Policy-Based Resource Management and Service Provisioning in GMPLS Networks*

Xi Yang[1], Tom Lehman[1], Chris Tracy[2], Jerry Sobieski[2], Shujia Gong[3], Payam Torab[3], Bijan Jabbari[3]
[1]Information Sciences Institute East, University of Southern California
3811 Fairfax Drive, Suite 200, Arlington, VA 22203; Tel: (703) 812-3736; Email: {xyang, tlehman}@isi.edu
[2] Mid-Atlantic Crossroads, University of Maryland
8400 Baltimore Ave, Suite 102, College Park, MD 20740; Tel: (301) 314-6662; Email: {chris, jerrys}@maxgigapop.net
[3]Electrical and Computer Engineering Department, George Mason University
4400 University Drive, Fairfax, VA 22030-4444; Tel: (703) 993-1618; Email: {sgong, ptorab, bjabbari}@gmu.edu

Abstract* — Emerging network applications tend to be built over heterogeneous network resources spanning multiple management domains. Many such applications have dynamic demands for dedicated, deterministic, high-bandwidth connections. The Generalized Multi-Protocol Label Switching (GMPLS) networks under development can address these kinds of demands by using policy-based resource management and service provisioning technologies. In this paper, we present the architecture and implementation for policy-based resource management and service provisioning as part of the work on the NSF funded Dynamic Resource Allocation via GMPLS Optical Networks (DRAGON) Project. This work captures several critical features of service-oriented GMPLS networks, including a) collaborative interdomain resource management; b) interdomain end-to-end path computation; c) advance scheduled provisioning; and d) Authentication, Authorization and Accounting (AAA). These features rely on the capability of exchanging and coordinating resource and policy information among multiple participating domains. In pursuit of this capability, we propose a three-dimensional (3D) Resource Computation Model (RCM). The three broad dimensions of this model are Traffic Engineering (TE) constraints, time schedule constraints, and AAA policy constraints. The 3D RCM facilitates policy based resource allocation based on many specific constraints within these three broad categories. Based on this model we describe our approach to GMPLS interdomain end-to-end path computation, advance scheduled provisioning, and AAA policy based provisioning, respectively. Our current DRAGON implementation status is also reported in this paper.

***Keywords* —** *Generalized Multi-Protocol Label Switching (GMPLS); Optical networks; Service provisioning; Policy based management; Interdomain path computation; Dynamic Resource Allocation via GMPLS Optical Networks (DRAGON).*

I. INTRODUCTION

There is an increasing number of high performance applications and users who require network services beyond what are typically available on best effort infrastructures today. These super users require what have been referred to as "deterministic" network services. In this context, "deterministic" implies defined and guaranteed service level. These service level parameters include bandwidth as a minimum. An ability to specify loss rates, latency, jitter, and other parameters is also envisioned. In addition, these services need to be provisioned on an interdomain basis, across heterogeneous network technologies, and include features for Authentication, Authorization, Accounting (AAA) [1] and scheduling.

The continued evolution of optical network technologies combined with dynamic provisioning mechanisms such as Generalized Multi-Protocol Label Switching (GMPLS) holds the promise to enable these types of advanced services. GMPLS is addressing the Traffic Engineering (TE) constraints as applied to a heterogeneous network environment and includes issues associated with routing, path computation, and signaling. This is a very complicated space, and there are many unresolved issues, particularly in the areas of interdomain and multiregion topologies. Another important topic that is missing from the current GMPLS development activities is the notion of "policy based provisioning" [2][3]. This implies the application of additional constraints to resource provisioning and allocation decisions. The new constraint dimensions we refer to are AAA and scheduling. This forms the basis of what we refer to as a three-dimensional (3D) Resource Computation Model (RCM). The three broad dimensions of this model are TE constraints, AAA policy constraints, and time schedule constraints. This model introduces the notion of a 3D Resource Computation Element (3D RCE) which includes a 3D Traffic Engineering Database (3D TEDB) and a 3D Path Computation Engine (3D PCEN). These components are used to reduce complex policy information to a simple policy directive which enables Label Switch Routers (LSR's) to process provisioning requests rapidly.

---

*This work is supported by National Science Foundation under Grants Numbers 0335300, 0335266 and 0335230.

