

**Project Title:** Phoebus: Network Middleware for Next-Generation Network Computing

**Principal Investigator:** Martin Swany

**Organization:** University of Delaware, Department of Computer and Information Sciences

**Address:** 18 Amstel Avenue, Rm. 101, Newark, DE 19716

---

## **1 Summary**

The Phoebus project investigated algorithms, protocols, and middleware infrastructure to improve end-to-end performance in high speed, dynamic networks. The Phoebus system essentially serves as an adaptation point for networks with disparate capabilities or provisioning. This adaptation can take a variety of forms including acting as a provisioning agent across multiple signaling domains, providing transport protocol adaptation points, and mapping between distributed resource reservation paradigms and the optical network control plane. We have successfully developed the system and demonstrated benefits. The Phoebus system was deployed in Internet2 and in ESnet, as well as in GEANT2, RNP in Brazil and over international links to Korea and Japan.

Phoebus is a system that implements a new protocol and associated forwarding infrastructure for improving throughput in high-speed dynamic networks. It was developed to serve the needs of large DOE applications on high-performance networks. The idea underlying the Phoebus model is to embed Phoebus Gateways (PGs) in the network as on-ramps to dynamic circuit networks. The gateways act as protocol translators that allow legacy applications to use dedicated paths with high performance.

### **Project Highlights:**

- Transparent ESnet OSCARS and Internet2 DCN circuit allocation (Described in [6, 7].)
- Phoebus gateways deployed by Internet2 (Described [3].)
- Phoebus gateways deployed by ESnet (Described in [9].)
- Experimentation with Phoebus over international links (Described in [7].)
- Direct GridFTP integration with XIO module (Described in [6].)

- Phoebus forwarding over 9Gb/sec (Described in [6, 7].)
- Buffer and Burst with dynamic circuit allocation (Described in [5, 8].)

## 2 Year 1

During the initial project period, we changed focus to accommodate growing understanding of DOE target application environments and networks. Essentially, the project targeted agile, optical networks, with the UltraScience Net as an exemplar. The initial emphasis of the Phoebus project was enabling legacy applications and heterogeneous networks. In Year 1, we broadened our focus to include scheduling in next-generation all-optical networks, interdomain optical peering, and protocols for reservation and signaling.

The goal of the Phoebus project was to understand and leverage the radical changes that will come about due to the emergence of agile, optical networks typified by the DOE UltraScience Net. To achieve this end, our initial aim was to investigate the technological gap between current and desired functionalities and to:

- Design new capabilities and protocols to address the gap, and
- Provide a body of theory through simulation and emulation to aid in understanding these emerging environments.

**Simulation/Emulation Environment** - The first phase of this new emphasis centered around the investigation of a hybrid simulation/emulation environment called PhoSim. The hybrid simulation/emulation environment was intended to investigate three main areas that are critical to efficient and effective utilization of end-to-end switched circuits.

- **Distributed Advance Reservation** As optical networks scale up in size, it is critical that the reservation process be distributed. It must be possible to reserve network resources in advance, as applications often know their requirements well before they actually need the resources, and critical applications must be able to reserve appropriate resources. Scheduling and reservation have a distinct impact on utilization and on usability. This part of the project studied the quantification of efficiency with various scheduling algorithms and approaches.

In Year 1, we focused on modeling LSPs and examining appropriate algorithms to maximize bandwidth. The optimal solution of this problem is NP-hard, thus we investigated approximation algorithms that produce workable

solutions. We experimented with a kernel-level implementation of RSVP for Linux and got a small PhoSim testbed operational.

- **Interdomain Optical Peering** Another important challenge for signaling and reservation is broadening these activities to multiple network administrators. Peering agreements between networks may specify reservation budgets. The problem lies in the fact that these networks may have multiple points of contact. Again a single scheduling location will not scale. The challenge is to develop a distributed policy engine for multi-domain optical peering.

In Year 1, we began to examine distributed algorithms that group interdomain policies into distributed access locations that encapsulated site policy while shielding the individual switching elements from excessive state maintenance.

