Advance Reservation Frameworks in Hybrid IP-WDM Networks

Neal Charbonneau, *Member, IEEE*, Chin Guok, Inder Monga *Member, IEEE*, and
Vinod M. Vokkarane, *Senior Member, IEEE*

***Abstract*—New e-Science and Grid applications require the coordination of geographically distributed scientific instruments along with data and computing resources. Due to the quality of service (QoS) requirements of these applications, these distributed resources can be connected by a wavelength-routed optical network allowing each application to get dedicated bandwidth. These networks are referred to as LambdaGrids. One important service provided in these networks is *advance reservation*. Applications need to coordinate the use of both grid resources and the network. Advance reservation allows these applications to reserve bandwidth in-advance to guarantee availability. In this article, we discuss different networks and frameworks that support advance reservation of bandwidth. We will discuss the general architecture of each network and the type of advance reservation services supported.**

***Index Terms*—advance reservation, scheduled demands, WDM, wavelength-routed, RWA, testbed, and frameworks.**

I. Introduction

A Grid network is a collection of geographically distributed resources, such as storage clusters, super computers, and scientific equipment, that are accessible to users over a network. Examples of e-Science Grids include the Large Hadron Collider Computing Grid Project, the Biomedical Informatics Research Network, and the George E. Brown Network for Earthquake Engineering and Simulation. These networks typically deal with the transfer of large amounts of data in the terabytes and petabytes range. When the Grid resources are connected by application-configurable optical paths (e.g. on an optical wavelength-routed network), the Grid can be considered a LambdaGrid [1].

Optical wavelength-routed wavelength division multiplexing (WDM) networks, or optical circuit switched (OCS) networks are being used to interconnect computing, storage, and instruments in Grid networks. An optical WDM network consists of fibers connected by switches, or optical cross connects (OXCs). In WDM networks, each fiber is partitioned into a number of wavelengths, each of which is capable of transmitting data at very high-speeds. This allows each fiber to support data transmission rates of terabits per second. In order to transmit data over the network, a dedicated circuit is first established when a user submits a connection request. When a connection request arrives at the network, the request must be routed over the physical topology and also assigned a wavelength. The combination of a route and wavelength is known as a *lightpath*. Typically, the objective is to minimize the number of wavelengths used in the network in order to handle a given static traffic demand. In the dynamic traffic scenario, the objective is to minimize the cost of each lightpath request based on different cost metrics, such as path-length and network load. This is known as the *routing and wavelength assignment* (RWA) problem. The RWA problem is NP-complete, therefore heuristics are typically used to solve the problem. The signal may be transmitted all-optically (uses the same wavelength on every hop along the path) or may be converted to electronics and back to optics (O/E/O conversion) at multiple hops.

LambdaGrid networks typically support *dynamic traffic*. Dynamic traffic requests arrive one-by-one according to some stochastic process and they are also released after some finite amount of time. The goal of the provisioning system of the network is to minimize request blocking. A request is said to be blocked if there are not enough resources available to route it. There is extensive work dealing with these problems, e.g. [2].

We can further classify the dynamic traffic model as *immediate reservation* (IR) or *advance reservation* (AR) [3] requests. The data transmission of an IR demand starts immediately upon arrival of the request and the holding time is typically unknown for dynamic traffic or assumed to be open-ended for static traffic. AR demands, in contrast, typically specify a data transmission start time that is sometime in the future and also specify a finite holding time. Fig. 1 shows the difference between an AR and IR requests. We can observe from Fig. 1(a), in IR the resource allocation occurs when the request arrives at the network. In this case, the duration of the request is unknown. In AR (refer Fig. 1(b)), the actual allocation of resources does not occur until a later time. The resources are reserved when the request arrives, but they can be used by other requests before the reservation time. The difference between the arrival of the request and beginning of the transmission is the *book-ahead time*, which is specified by the request. The duration of the request is also specified in advance and known by the network. The fact that holding time and book-ahead time is known allows the network to efficiently optimize resource usage.

Advance reservation was initially proposed for non-optical networks, focusing on circuit-switched networks, packet-switched networks, and ATM networks. Initial work focused on traffic modeling and call admission for telecommunication systems. Additional work was proposed to provide quality-of-service (QoS) for multimedia applications like video conferencing. There were also extensions for RSVP to support advance reservation. Advance reservation for optical networks was first proposed by Zheng and Mouftah in [3].

There are a number of reasons why it is beneficial to provide advance reservation services for Grid applications. With user driven traffic, often bandwidth requirements and request durations are known in advance due to requests being for specific tasks. Advance reservation requests allow applications to ensure network resources are available when certain computing resources are also available; hence enabling user access to certain Grid resources for specified times in the future. In order to access these resources, the user must be able to receive guarantees about network availability. This is known as resource co-allocation.

Also, many Grid applications involve delay-tolerant background or recurring tasks. For example, once a scientific instrument finishes an experiment, the data set usually must be transferred to other sites over the Grid. Instead of issuing these transfers as immediate reservation requests, the user can submit them as advance reservation requests that specify a deadline or window in which the transfer must take place. By providing advance reservation for such tasks, the network can achieve higher utilization while increasing the probability that the Grid applications will be able to successfully reserve the required network resources.

Collaboration is an important part of large-scale scientific computing. Advance reservation can support real-time collaboration through real-time experimentation or high-definition video conferencing. It is easier to allocate resource to advance reservation requests with booking-ahead instead of immediate reservation requests.

In this article we will discuss a number of real-world networks that support advance reservation over optical networks. There is a significant amount of work dealing with advance reservation of network resources in general, but this article will focus on actual implementations.

II. Advance Reservation Frameworks

We now discuss the various advance reservation frameworks and architectures that have been implemented. Some projects support advance reservation across a single domain, Whereas other projects focus on providing multi-domain support as well. There is also work on co-allocation of network and Grid resources (e.g., computing/storage) using advance reservation. Table I provides a comparison of the different frameworks discussed below.