The material presented in this paper is based on work underway as part of the DRAGON (Dynamic Resource Allocation via GMPLS Optical Networks) project which is funded by the National Science Foundation (NSF). DRAGON is collaborative project between University of Maryland (UMD) Mid-Atlantic Crossroads (MAX), University of Southern California Information Sciences Institute East (USC/ISI), and George Mason University (GMU). The DRAGON project is developing technology and deploying network infrastructure to allow dynamic provisioning of deterministic network paths in direct response to end-user requests. The initial motivation for these advanced network services is in support of the eScience community who desire the dynamic acquisition of dedicated and deterministic network resources to link expensive equipment such as radio telescopes, computational clusters, storage arrays, visualization facilities, remote sensors, and other instruments into globally distributed and application specific topologies. Other communities and applications whose interest is anticipated include emergency response, mission/business critical services, and building (or traffic engineering) of a best effort IP network. We present here an architecture for policy based resource management and service provisioning in GMPLS networks which is motivated by our work on the DRAGON project. This includes descriptions of our approach for interdomain routing, end-to-end path computation, and signaling; advance schedule provisioning; and AAA policy based provisioning. In addition, a reference implementation of this architecture is being constructed in the Washington D.C. area. The current implementation status is reported in this paper.

The remaining paper is organized as follows: in Section II we outline an architecture for policy based resource management and service provisioning. In Section III we discuss our 3D RCM. In Section IV we discuss interdomain path computation. In Section V we discuss AAA policy based provisioning. In Section VI we discuss schedule constraints as applied to resource computation. In Section VII we discuss our implementation status. In Section VIII we present our conclusions.

II. AN ARCHITECTURE FOR POLICY BASED RESOURCE MANAGEMENT AND SERVICE PROVISIONING

Development of an architecture which allows multi-domain policy based service provisioning involves many issues and tradeoffs which must be balanced against the objectives as defined by the users of the system. For us, these objectives were initially driven by our primary project applications which are expensive resource eScience applications and the building (and traffic engineering) of IP networks. At a very high level we identified several requirements or architectural objectives to assist us in the formulation of an architecture. These are as follows:

- The infrastructure should allow provisioning actions to be requested by a user. This should be in the form of a "network service" where upon receipt of a request, the network responds in a deterministic manner with respect to its ability to satisfy the request.
- The primary unit of request is a connection (Label Switch Path (LSP) in GMPLS nomenclature). Individual LSPs can be combined into Application Specific Topologies (AST's) as a higher level service on top of the provisioning infrastructure.
- Users should be able to specify certain parameters associated with the requested LSP. As a minimum this should include endpoints, bandwidth, time (duration). Additional parameters such as latency, packet loss guarantees and jitter are desired, but considered research topics at this time. These parameters should be defined in the form of a set of "common services" which are well understood and easily measured.
- Service provisioning should be possible on an interdomain basis, across heterogeneous network topologies, and include features for AAA and scheduling.
- Provisioning times on the order of seconds to possibly tens of seconds (for complicated interdomain topologies) are acceptable for the targeted class of applications.
- As an architectural goal it is desired that once service provisioning begins, there be no requirement for signaling messages to be processed in out of band elements (such as offline path computation or policy server elements).
- This architecture should maximize the ability to utilize vendor GMPLS implementations and require few if any changes to realize the goals above. This should include mechanisms to utilize features now available in vendor implementation, and also be able to adapt to use future vendor features as they become available.

The architecture we developed is depicted in Figure 1. We have identified the two primary categories of interdomain data flow as call control and connection control. The connection control phase is responsible for interdomain routing, path computation, and signaling. This architecture is built on the foundation of GMPLS, and the interdomain call control architecture is an extension of what has been defined for intradomain GMPLS. While some of these interdomain issues are under discussion within the various standards bodies, there are many open and unresolved issues. We develop solutions for several of these and they are presented in this paper. Of particular note are the difficulties in dealing with heterogeneous network technologies which requires development of multiregion path computation and signaling techniques, and also development of scalable distribution of traffic engineering data in a multi-domain environment.

The call control phase is responsible for establishing relationships between end users/systems; distribution of user information; AAA data exchange; and scheduling information exchange. Some of these data may be user specific while some will be resource or domain specific. The specific details of the call control phase of this architecture are still in an early