- **Grid/Web Services Interfaces** The Grid and Web Services environment offer a rich set of protocols that can be leveraged in optical network environments. The problems that optical circuit networks will face as they become widespread include advance reservation, authorization and negotiation. Each of these areas has potentially rich policy that needs to be expressed. This part of the Phoebus effort will produce application interfaces that represent the wide range of policies that are possible as well as providing applications an emulated environment that will allow them to be modified to use optical network technology before they are actually connected to a testbed network.

In Year 1 we studied various Grid/Web Service approaches, and continued engaging with various application communities to understand their requirements.

### 3 Year 2

The goal of this research was to explore next-generation network environments for high-performance distributed computing. Both the core network protocols and the way in which networks are used have been relatively static for a number of years. The emergence of environments such as the DOE UltraScienceNet, driven by the ever-increasing demands on the network by DOE science applications, indicate that the community must continue to investigate new network architectures to ensure that the needs of applications are met. Serving that interest, we investigated a novel set of protocols and middleware to help build this bridge to the future of high-impact science.

The core of this work was the development and investigation of new network services, exposed via a session layer. As we have indicated in other work, releasing applications from a direct dependence on an end to end transport layer connection offers many opportunities for functional and performance improvements. Essentially the historic binding of transport to network layer stems from the decision to superficially hide all differences in various network components and take a one size fits all approach. In the modern world, we must deal with heterogeneity of network technology and administrative domains and policies. The essence of the functional improvements can be described as providing a framework for explicit negotiation as a part of connection establishment that can adapt an end to end communication based on the various networks it traverses. Instead of a simple, end-to-end connection, end-systems would have a session view that linked together the services in each network, allowing us to deal with the heterogeneity that exists between networks.

Another new direction of this investigation was targeting implementation on network processors. These systems permit complex packet handling operations at line rate speeds. Network processors have gotten significantly more powerful over the generations, currently able to operate at 10Gb/sec, and there is no reason to think that this trend would not continue. Based on their current viability and their future capabilities, these systems seem like perfect matches for implementing the ideas of the Phoebus project in an easy to deploy setup, and thus, were a major focus of the middleware to be produced by the project. The project developed a preliminary implementation on the Intel IXP 2400 network processor, and we investigated techniques that were viable in this environment for transport protocol and control signal modification.

- **Session protocol design and implementation** The eXtensible Session Protocol was revised approaching a first viable version. This protocol is extensible via the use of options, but it currently explicitly targets authentication and authorization, transport and traffic engineering transformation points, as well as the negotiation based on the intersection of current network conditions and policy.
- **Preliminary implementation experience on Intel IXP2400 NP platform** We deployed a functioning Intel IXP2400 development environment and a basic working prototype. The IXP system that we had featured 3 Gbit Ethernet interfaces only, but it contained the same micro-architecture as the newer IXP 2800, which can support 10Gbit Ethernet as well as SONET OC-192.
- **GSI authentication implementation and protocol support** We developed protocol and implementation support for GSI authentication in the session

framework. This functionality enabled an end system to dynamically open forwarding channels in the network after transparently authenticating with X.509 credentials.

- **Link-state routing protocol to exchange flow information inside an administrative domain to enable policy evaluation** This protocol was designed to address the case where networks peer in multiple locations and when the admission of new flows should be sensitive to the current traffic patterns between domains and their associated policy.
- **Network Models** In our efforts, we came to the conclusion that the models we used for heterogeneous, multi-technology networks are the key to pathfinding as well as interdomain optical peering and advanced reservation (as investigated in Year 1). We worked to develop network models based on the topology models in perfSONAR that could address these use cases, and we added support for optical networks and reservations. We worked with Internet2, GEANT and ESnet in the DICE collaboration to develop the network model used in OSCARS for the Interdomain Control Protocol (IDCP), which was described in [4].

We have experimented with using the Security Assertion Markup Language (SAML) to allow indirection in the way in which resources are authorized. Thus rather than have network access tied to an IP address or user, it could be allowed based on site-based certificates.

