for the design and implementation of the Logical Infrastructure Composition Layer in the EU-FP7 GEYSERS project. He supported the design and implementation of the bandwidth on demand tool named Harmony in the EU FP6 Phosphorus project. He was also one of the contributors to the Network Services Framework and Interface recommendation from the Open Grid Forum.

Chin Guok joined ESnet in 1997 as a network engineer, focusing primarily on network statistics. He was a core engineer in the testing and production deployment of MPLS and QoS (Scavenger Service) within ESnet. He is the technical lead of the ESnet On-Demand Secure Circuits and Advanced Reservation System (OSCARS) project, which enables end users to provision guaranteed bandwidth virtual circuits within ESnet. He also serves as a co-chair of the Open Grid Forum On-Demand Infrastructure Service Provisioning Working Group

Jeroen van der Ham received his MSc in Artificial Intelligence from Utrecht University in 2002, his MSc in System and Network Engineering in 2004 from the University of Amsterdam, and received his PhD in 2010 at the University of Amsterdam on the topic of "Semantic descriptions of complex computer networks". He is currently working as a researcher at the System and Network Engineering research group at the University of Amsterdam. His research interests are in semantic descriptions of multi-layer and multi-domain networks and (virtualised) resources, as well as associated algorithms and architectures. Jeroen is also actively involved in the OGF NML-WG as the editor of the NML Schema document.

Tomohiro Kudoh received his Ph.D. degree from Keio University in Japan in 1992. He joined National Institute of Advanced Industrial Science and Technology (AIST) in 2002. He currently serves as the group leader of the Grid Infraware Research Group of Information Technology Research Institute, AIST. In the past few years his research has focused on network as a Grid infrastructure. His recent work also includes the G---lambda project which target is to define an interface to manage network as a Grid resource. He is a co---chair of the OGF NSI working group.

John MacAuley received a M.Sc. degree in Computer Science from The University of Western Ontario in 1996 while working as a software designer in the optical networking division of Bell-Northern Research. He had a long career at Nortel performing varying architecture roles within the company ranging from network management software architecture to protocol design and standardization. In 2005 while at Nortel he developed the Dynamic Resource Allocation Controller (DRAC) for dynamic reservation and provisioning of optical, SONET/SDH, and Ethernet services, which later that year was deployed as a service within the SURFnet network. He joined SURFnet in 2009 as a consultant to continue his work as technical lead on the newly open sourced OpenDRAC project.

Joel Mambretti is Director of the International Center for Advanced Internet Research at Northwestern University, which is focused on developing digital communications for the 21st Century. The Center, which was created in partnership with a number of major high tech corporations (www.icair.org), designs and implements large scale infrastructure and applications (metro, regional, national, and global). He is also Director of the Metropolitan Research and Education Network (MREN, http://www.mren.org), an advanced high-performance network interlinking organizations in seven upper-midwest states. With its research partners, iCAIR has established multiple major network research testbeds to develop new architecture and technology for dynamically provisioned communication services and networks, including those based on lightpath switching. iCAIR and its partners also manage the StarLight (www.startap.net/starlight) advanced global communications exchange based on leading-edge optical technologies.

Inder Monga is developing new ways to advance networking services for collaborative and distributed science by leading research and services within ESnet. He also serves as the co-chair of the Network Services Interface working group in the Open Grid Forum. Monga's research interests include network virtualization, network energy efficiency, grid/cloud computing and sensor networking. He currently holds 10 patents and has over 15 years of industry and research experience in telecommunications and data networking at Wellfleet Communications, Bay Networks, and Nortel. He earned his undergraduate degree in electrical/electronics

engineering from Indian Institute of Technology in Kanpur, India, before graduate studies in Boston University's EECS Department.

Guy Roberts received his BEng degree from RMIT University in Australia in 1991 and completed a PhD on integrated semiconductor optical amplifiers for fast packet switching at the University of Cambridge in 2007. His experience in the telecommunications sector began with the rollout of SDH into Telstra's transmission network. In 1995 he moved to Fujitsu as product architect for the multiservice access platform FSX2000 and later relocated to the UK where he was involved in the development of the FDX --- Fujitsu's xDSL access platform. In 2006 Guy joined DANTE where he works with the network engineering and planning team, he is also co---chair of the OGF NSI working group.

Jerry Sobieski joined NORDUnet (Copenhagen, DK) in 2008 as the Director for International Research Initiatives. He received his degree in Computer Science from the University of Houston in 1985. He headed the Laboratory for Parallel Computing at the University of Maryland Institute for Advanced Computer Studies from 1990 to 1997, and then Joined Internet2 to construct the Abilene network in the US from 1998-1999. Mr. Sobieski served as Director of Engineering and then Director of Research at the Mid-Atlantic Crossroads, the regional network in Washington DC, from 2000 to 2008. Mr. Sobieski has been actively involved in protocol development for multi-layer connection oriented services for over 20 years, and is an active participant in the GLIF and the OGF NSI WG. He currently resides in Washington DC and represents the Nordic R&E community in a wide array of advanced networking topics in Europe, the Americas, and the Asia/Pacific region.

Alexander Willner studied Computer Science at the University of Göttingen and received his B.Sc. in 2004 and his M.Sc. in 2006 with specific emphasis on telecommunication networks and distributed systems. Afterwards, he joined the Institute of Computer Science 4 (Communication and Networked Systems) at the University of Bonn as a Research Assistant. His main tasks were planning, coordination and operation of externally funded research projects (such as BMBF VIOLA, IST PHOSPHORUS, and BMBF SLA4D-Grid). Besides this he supervised students in the context of labs, seminars and diploma theses; and he was responsible for basic course tutorials and exams. In January 2012 he joined the Future Internet team at the chair of Next Generation Networks (AV) at the Technical University Berlin as a Research Assistant as well as the group for Next Generation Network Infrastructures (NGNI) at Fraunhofer FOKUS. His work and PhD research focuses on dynamic end-to-end network services and federated network management systems.