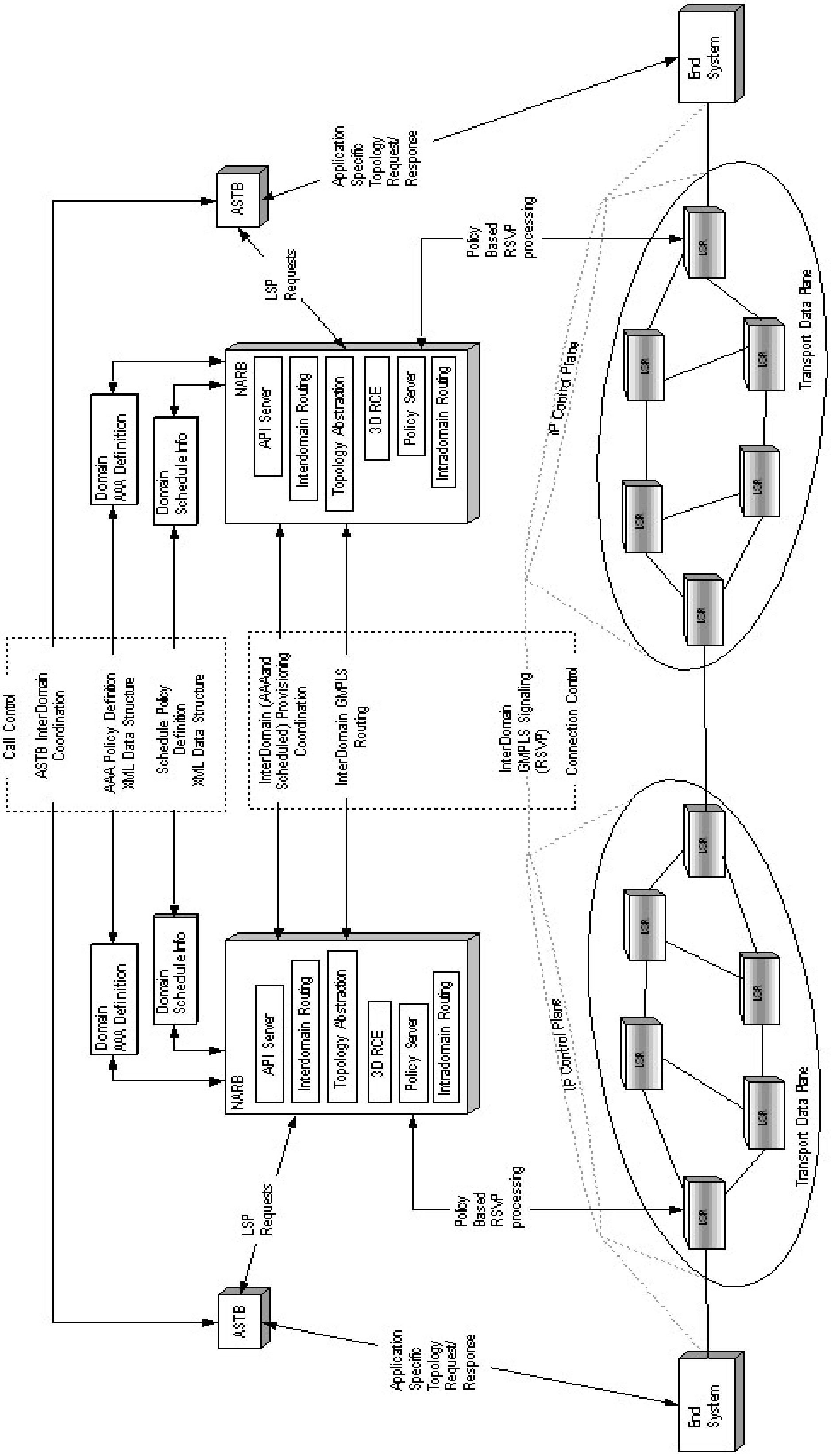

Figure 1: Diagram of the DRAGON architecture for policy based resource management and service provisioning in GMPLS networks.

phase of development and various options are under consideration. This includes extensions of some of the Grid Networking concepts for application to service provisioning in GMPLS networks [4].

An important item to note about this architecture is that while it is convenient to utilize the call and connection control categories, significant interaction between the two will be required. This is depicted by the information flow seen between components primarily dedicated to one phase or the other. A key architectural objective is to minimize the impact that changes in one category will have on the other. In other words, connection control relies on the standard GMPLS control plane while call control provides a common service architecture that utilizes the intelligence of connection control but is independent of any GMPLS specific implementation.