- **Transport protocol translation investigation on the IXP** We began work on the design and implementation and of line-speed TCP termination for multi-transport connections. This allowed a network device to do transport-layer transcoding within a session. This also translated a single transport connection to a series of shorter, better-performing TCP connections, potentially utilizing a rate-based protocol in parts of the network where flows are isolated, namely dynamic circuits or lambdas.
- **Investigation and evaluation of signaling and traffic engineering translation points** We began to investigate support for RSVP-TE and the ability to evaluate the addition of Explicit Route options to the Path messages. In addition, we began to work with network models (described above) in which DSCPs and MPLS labels can be translated at administrative boundaries.
- **PhoSim** - We implemented an emulation testbed that is described in [6, 7].

## 4 Year 3

The Phoebus project focused on two main emphases. The first was scheduling, reservation and management of optical networks within distributed computing environments such as the Grid. The second focused on our session layer protocol that enables adaptation and negotiation at line speed in the network.

- Scheduling, reservation and management of optical networks in the Grid
  - Security assertion markup language (SAML) models to express policy
  - Grid services resource broker for scheduling and reservation
  - Work with Internet2 and Geant2 to express cross domain models of advanced networks
- Extensible Session Protocol for heterogeneous networks
  - Session control abstraction for unification of disparate signaling domains
  - Dynamic firewall configuration with automatic credential presentation and policy negotiation
  - Intel IXP Network Processor implementation for scalable session protocol implementation

The goal of the Phoebus project was to understand and leverage the radical changes that will come about due to the emergence of agile, optical networks such as the DOE UltraScience Net.

- **Grid Services interface** The Grid and Web Services environment offer a rich set of protocols that can be leveraged in optical network environments. The problems that optical circuit networks will face as they become widespread include advance reservation, authorization and negotiation. Each of these areas has potentially rich policy that needs to be expressed. This part of the Phoebus effort produced application interfaces that represent the wide range of policies that are possible. We continued to engage with various application communities, as well as other advanced network organization such as Internet2 and Dante in Europe, in order to understand and contribute to their requirements. We worked toward another resource brokering reservation platform that is based on Grid service technology. We developed a Web Services interface to Phoebus, as well as implemented support to act as a client to DRAGON and OSCARS for dynamic path creation.

- **Network models** One of the key efforts in exposing critical network functionality to distributed computing applications was the way in which those networks are represented. This strongly impacts the ease with which various operations can be performed. We continued working with Internet2, GEANT and ESnet in the DICE collaboration to develop the network model used in OSCARS for the Interdomain Control Protocol (IDCP), which was described in [4].
- **eXtensible Session Protocol** This part of the Phoebus project engaged in a redesign of the previous session protocol and we called the new version the eXtensible Session Protocol (XSP). We documented the new version and intend to submit it for standardization. A prototype version of XSP was used in the Internet2 HOPI testbed.
- **Dynamic Firewalls** One of the interesting uses of XSP we identified and prototyped is the transparent control of firewalls. XSP can enable strong policy and authentication at session establishment (connect) time.
- **Network Processors** In order to keep pace with network speeds we worked toward an implementation of XSP that uses the Intel IXP network processor. This processor family can forward at 10Gb/sec and a working instance of XSP protocol on this platform would have enabled the adaptation mechanisms described above as well as others that we are investigating. We were not able to complete this implementation as it proved challenging to find students with the necessary skills.

## 5 Phoebus Results and Findings

Emerging dynamic networks<sup>1</sup> are unlike traditional networks because high-bandwidth virtual channels, or lambdas, are dynamically configurable. On the one hand, these networks allow extreme flexibility in dedicated resource allocation. On the other hand, this paradigm shift has profound effects on the way in which applications use networks.