A. On-demand Secure Circuits and Advance Reservation System (OSCARS)

OSCARS is a project funded by the U.S. Department of Energy (DOE) that enables dynamic provisioning and advance scheduling of network resources within a single or over multiple domains between the end user/customer edge devices. The basic architecture of OSCARS is shown in Fig. 2(a). OSCARS is deployed in DOE‘s Energy Sciences Network (ESnet) as a production service to deliver layer 2/3 Virtual Circuits (VCs) to users. OSCARS is implemented as a centralized service that provides web-services programmatic API for peer Inter-Domain Controllers (IDCs) and middleware clients, as well as a Web-Based User Interface (WBUI) for users to make advance reservations (refer Fig. 2(b)). All network service requests to OSCARS requires the type of service (i.e., layer 2 or 3), termination points of the VC, the minimum guaranteed bandwidth, and the duration of use. For advanced users, the option to specify the path via an Explicit Route Object (ERO) is also permitted. OSCARS (version 0.6) consists of 11 distinct pluggable modules; a web-based user interface, a programmatic API, a lookup bridge (tied to an external lookup service), a topology bridge (tied to an external topology service), authentication, authorization, a coordinator, a notification broker, a resource manager, a path computation engine, and a path setup module. A request for network services (i.e., network resources) is first authenticated, and subsequently processed through the Path Computation Engine (PCE) module where one or more atomic PCEs can prune the service topology based on specific constraints, such as bandwidth availability, latency, and authorization. The resulting service topology solution is subsequently stored in the Resource Manager as a confirmed reservation. Just prior to when the reservation becomes active, the Path Setup module contacts the appropriate network elements and initiates the RSVP signaling to setup the VCs. OSCARS is designed to manage resources within a single administrative domain, but supports provisioning of inter-domain virtual circuits using the DICE IDC protocol [1] to communicate with other deployed IDC-compatible instances of OSCARS (e.g., Internet2's ION), or in an interoperable manner with other provisioning systems like AutoBAHN.

B. Dynamic Resource Allocation Controller (DRAC)

Dynamic Resource Allocation Controller (DRAC)[4], developed by Nortel (now part of Ciena), enables automated and dynamic provisioning of network resources to meet application's network requirements. Applications leverage DRAC to request and control their share of network resources, brokering and configuring on an end-to-end basis all the necessary pieces of the network, regardless of the type of network - circuit or packet, wireless or wireline. A key application that DRAC enabled in SURFnet, a R&E network in Netherlands, was "cut-through" capabilities - by steering very large flows of packets or low-latency applications dynamically over Layer 1 circuits instead of Layer 3. For example, instead of routing high-bandwidth point-to-point traffic transfers over intermediate Layer 3 hops (i.e., IP routers) which can incur additional jitter and latency due to buffering, or setting up long-term dedicated and costly high-bandwidth optical connections to address transient bandwidth demands, DRAC simply sets up and takes down "ephemeral" optical circuits as needed by the application, minute-by-minute, hour-by-hour.

The DRAC core framework (refer Fig. 3(a)) includes authentication, authorization, accounting (AAA) services, policy engine, topology discovery engine, resource reservation services, workflow utilities, inter-domain routing facilities, and smart bandwidth management capabilities. Its interface provides applications with the means to directly drive their share

[1] http://www.controlplane.net/

of network resources within a policy-defined envelope of flexibility while receiving abstracted availability and performance information from the network. Network resources include bandwidth, acceleration appliances, and sensors, to name a few. The DRAC interface to the network uses existing management protocols and standards like SNMP/TL1 and toolsets for interfaces (equipment CLI's), which greatly simplifies deployment in multi-vendor, multi-technology environments. DRAC is now available as an open-source implementation called OpenDrac [2].

C. EnLIGHTened

EnLIGHTened (initially funded by the U.S. NSF) is a project that focuses on advance reservation of both Grid and network resources (lightpaths) [5]. The goal of the project is to allow Grid applications to request in-advance or on-demand compute, storage, instrumentation, and network resources. These resources have to be co-allocated and may be spread across multiple domains. The architecture includes a resource broker (ERB), resource monitoring, and network scheduling (in a Network Domain Manager). The architecture utilizes the Highly-Available Resource Co-allocator (HARC) system, which allows clients to reserve multiple distributed resources in one step. HARC consists of Acceptors that manage co-allocation and Resource Managers that are the interfaces used to make reservations. Custom acceptors and resource managers were implemented specifically for the project. The resource broker provides the primary interface to the client and the ERB uses HARC internally. The architecture of the ERB is shown in Fig. 4(a). The ERB accepts requests from the user, who is authenticated through the AAA module. From the user request, the broker then gets availability information from each of the local managers (e.g., the storage managers and network managers). With this information, it uses its own scheduler to find a resource set able to satisfy the request. Given a set of resources, it then uses HARC acceptors to co-allocate the required resources. The acceptors in turn use the HARC Resource Managers to schedule the final resources with the local managers. The local managers are then responsible for activating and deactivating resources at the proper times.

The Domain Network Manager (DNM) is responsible for controlling network resources, resource reservation and path computation, restoration, and dynamic teardown/setup of lightpaths for a single administrative domain. Each domain will have an DNM. An example of a DNM is shown in Fig. 4(b).

The Discovery and Monitoring System (DMS) is required to support the dynamic addition or removal of resources. It is also responsible for monitoring the performance and reliability of resources in real-time. There are three types of resource monitoring. At-reservation-time monitoring is used to verify lightpath establishment. During-reservation monitoring is used to assess QoS/SLA requirements are being met. Ongoing grid resource status monitoring is used to collect relevant performance metrics from grid resources in real time. All the monitoring types use the perfSONAR framework. Details of this module can be found in [6].

Lastly, EnLIGHTened also emphasizes the importance of standardized interfaces. Interaction with all local managers is done through well defined interfaces, such as Grid Security Infrastructure (GSI) and Grid Network Infrastructure (GNI). The project as promotes collaboration with other groups to define a GNI as well as an application interface, Grid Application Interface (GAI).