A. Architecture Components

Network Aware Resource Broker (NARB). Routing, path computation, and signaling on an InterDomain basis across topologies which include a heterogeneous mix of network technologies and vendor equipment is beyond what is defined in standards and also beyond the capability of current vendor equipment. To enable routing, path computation, and signaling in this environment the NARB provides several important functions. The NARB is an agent which represents a local Autonomous Domain (AD) and acts as a protocol listener to the intradomain routing protocols. In our implementation, the intradomain protocol is OSPF-TE [5]. The NARB is also responsible for inter-domain routing. NARB's peer across domains and exchange topology information to enable inter-domain path computation and Label Switched Path (LSP) provisioning. The NARB's utilize a modified version of OSPF-TE to share a link state database between domains. This inter-domain topology exchange can be based on the actual topology as discovered by listening to the local OSPF-TE protocol, or optionally based on an "abstracted" view of the domain topology (generated by configuration file or automatic synthesis of the OSPF-TE link state database). Domain abstraction provides mechanisms for an administrative domain to advertise to the outside world a highly simplified view of its topology. This allows domains to hide their real topologies as well as minimize the amount of external updates required. The trade-off is reduced accuracy for path computations. Each administrative domain can utilize configuration parameters to tailor its domain abstraction to the level desired. One of the goals of our project is to evaluate various architectural issues. The domain abstraction features of NARB are geared toward allowing experimentation with differing levels of topology hiding. The resulting interdomain architecture would most accurately be described as a hybrid between the peer-to-peer and overlay models. The NARB currently holds the interdomain link state topology and does not advertise that data within its own domain. There are several reasons for this including the general inability for current GMPLS implementations to utilize such data as part of their CSPF calculations. Future configurations may leak some or all of this topology into local domain routing.

The NARB also includes advanced algorithms which allow path computation with multiple constraints. These constraints include the standard GMPLS TE parameters as well as AAA, scheduling, multi-region switching capabilities, and vendor specific limitations such as switching capability adaptation abilities. These features are part of the 3D RCE component which is described in more detail in a subsequent section of this paper. The output of the path computation is an Explicit Route Object (ERO) which may be a set of strict hops, loose hops, or a mixture of both. In addition, the NARB includes functionality to determine if multi-region and multi-domain techniques such as LSP nesting or stitching may be required. The goal is that once the NARB path computation is complete, signaling can progress with no need to return to an out of band path computation element.

The NARB also plays an important role in distribution of policy information to LSR's so that appropriate action can be taken when processing provisioning messages. To accomplish this, the NARB translates the complex AAA and schedule information located in the 3D RCE into a simple policy directive which is distributed to the appropriate LSR's. Additional details on this are provided in a subsequent section.

Application Specific Topology Builder (ASTB). The DRAGON architecture includes the notion of establishing Application Specific Topologies (AST). These are requested by an end user and are generally a set of LSP's which an application domain desires to be set up as a group. The DRAGON element known as the Application Specific Topology Builder (ASTB) is responsible for coordinating this in response to application requests. The ASTB subsequently coordinates with the NARB which views these as individual LSP's.

End System Agent (ESA). The ESA is software that runs on (or on behalf of) the end-system which terminates the data plane (traffic engineering) link of the provisioned service. This is the software that participates in the GMPLS protocols to allow for on-demand end-to-end provisioning from end-system to end-system. The ESA typically runs in peer-to-peer mode or overlay mode via a UNI protocol [6]. The ESA may also interact with the ASTB if a more complicated topology is to be built.

Virtual Label Switch Router (VLSR). An important objective for the DRAGON architecture is to be able to provision across heterogeneous network technologies and vendor equipment. For vendor equipment which is not GMPLS capable the concept of the Virtual Label Switch Router (VLSR) is introduced. This is a control plane stack which includes OSPF-TE and RSVP-TE [7] and acts as a proxy agent for non-GMPLS capable devices. This allows non-GMPLS devices to be included in end-to-end path instantiations. The primary use for VLSR on the DRAGON project is to control ethernet switches via the GMPLS control plane. However, the VLSR has also been adapted to control TDM and Optical switches. While a VLSR is not identified directly in the architectural diagram, any of the LSR's identified could in fact be a VLSR.

B. *Provisioning Data and Message Flow*

There are several ways to effect an actual provisioning action. This is due to the fact that GMPLS offers several modes of operation and also because our architectural components can be configured in various modes. However, an example sequence of steps to invoke an interdomain policy based service provisioning can be described as follows (referencing components and data flows in Figure 1). It should be noted that is a simplified sequence of events intended to present an overview of the process. Additional details regarding these actions are described in subsequent sections.

- *Prior to Provisioning Request*

- AAA and Schedule Policy Exchange. It is expected that policy information will be exchanged well in advance of actual provisioning time. In particular, AAA and schedule policy definitions are not expected to be extremely dynamic. The typical scenario is that this information exchange and associated policies will be in place well in advance of the connection control phase.