Our approach was to provide a platform, in the form of protocols and middleware, which can provide a gateway to the emerging environment of agile, ultra-speed networks. The Phoebus architecture provides translation points for concatenating a series of network-appropriate resource reservation and transport layer connections. In cases where networks have radically different characteristics, having

---

<sup>1</sup>The UltraScience Net was a DOE funded dynamic network active during the initial Phoebus project.

an end-to-end transport protocol or control plane is quite problematic. In networks with dynamically allocated lambdas, for instance, the transport protocol may have a much different set of design goals than in networks with a shared uplink at the edge of the network. Using an eXtensible Session Protocol (XSP) library, a wide range of applications can be immediately enabled to use the new infrastructure and end-to-end performance can be improved. In addition, sites not directly connected to the dynamic backbone can take advantage of these advanced networks. Use cases such as this are described in [3].

Phoebus is a system that implements the XSP session layer protocol along with associated forwarding infrastructure for improving throughput in today's networks. Phoebus is a descendant of the Logistical Session Layer (LSL) [10], which used a more primitive protocol. Part of our work in this project was to design a general session protocol for use in dynamic networks.

The current Internet model binds all end-to-end communication to a transport layer protocol such as the Internet Protocol (IP) suite's Transmission Control Protocol (TCP). The Phoebus model binds end-to-end communication to a session protocol, specifically our XSP implementation. Thus, Phoebus is able to explicitly mitigate the heterogeneity in network environments by breaking the end-to-end connection into a series of connections, each spanning a different network segment. In this model, Phoebus Gateways (PGs) located at strategic locations in the network take responsibility for forwarding users' data to the next PG in the path, or to the destination host. Figure 1 demonstrates how PGs may be positioned at the border of regional networks.

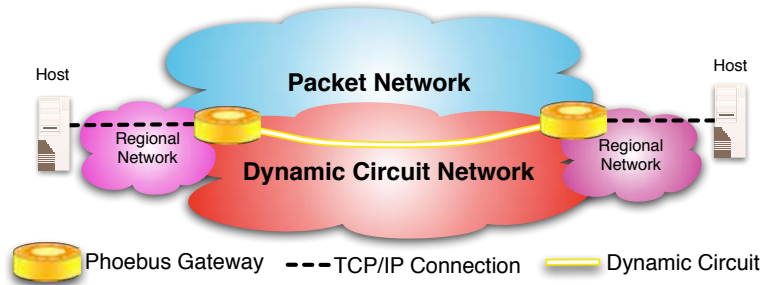


Figure 1: Phoebus Gateways at the border of regional networks.

The Phoebus network “inlay” allows data transfers to be adapted at application run time, based on available network resources and conditions. The Phoebus infrastructure creates an intelligent, articulated network. This network can take responsibility for ensuring good throughput for applications, while acting as an



adaptation point and “network on-ramp” to different network architectures via the *Path* framework provided by XSP. Phoebus bundles a variety of tuning and adaptation into a networked data movement service. One of the major benefits of such a system is the ability to offload network tuning to network administrators and the network itself, limiting the burden placed on a typical end-user.

The Phoebus architecture and its ability to adapt to different protocols between various network segments is described in detail in [6, 7]. In particular, we investigate a user space adaptation that provides benefits over dedicated WAN paths and discuss the scalability of Phoebus to 10Gb/s network speeds. These results were published in [6, 7].

The Phoebus architecture removes the direct binding between the application and transport layers. We refer to the intervening layer as a session layer. The session layer understands and negotiates segmented end-to-end connections. Essentially, the Phoebus Session Protocol handles hop-by-hop negotiation for an end-to-end path that may span multiple types of connections. Conceptually, the Phoebus approach separates the application from the transport interface. This paradigm is appropriate for a wide range of performance optimizations like parallel TCP streams or file transfer from distributed replicas. While we will discuss adapting applications with minimal changes, the semantics of this system are not the same as those of a transport layer. This is a network service that must be explicitly requested and therefore, doesn’t attempt to violate protocol layering or operation. Yet, by codifying the various techniques that are becoming common in application-level protocols, we do a service for the community and help advance the state of the art.