D. G-Lambda

Another project with similar goals was the Japanese project, G-Lambda [7]. The main goal of the project was to define a standard web interface between a grid resource scheduler (similar to the ERB) and network resource management systems (similar to DNM). Network operators could then implement this interface to allow the grid resource coordinators to make advance reservations of network resources. The grid resource coordinators are implemented as middleware and coordinate both network and computation/storage resources at the request of grid applications.

An overview of the G-lambda architecture is shown in Fig. 5(a). A detailed view of the architecture can be seen in Fig. 5(b) showing the modules that will be discussed. As introduced above, the main focus on the project was to define the Grid Network Service-Web Service Interface (GNS-WSI) shown in the figure. The GNS-WSI allows a number of commands to be sent to the network resource manager (NRM). These include commands to reserve paths, modify previously reserved paths, release paths, and query path or status information. An example architecture of an NRM is shown in Fig. 5(c).

The Grid Resource Scheduler (GRS) provides a web service interface to Grid clients using the Web Service Resource Framework (WSRF). The GRS is implemented using the Globus Toolkit 4. The modules include a client API, the web service module, and a scheduling module that uses the CRMs and NRMs to co-allocate and reserve resources. The scheduling module consists of a co-allocator and planner. The planner takes the user requests and selects a candidate set of resources, which are then reserved simultaneously by the co-allocator. A two-phase commit protocol is used to reserve resources in parallel. A detailed description of the modules and protocols can be found in [8].

The CRMs are essentially wrappers around existing software like PluS and GridEngine. The project implemented two NRMs (one by KDDI and another by NTT). The NRMs are responsible for path virtualization between endpoints, local scheduling, and activation/de-activation of lightpaths. Paths are virtualized to hide implementation details. NRMs have a web service module (interface), a mediation module which virtualizes the GMPLS optical network and performs the scheduling based on requests from the GRS, a network control module for the GMPLS switches to manage state information. Details about NRMs can be found in [9].

EnLIGHTened and G-Lambda established a collaboration to prove network co-allocation across the two network domains. Software wrappers around components in the architectures were used so that the systems could interoperate. The technical details can be found in [10].

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

E. PHOSPHOROUS

The EU's PHOSPHORUS project also incorporates the advance reservation of Grid and networking resources [11]. The goal of PHOSPHORUS is to provide on-demand and in-advance end-to-end provisioning of network and grid resources across multiple domains and multiple vendors. The PHOSPHORUS project comprises of two phases. In the first phase each independent domain is controlled by an existing Network Resource Provisioning System (NRPS), while in the second phase interoperability is added with other existing networks and Grid resources through standardized interfaces. The NRPSs are similar to the NRM/DRM of the previous two projects.

PHOSPHORUS defines an architecture composed of three planes. The first is the *service plane*, which consists of middleware extensions to allow applications to make advance reservations, co-allocate resources, provide AAA across multiple domains, etc. The next layer is the *network resource provisioning* plane. This is an adaptation layer between the service plane and existing NRPSs. Lastly, there is the *control plane*, which is used to control physical resources.

The general architecture is shown in Fig. 6. The project started with a centralized architecture with only one service plane (global broker). The service plane (network service plane, NSP) is responsible for finding end-to-end paths, managing AAA, etc. The individual NRPSs, shown underneath the NSP manage their own domains and intra-domain routing. Each NRSP has its own control plane, such as GMPLS. All interfaces are deployed as web services.

At present, PHOSPHORUS architecture works with three existing NRPSs: ARGON [12] from Germany, DRAC from Nortel, and User Controlled Lightpaths (UCLP) from CANARIE in Canada. The NRSPs publish border end-points to the NSP, which can then do inter-domain routing while NRSPs perform intra-domain routing based on the selected end-points. Each NPRS provides the ability to make a reservation request, cancel a reservation, get the status of a request, and bind and activate requests. This functionality is provided by an NPRS adapter which abstracts each specific NRPS. A number of interfaces (NBI, EWI, SBI), shown in the figure, for communication between all the components are defined.

The general flow when a grid request is received is as follows. The grid application or grid middleware sends a request to the reservation web service of the NSP. Once validated by the AAA module and available resources are found, the NSP finds an inter-domain path. With the inter-domain path selected, each independent domain is notified and the lookup of an intra-domain path is initiated.

Harmony [13] is the network service provisioning system in PHOSPHORUS. Harmony assumes a group of independent NRPSs and provides an abstract service plane to do end-to-end co-allocation. It defines three types of architectures: a centralized approach, hierarchical, and distributed. In each case, a Harmony inter-domain broker sits above each of the NRPSs. In the case of a centralized system there is only one, whereas in the case of a distributed system, there is one for each NRPS. A Harmony Service Interface (HSI) is used to exchange abstract topology and other information between the brokers and the NRPSs.

As discussed previously, Harmony is responsible for inter-domain paths while each NRPS is responsible for intra-domain paths. The NRPSs provide border endpoints and abstract links to Harmony. A reservation specifies one or more services, where each service requires one or more connections. The requests can be fixed or malleable advance reservation requests. A simple iterative search over the inter-domain paths is used for malleable requests, trying different start times and different bandwidths. For each path, Harmony checks whether each individual domain can create an intra-domain path or not.

Grid Enabled GMPLS (G^2MPLS) [14] is the new control plane defined in PHOSPHORUS which extends GMPLS to work with, both Grid and network resources. The goal is to incorporate *grid network services* (GNS) into GMPLS. GNS is a service that allows provisioning of network and grid resources in a single step. G^{2}MPLS allows selection, co-allocation, and maintenance of grid and network resources. Some of the extensions include:

- Discovery and advertisement of Grid capabilities (e.g. number of CPUs) and network resources (e.g. bandwidth).
- Service setup, which involves coordination with Grid middleware, configuration of network connections, advance reservation, etc.
- Service monitoring.