- NARB Processing of Call Control Data. A key component of the 3D RCM is the integration of AAA and scheduling information into the 3D RCE, 3D TEDB, and Policy server. With the expectation that this policy information will be distributed well in advance of provisioning time, this integration should happen well in advance as well.

- Interdomain Routing. This is an ongoing exchange of data in the form of modified link state protocol. The exchange may be based on actual topologies or abstracted topologies.

- *At Provisioning Time*

- ESA Request. When a provisioning action is desired, the ESA requests a network path from the ASTB or optionally it can make this request directly to the NARB if it is for a single LSP.

- NARB Response. The NARB will evaluate the request based on policy, consult 3D RCE and return a result. This process can actually be quite complex and may require multiple communications with NARB's in other domains in some instances. The more complex scenarios are described in subsequent sections. If the requested path is possible and permitted, the result is returned in the form of an Explicit Route Object (ERO). The NARB may also be required to load a policy directive in LSR's if this was not possible to be done in advance.

- Signaling. Based on the received ERO, the ESA can initiate signaling via use of RSVP-TE or GMPLS UNI.

III. 3D RESOURCE COMPUTATION MODEL

A. *Model Description*

End-to-end service provisioning in a multi-domain GMPLS network involves three phases of control processes. In the first phase, resource and policy information, such as network link states, reserved resource time slots and AAA policies, is exchanged between control-plane entities in both intradomain and interdomain scopes. In the second phase, the resource and policy information is used in a coordinated manner to determine which, when and how network resources should be allocated. We call this a resource computation phase. The main control process in this phase is GMPLS routing or path computation. In the third phase, results or decisions from the resource computation phase are rendered into actual resource allocation and service provisioning. GMPLS signaling is a typical control process in the third phase.

Resource computation is the key phase that transforms resource and policy information into policy based GMPLS routing and signaling decisions. In this section, we present a three-dimensional (3D) constrained Resource Computation Model (RCM) that serves as the key to incorporate policy constraints into GMPLS routing computation. 3D refers to the three kinds of resource and policy information in policy based GMPLS networks, including resource states, time schedule and AAA policy rules. They correspond to the three dimensions of constraints on resource allocation, i.e., traffic engineering (TE) constraints, time schedule constraints and AAA policy constraints, respectively.

Figure 2 shows the 3D RCM. The three dimensional constraints together compose a solution space for each individual LSP request. For example, the requestor's privileges and certain restrictions on access time dictate a unique solution space for an LSP request. Searching the solution space with the criteria provided by the LSP request, such as source, destination and bandwidth, etc., will result in a feasible solution or a failure. A feasible solution indicates the resources to be allocated for the requested LSP as well as the LSP uptime and duration.

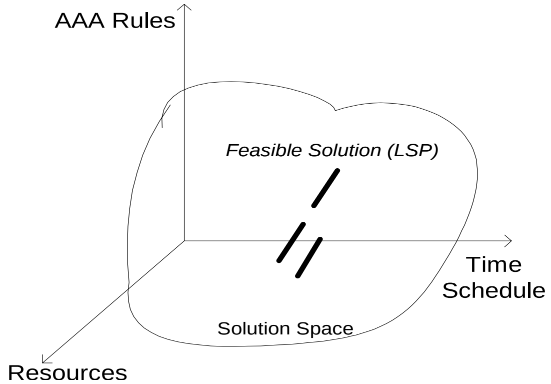

Figure 2: 3D constrained Resource Computation Model (RCM).

B. *Resource Computation Engine (RCE)*

Due to the complexity of resource computation, it is desirable to have all the three dimensions of resource and policy information available in a single computation space, which we refer to as Resource Computation Engine (RCE) in the DRAGON architecture. RCE has the functionalities of both a resource information base, including the TEDB, and a PCEN. The DRAGON PCEN features are similar or comparable to those described in the IETF Path Computation Element Architecture [8]. By eliminating the need for repeated information retrieval from other network elements, path computation cannot only run fast but also avoid some inconsistent resource states. A dedicated, deterministic LSP can be obtained from RCE by a single request. Additional routing constraints carried in the LSP request, e.g., some Service Level Agreement (SLA) requirements and user specified restrictions, are also incorporated into each path computation.

Figure 3 shows the diagrammatic RCE architecture. RCE supports a variety of protocol API's in order to collect resource information from other control plane entities. For instance, a specific OSPF API allows RCE to collect OSPF link state information from an OSPF daemon that supports the API. Resource information, such as OSPF Link State Advertisement (LSA), is parsed and stored into TEDB. In addition, local and global AAA rules and LSP schedule information can be exchanged through the RCE API. By cross-referencing to the constraints in the AAA Rules Table and the LSP Schedule Table, the TEDB becomes a 3D TEDB, which provides the input for the 3D CSPF PCEN module to perform 3D path computation. In addition to 3D path computation, RCE provides some computation functions for resource management, LSP scheduling and policy management.