This approach allows us to take advantage of ultra-high speed networks while at the same time improving performance for legacy applications with minimal changes. Phoebus encapsulates link scheduling techniques along with an ability to drive core network configuration and allocation, and acts as a location to apply performance improving techniques such as ultra-large frames. Phoebus also provides a buffering point that serves to improve performance for legacy hosts while marshalling data for a burst over a temporarily dedicated lambda. This approach will contribute to our understanding of how to use ultra-speed networks, while at the same time providing a migration path that enables a wide range of applications to enjoy improved

The ESnet deployment and OSCARS integration are described in [9]. The dynamic circuit, buffer and burst implementation and results are described in [5]. A browser-based Java version of the system was presented in [1].

## 6 Other Products

Ezra Kissel completed his PhD thesis on Phoebus and XSP [8].

As part of the project’s outreach, we gave various talks. In addition to the talks at conferences presenting papers, there were invited talks [12, 13], talks at the Joint Techs and ESCC meetings [2, 16], Internet2 Member Meetings [11, 14] and to NITRD’s JET [15].

## References

- [1] Mnemonic: A network environment for automatic optimization and tuning of data movement over advanced networks. In *Workshop on Grid and P2P Systems and Applications (GridPeer 2009)*, in *Proceedings of 18th International Conference on Computer Communications and Networks (ICCCN 2009)*, August 2009.
- [2] G. Almes, A. Brown, and M. Swany. Achieving dependable bulk throughput in a hybrid network. Presentation at Joint Techs Int’l Conference of Network Engineers, July 2006.
- [3] A. Brown, E. Kissel, M. Swany, and G. Almes. Phoebus: A session protocol for dynamic and heterogeneous networks. UDCIS Technical Report 2008:334, [http://damsl.cis.udel.edu/projects/phoebus/phoebus\\_tech\\_report.pdf](http://damsl.cis.udel.edu/projects/phoebus/phoebus_tech_report.pdf).
- [4] A. Brown, M. Swany, and J. Zurawski. A general encoding framework for representing network measurement and topology data. *Concurrency and Computation: Practice and Experience*, 21(8):1069–1086, June 2009.
- [5] E. Kissel and M. Swany. Session layer burst switching for high performance data movement. In *Proceedings of PFLDNet*, 2010.
- [6] E. Kissel, M. Swany, and A. Brown. Improving gridftp performance using the phoebus session layer. In *SC ’09: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, pages 1–10, New York, NY, USA, 2009. ACM.
- [7] E. Kissel, M. Swany, and A. Brown. Phoebus: A system for high throughput data movement. *Journal of Parallel and Distributed Computing*, 71:266–279, February 2011.

- [8] E. D. Kissel. *IMPROVING WIDE-AREA NETWORK PERFORMANCE WITH THE EXTENSIBLE SESSION PROTOCOL: A PROTOCOL FOR FUTURE INTERNET ARCHITECTURES*. PhD thesis, University of Delaware, May 2012.
- [9] L. Ramakrishnan, C. Guok, K. Jackson, E. Kissel, D. M. Swany, and D. Agarwal. On-demand overlay networks for large scientific data transfers. *IEEE International Symposium Cluster Computing and the Grid (CCGRID)*, 0:359–367, 2010.
- [10] M. Swany. Improving Throughput for Grid Applications with Network Logistics. In *Supercomputing 2004*, November 2004.
- [11] M. Swany. Phoebus: High-performance data transfer for hybrid optical / packet networks. Internet2 Member Meeting, December 2006.
- [12] M. Swany. Phoebus: Next-generation network middleware for distributed computing. Invited Presentation at HPCAsia’07, September 2007.
- [13] M. Swany. Network performance for clusters and grids. Keynote Speaker at Mid-Southeast ACM Conference, November 2008.
- [14] M. Swany. Phoebus: Case studies for dynamic circuit networks. Internet2 Member Meeting, April 2008.
- [15] M. Swany. Phoebus: Network middleware for high-performance networking. Invited Presentation to NCO NITRD Large Scale Networking JET Meeting, January 2011.
- [16] M. Swany and E. Kissel. Phoebus: Network middleware for high-performance data transfer. Presentation at Joint Techs International Conference of Network Engineers, July 2008.