G^{2}MPLS provides two models of operation. One is an overlay model designed for legacy systems that only supports standard GMPLS. In this case, NRPSs still exist and can use G^{2}MPLS in the same way they would use GMPLS. The other model is the integrated model, where G^{2}MPLS provides an interface directly to the NSP. It can then be used to co-allocate grid and network resources for a given domain. The basic unit of work is a grid job.

III. Conclusion

In this article we discussed a number architectures for supporting advance reservation of network resources. Supporting advance reservation is becoming increasingly important due to the requirements of Grid and e-Science applications. These applications must coordinate geographically distributed resources and must be able to reserve network bandwidth in-advance.

The various frameworks discussed in the paper followed very similar architectural concepts of abstraction and orchestration to reserve, allocate, and provision dynamic circuits across heterogenous network resources. These concepts and high-level design can be leveraged to implement similar techniques to meet requirements of commercial Cloud Computing providers.

IV. Acknowledgement

We would like the thank John MacAuley for providing access to a DRAC test system for screen captures and the anonymous reviewers for their insightful comments and suggestions to improve the quality of the article.

References

[1] T. DeFanti, M. Brown, J. Leigh, O. Yu, E. He, J. Mambretti, D. Lillethun, and J. Weinberger, "Optical switching middleware for the OptiPuter," *IEICE Transactions on Communications [Invited]*, pp. 2263–2272, 2003.

[2] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *SPIE Optical Networks Magazine*, vol. 1, pp. 47–59, 2000.

[3] J. Zheng and H. T. Mouftah, "Routing and wavelength assignment for advance reservation in wavelength-routed WDM optical networks," in *Proceedings, IEEE International Conference on Communications (ICC)*, 2002, vol. 5, pp. 2722–2726.

[4] D.B. Hoang, T. Lavian, S. Figueira, J. Mambretti, I. Monga, S. Naiksatam, H. Cohen, D. Cutrell, and F. Travostino, "DWDM-RAM: an architecture for data intensive services enabled by next generation dynamic optical networks," in *Proceedings, IEEE GLOBECOM*, Nov. 2004, pp. 400 – 409.

[5] L. Battestilli, A. Hutanu, G. Karmous-Edwards, D.S. Katz, J. MacLaren, J. Mambretti, J.H. Moore, Seung-Jong Park, H.G. Perros, S. Sundar, S. Tanwir, S.R. Thorpe, and Yufeng Xin, "EnLIGHTened computing: An architecture for co-allocating network, compute, and other Grid resources for high-end applications," in *International Symposium on High Capacity Optical Networks and Enabling Technologies*, Nov. 2007, pp. 1 –8.

[6] S. Tanwir, L. Battestilli, H. Perros, and G. Karmous-Edwards, "Monitoring and Discovery for EnLIGHTened Computing," in *High-Capacity Optical Networks and Enabling Technologies*, 2007.

[7] Atsuko Takefusa, Michiaki Hayashi, Naohide Nagatsu, Hidemoto Nakada, Tomohiro Kudoh, Takahiro Miyamoto, Tomohiro Otani, Hideaki Tanaka, Masatoshi Suzuki, Yasunori Sameshima, Wataru Imajuku, Masahiko Jinno, Yoshihiro Takigawa, Shuichi Okamoto, Yoshio Tanaka, and Satoshi Sekiguchi, "G-lambda: Coordination of a Grid scheduler and lambda path service over GMPLS," *Future Generation Computer Systems*, vol. 22, no. 8, pp. 868 – 875, 2006.

[8] A. Takefusa, H. Nakada, T. Kudoh, Y. Tanaka, and S. Sekiguchi, "GridARS: An advance reservation-based Grid co-allocation framework for distributed computing and network resources," in *Internatioanl Workshop Job Scheduling Strategies for Parallel Processing*, 2007, pp. 152–168.

[9] M. Hayashi, H. Tanaka, and M. Suzuki, "Advance reservation-based network resource manger for optical networks," in *Proceedings, Optical Fiber Communication Conference (OFC)*, 2008, pp. 1 –3.

[10] Steven R. Thorpe, Lina Battestilli, Gigi Karmous-Edwards, Andrei Hutanu, Jon MacLaren, Joe Mambretti, John H. Moore, Kamaraju Syam Sundar, Yufeng Xin, Atsuko Takefusa, Michiaki Hayashi, Akira Hirano, Shuichi Okamoto, Tomohiro Kudoh, Takahiro Miyamoto, Yukio Tsukishima, Tomohiro Otani, Hidemoto Nakada, Hideaki Tanaka, Atsushi Taniguchi, Yasunori Sameshima, and Masahiko Jinno, "G-lambda and enLIGHTened: wrapped in middleware co-allocating compute and network resources across Japan and the US," in *GridNets '07: Proceedings of the first international conference on Networks for grid applications*, 2007.

[11] S. Figuerola, N. Ciulli, M. de Leenheer, Y. Demchenko, W. Ziegler, and A. Binczewski, "PHOSPHORUS: single-step on-demand services across multi-domain networks for e-Science," in *SPIE*, 2007.

[12] M. Pilz, C. Barz, U. Bornhauser, P. Martini, C. deWall, and A. Willner, "ARGON: Reservation in Grid-enabled networks," in *DFN Forum on Communication Technologies*, 2008.

[13] A. Willner, C. Barz, J. A. G. Espin, J. F. Riera, S. Figuerola, and P. Martini, "Harmony-advance reservations in heterogeneous multi-domain environments," in *Proceedings, International IFIP-TC 6 Networking Conference*, 2009.

[14] G. Zervas, E. Escalona, R. Nejabati, D. Simeonidou, G. Carrozzo, N. Ciulli, B. Belter, A. Binczewski, M.S. Poznan, A. Tzanakaki, and G. Markidis, "Phosphorus grid-enabled GMPLS control plane (GMPLS): architectures, services, and interfaces," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 128 –137, Jun. 2008.

Neal Charbonneau (S'08) received the B.S. degree in Computer Science and the M.S. degree in Computer Science from the University of Massachusetts, Dartmouth in 2008 and 2010, respectively. His interests include computer networks and software design and development.