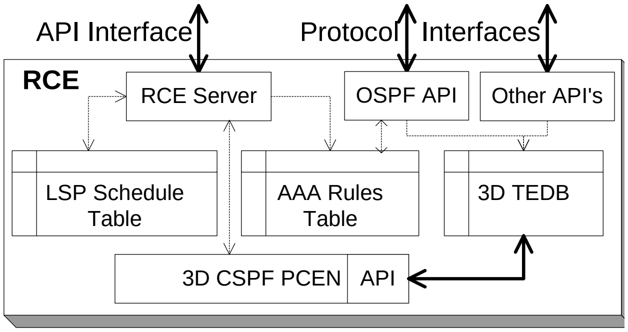

Figure 3: Resource Computation Engine (RCE) architecture.

C. 3D Constrained Path Computation

The diagram in Figure 4 shows our 3D constrained path computation process. Upon the LSP request, RCE checks out all the AAA rules related to this request from the AAA rules table. These AAA rules, plus some user-specified rules carried by the LSP request, are parsed into constraints that instruct RCE to only retrieve the related resources from the TEDB and create a copy of the retrieved resource information in the memory.

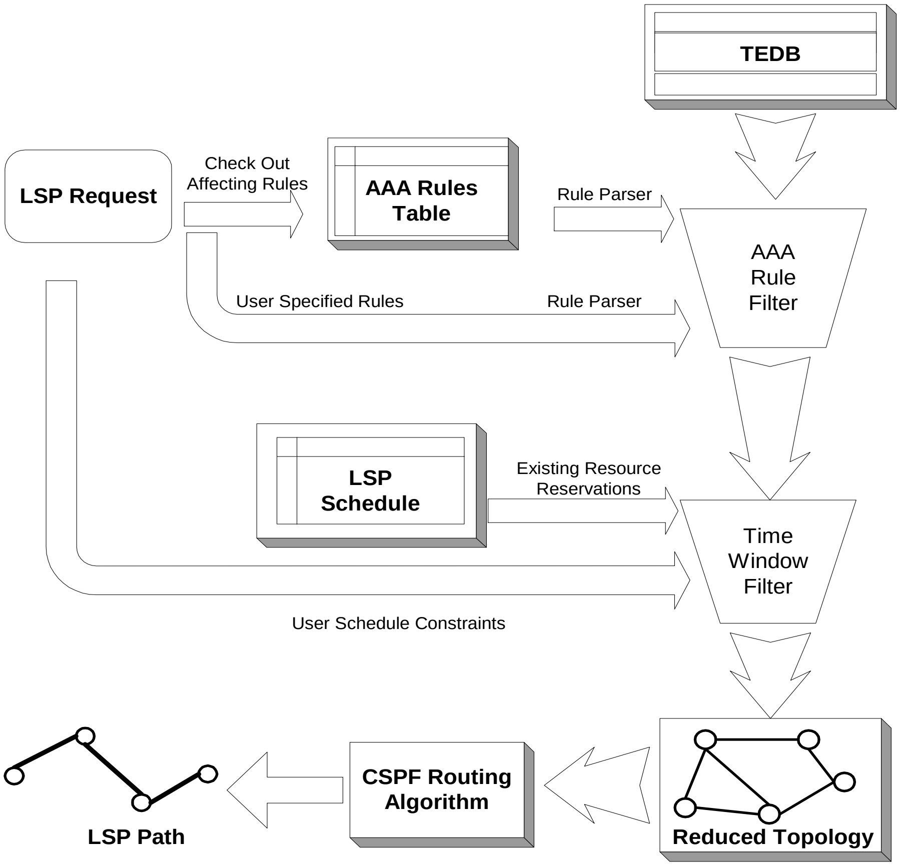

Figure 4: 3D constrained path computation process.

As resource availability is measured by time slots, the retrieved resource information is further constrained by the existing LSP schedule and therefore those time slots already reserved shall be removed. Next, to satisfy the user specified time schedule in the LSP request, RCE filters the retrieved resources using the time schedule constraints as requested. The resulting resources are used to create a reduced network topology. CSPF computation is carried out on this reduced topology to select a routing path. If a fixed time schedule is requested, time filtering and CSPF computation are carried out only once. If a flexible schedule in a given a time window is requested, RCE needs to scan the resources in the whole time window for W times, where W is the window size counted by the number of time slots in that time window. After each scanning, RCE increments the LSP uptime by one time slot, creates an instance of topology and carries out the CSPF computation. This procedure is repeated up to W times until a path is found. Note that this process could be very time-consuming when W is big, say over 100. A more efficient heuristic will be developed in future implementation.