Chin P. 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 to the Open Grid Forum On-Demand Infrastructure Service Provisioning Working Group.

Inder Monga (M'00) is developing new ways to advance networking services for collaborative and distributed science by leading research and services within ESnet. He contributes to ongoing ESnet research projects, such as Advanced Resource Computation for Hybrid Service and TOpology NEtworks (ARCHSTONE) and ANI-Testbed as well as to the OSCARS and Advanced Network Initiative (ANI) 100G projects. 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 coming to the U.S. to pursue his graduate studies in Boston Universitys Electrical and Electronics Communication Systems Department.

Vinod M. Vokkarane (S'02-M'04-SM'09) received the B.E. degree with Honors in Computer Science and Engineering from the University of Mysore, India in 1999, the M.S. degree in Computer Science from the University of Texas at Dallas in 2001, and the Ph.D. degree in Computer Science from the University of Texas at Dallas in 2004. He is a recipient of the Texas Telecommunication Engineering Consortium Fellowship 2002-03, and the University of Texas at Dallas Computer Science Dissertation of the Year Award 2003-04. Dr. Vokkarane is the co-author of a book, "Optical Burst Switched Networks," Springer, 2005. Dr. Vinod Vokkarane is currently an Associate Professor of Computer and Information Science at the University of Massachusetts, Dartmouth. His primary areas of research include design and analysis of architectures and protocols for optical and wireless networks.

TABLE I
COMPARISON OF ADVANCE RESERVATION FRAMEWORKS.

Framework	Region and Funding Body	Bandwidth Provision	Provisioning Layer	Network Resource Provisioning System	Grid Co-Scheduling Capabilities	Single-/Multi-Domain
OSCARS	U.S.A. (U.S. Department of Energy)	Hybrid: packet/circuit	Layer 2 & 3	Integrated (MPLS-based)	No	Single (Multi-domain enabled with IDC)
DRAC	Europe (SurfNet)	Hybrid: packet/circuit	Layer 1 & 2	Integrated	No	Single
EnLIGHTened	U.S.A. (U.S. NSF)	Circuit	Layer 1	Integrated (GMPLS)	Yes	Multi-domain
G-Lambda	Japan (KDDI R&D, NTT, NICT and AIST)	Circuit	Layer 1 & 2	Integrated (GMPLS)	Yes	Multi-domain
PHOSPHORUS	Europe (E.C. FP6)	Circuit	Layer 1 & 2	ARGON, DRAC and UCLP	Yes	Multi-domain

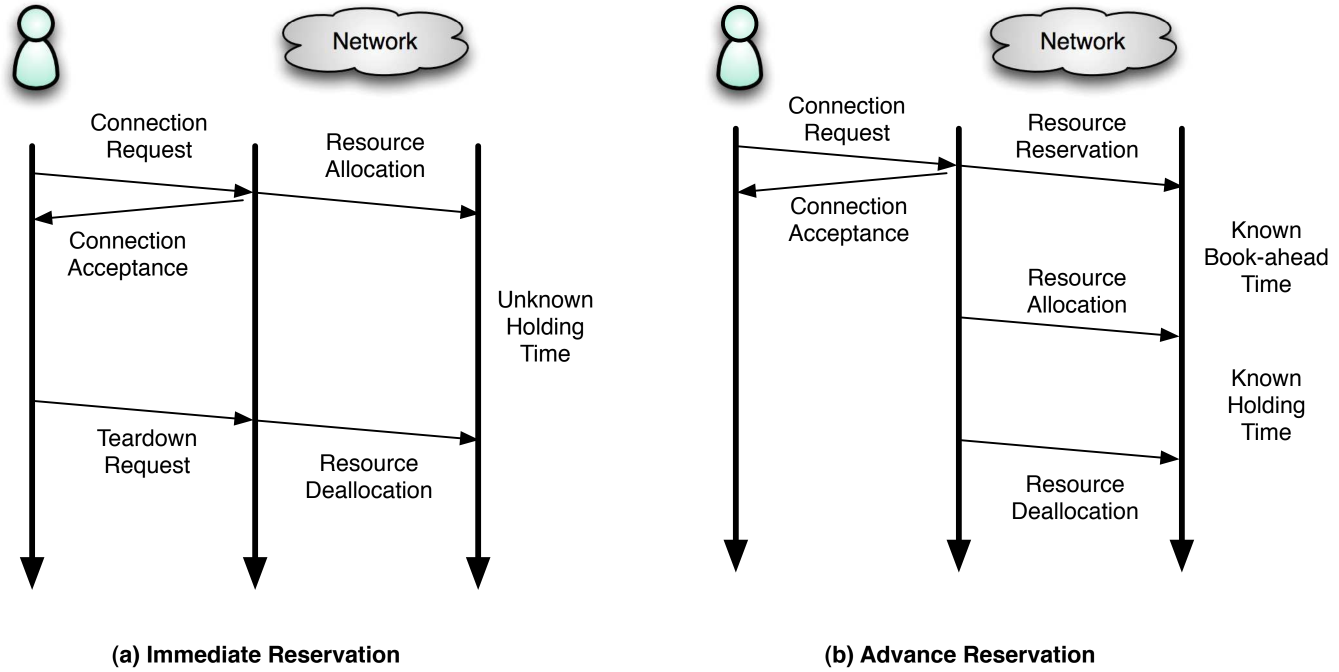
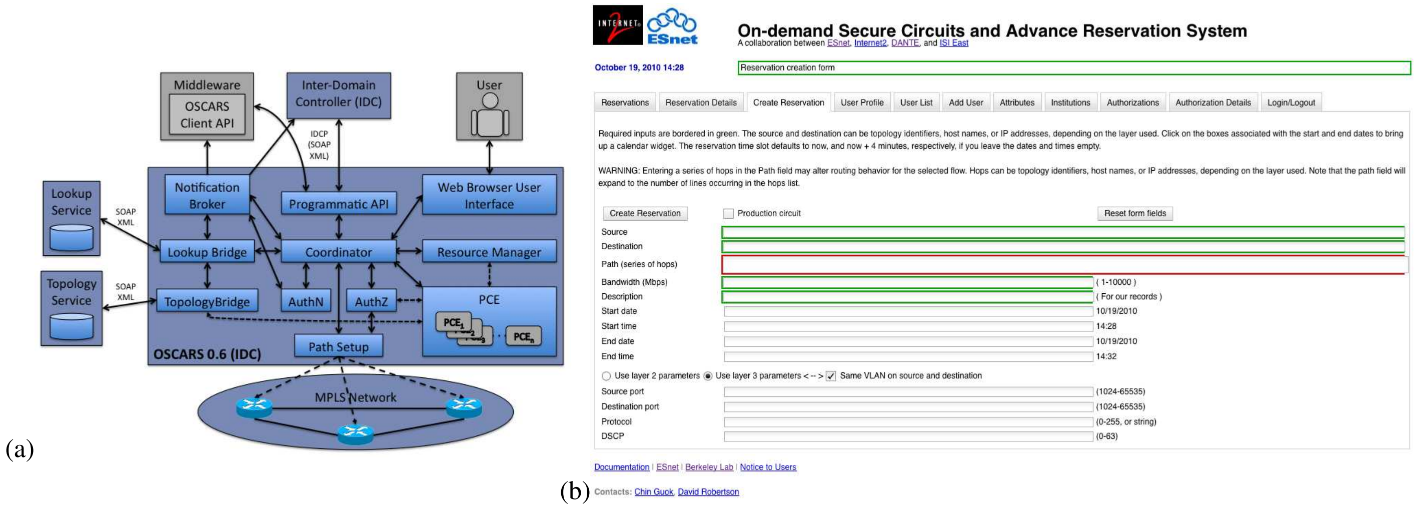


Fig. 1. The request and allocation of resources for immediate and advance reservation requests. In the figure, we assume that the requested resources are available. Before reservation/allocation the network must find an appropriate lightpath. For immediate reservation (a), the allocation is at the same time as the request arrival and the duration is typically unknown. For advance reservation (b) the allocation is some (known) time after the arrival and the duration is also known.



On-demand Secure Circuits and Advance Reservation System

A collaboration between ESnet, Internet2, DANTE, and ISI East

October 19, 2010 14:28

Reservation creation form

Reservations | Reservation Details | Create Reservation | User Profile | User List | Add User | Attributes | Institutions | Authorizations | Authorization Details | Login/Logout

Required inputs are bordered in green. The source and destination can be topology identifiers, host names, or IP addresses, depending on the layer used. Click on the boxes associated with the start and end dates to bring up a calendar widget. The reservation time slot defaults to now, and now + 4 minutes, respectively, if you leave the dates and times empty.

WARNING: Entering a series of hops in the Path field may alter routing behavior for the selected flow. Hops can be topology identifiers, host names, or IP addresses, depending on the layer used. Note that the path field will expand to the number of lines occurring in the hops list.

Create Reservation   □ Production circuit   Reset form fields

Source		
Destination		
Path (series of hops)		
Bandwidth (Mbps)		(1-10000)
Description		(For our records)
Start date		10/19/2010
Start time		14:28
End date		10/19/2010
End time		14:32

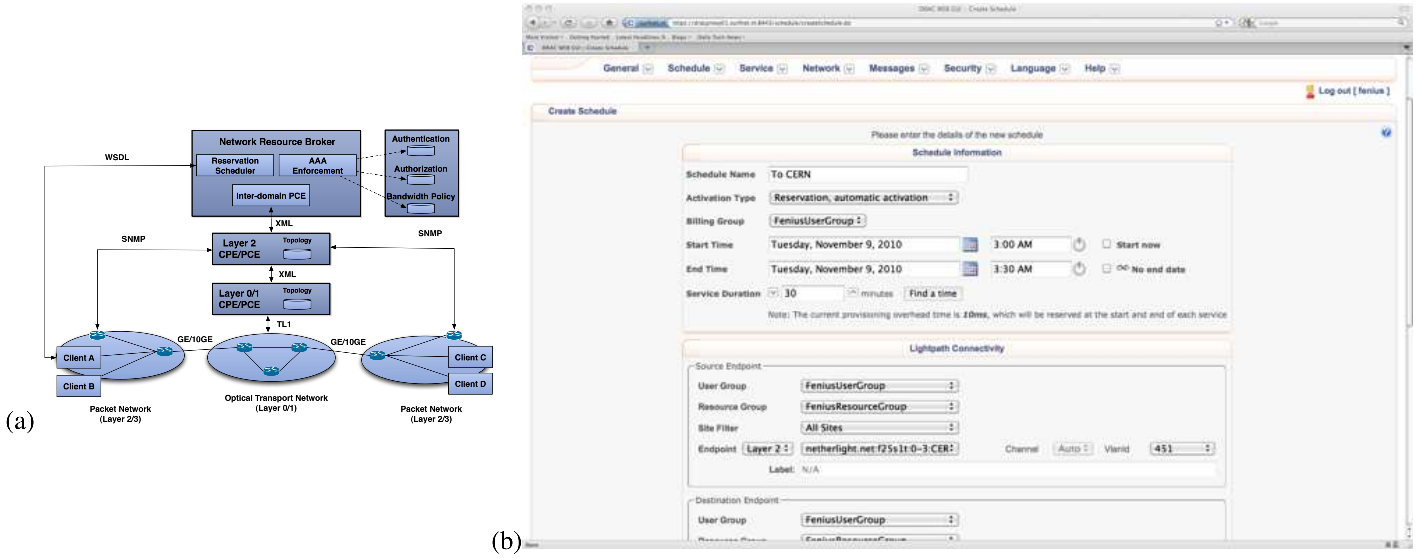
○ Use layer 2 parameters ● Use layer 3 parameters < -- > ☑ Same VLAN on source and destination

Source port		(1024-65535)
Destination port		(1024-65535)
Protocol		(0-255, or string)
DSCP		(0-63)

Documentation | ESnet | Berkeley Lab | Notice to Users

Contacts: Chin Guok, David Robertson

Fig. 2. (a) Overview of OSCARS architecture. (b) Screenshot of OSCARS reservation system.