D. *Operational Description*

As discussed above, the 3D RCM can effectively incorporate the policy constraints, such as time schedule and AAA rules, into path computation. This model is fully supported by RCE. With RCE, the GMPLS control plane can handily provide the below features:

1) Interdomain end-to-end (E2E) path computation.

   RCE collects all the local and global resource and policy information known to its domain. It understands and uses these information as a whole under the 3D RCM and thus provides a information-rich global network view. In either centralized or per-domain path computation, RCE can respond to each interdomain E2E LSP request with a deterministic answer.

2) Advance scheduled service provisioning.

   RCE takes LSP schedule information into its path computation. In addition, RCE maintains an LSP schedule table and updates resource reservation states dynamically so that the path computation results can always be reflected in resource states and be disseminated to other domains. Therefore, any user requesting an advance scheduled service can be provided instantly with a deterministic answer. Upon successful advance reservation, the reserved time slots of related resources will be dedicated to that service and will not be allocated to any other services under the 3D RCM.

3) AAA policy based provisioning and admission control.

   RCE learns local or global AAA policy rules dynamically and takes the latest AAA policy constraints into path computation. The 3D constrained resource computation model guarantees that no AAA rules should be violated during resource allocation. In addition to individual path computation, this model can be used by RCE to compute access lists in support of policy based admission control.

Design and implementation details of these features will be presented in the subsequent sections.

IV. INTERDOMAIN END-TO-END PATH COMPUTATION

A. *Interdomain Routing Architecture*

In multi-domain GMPLS networks, the first architectural challenge for end-to-end path computation and traffic engineering is to exchange resource and policy information across domains. In particular, some traffic engineering attributes, such as link bandwidth information and switching capability description, must be disseminated beyond their local domain. A link-state routing information exchange mechanism, e.g., OSPF flooding, is a natural fit for this purpose. By coordinating with RCE and signaling modules, a flooding based link-state routing protocol allows distributed network entities to update resource information dynamically and continuously. However, flooding full link states in the interdomain scope causes high communication overhead and processing latency in the control plane, resulting in poor scalability. To solve this problem, we designed a two-hierarchy interdomain routing architecture as illustrated in Figure 5 and described below.

In our DRAGON interdomain routing architecture, the Network Aware Resource Broker (NARB, see Section 2) in each domain summarizes its intradomain traffic engineering information and advertises an abstract topology to other domains. An instance of link-state routing protocol is responsible for flooding the abstract topology information at the interdomain level. RCE in each domain can therefore construct a global topology with summarized traffic engineering information. Meanwhile, one instance of link-state routing protocol, say OSPF-TE, runs inside each domain, responsible for flooding physical traffic engineering information in an intradomain scope, which allows RCE to construct a local domain topology with detailed traffic engineering information. In our design, global LSP schedule information is exchanged as resource reservation TE attributes by the interdomain link state routing protocol while global AAA policy information is exchanged via NARB-to-NARB communication. This architecture provides an effective interdomain routing information exchange mechanism with reasonable scalability.

B. *Interdomain E2E Path Computation Schemes*

The DRAGON architecture supports three interdomain E2E path computation schemes. The first scheme uses a centralized path computation scenario, in which the source-domain RCE computes a complete end-to-end routing path for each LSP request. The other two schemes use a per-domain path computation scenario, which involves the RCE’s of all the domains along the routing path. While we dedicate these schemes to the DRAGON environment, we tend to design them in compliance with or being comparable to those proposed in the IETF drafts such as [9], [10].