Create Schedule

Please enter the details of the new schedule

Schedule Information

Schedule Name	To CERN		
Activation Type	Reservation, automatic activation		
Billing Group	FeniusUserGroup		
Start Time	Tuesday, November 9, 2010	3:00 AM	□ Start now
End Time	Tuesday, November 9, 2010	3:30 AM	□ No end date
Service Duration	30 minutes	Find a time	

Note: The current provisioning overhead time is 10ms, which will be reserved at the start and end of each service

Lightpath Connectivity

Source Endpoint

User Group	FeniusUserGroup
Resource Group	FeniusResourceGroup
Site Filter	All Sites
Endpoint Layer 2	netherlight.net:f25s1t:0-3:CER
Channel	Auto
Vlanid	451
Label:	N/A

Destination Endpoint

User Group	FeniusUserGroup

Fig. 3. (a) Overview of DRAC architecture. (b) Screenshot of DRAC reservation system.

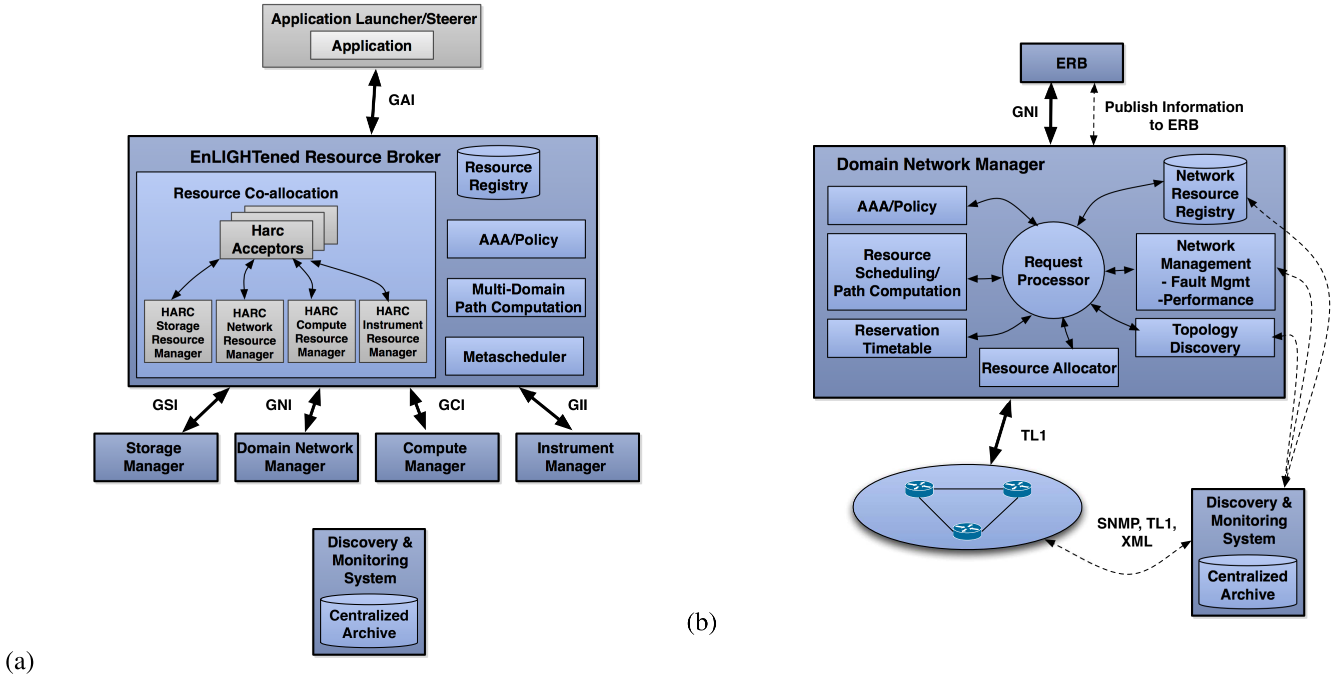

Application Launcher/Steerer
Application
GAI
EnLIGHTened Resource Broker
Resource Registry
Resource Co-allocation
Harc Acceptors
AAA/Policy
Multi-Domain Path Computation
HARC Storage Resource Manager
HARC Network Resource Manager
HARC Compute Resource Manager
HARC Instrument Resource Manager
Metascheduler
GSI
GNI
GCI
GII
Storage Manager
Domain Network Manager
Compute Manager
Instrument Manager
Discovery & Monitoring System
Centralized Archive
(a)
ERB
GNI
Publish Information to ERB
Domain Network Manager
AAA/Policy
Network Resource Registry
Resource Scheduling/ Path Computation
Request Processor
Network Management - Fault Mgmt -Performance
Reservation Timetable
Topology Discovery
Resource Allocator
TL1
SNMP, TL1, XML
Discovery & Monitoring System
Centralized Archive
(b)

Fig. 4. Overview of EnLIGHTened architecture.

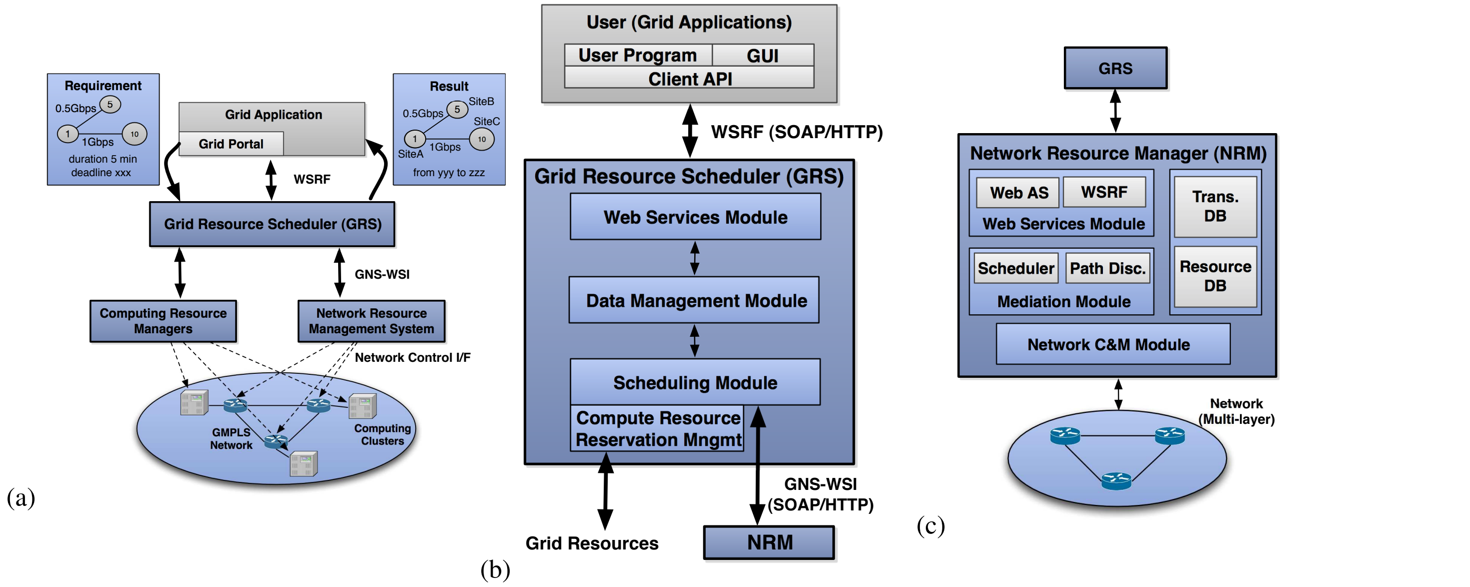


Fig. 5. (a) Overview of G-lambda architecture. Detailed view of the G-lambda architecture (b) Grid Resource Scheduler and (c) Network Resource Manager.

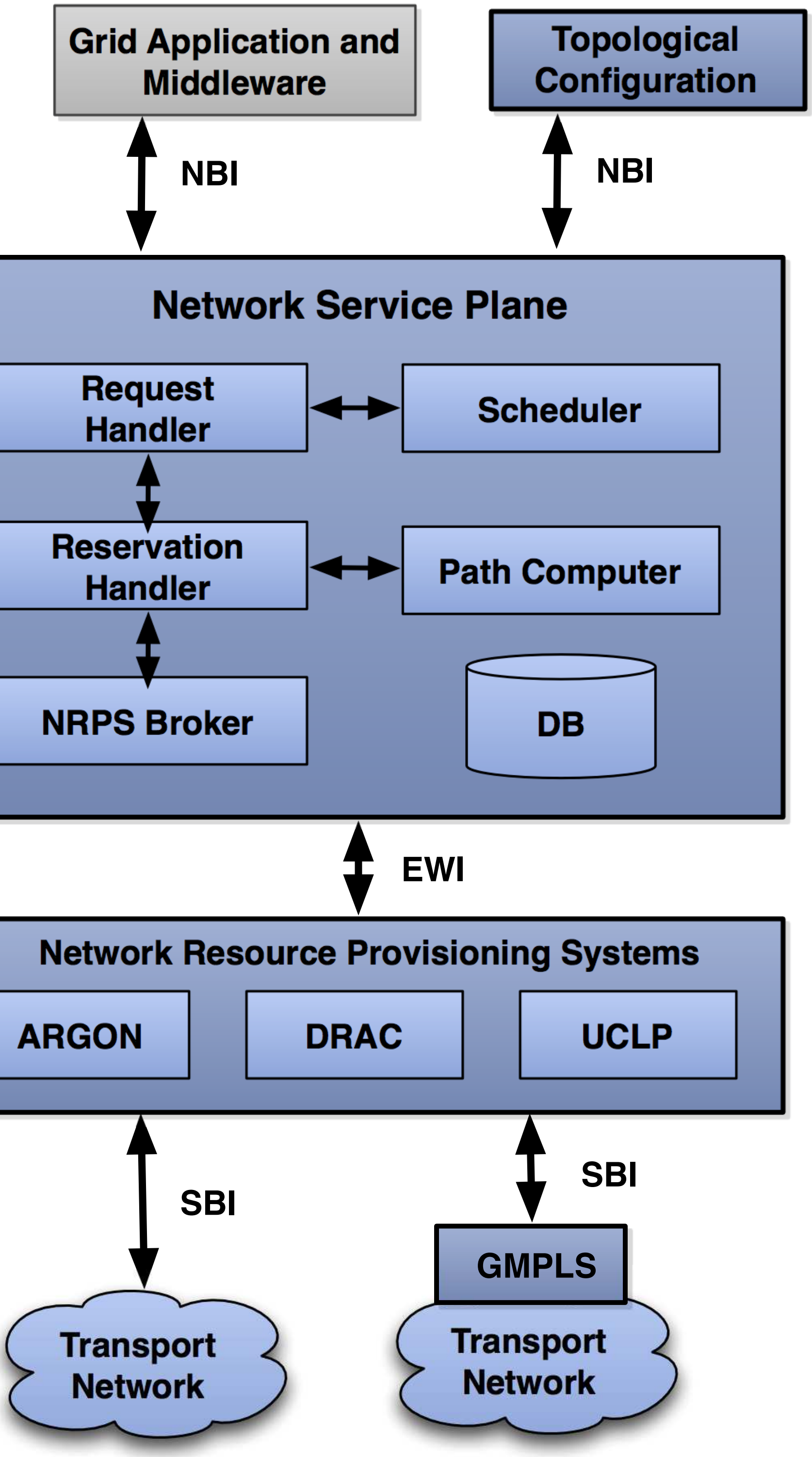


Fig. 6. Overview of PHOSPHORUS architecture.