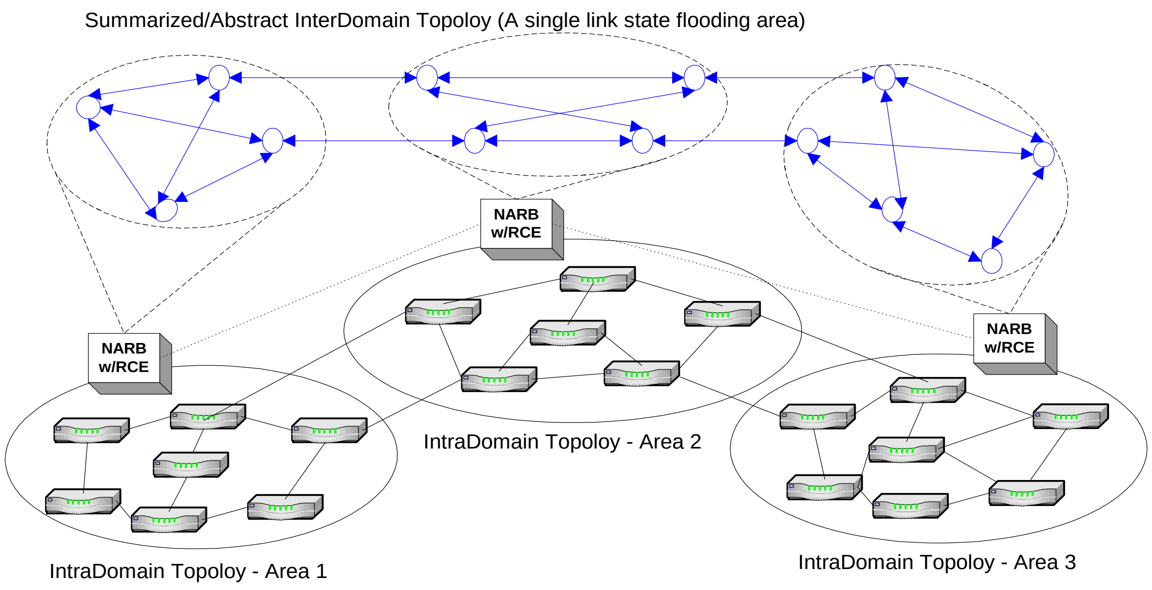

Figure 5: Illustration of the DRAGON interdomain routing architecture.

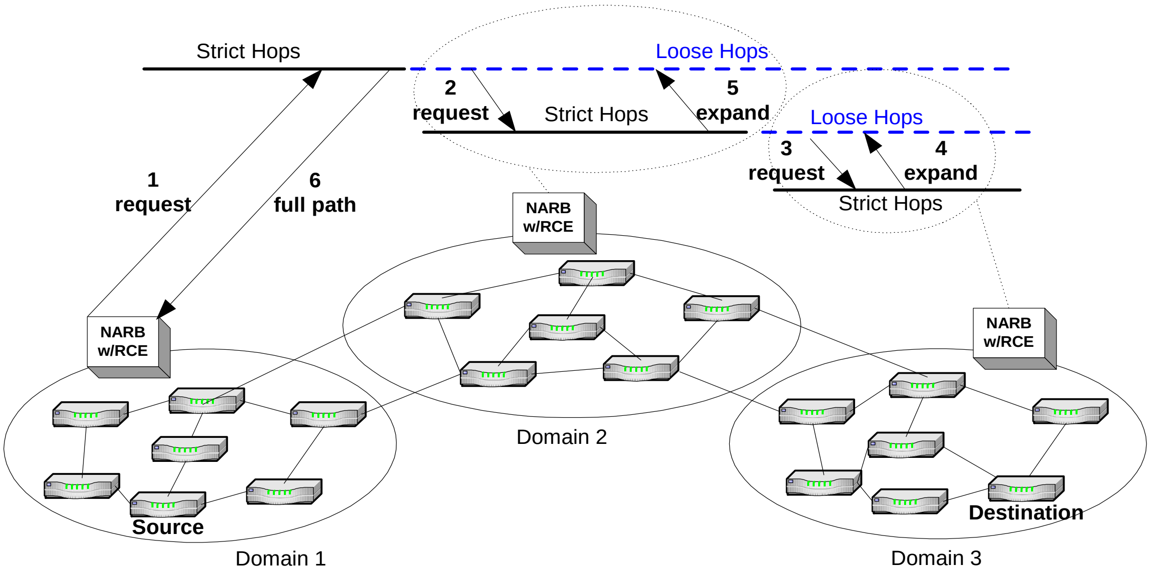

Figure 6: Illustration of the recursive per-domain path computation scheme.

- *Centralized path computation scheme*

RCE can synchronize its TEDB to the link state databases (LSDB) of both the interdomain and intradomain link state routing protocol daemons. Thus, RCE always maintains the latest view of a summarized global TE topology and a detailed local-domain TE topology. Upon an LSP request, RCE can return an ERO with strict hops leading to an egress border router in the local domain, followed by a sequence of loose hops leading all the way to the destination. When a complete routing path, i.e., an ERO with all strict hops, is desired, RCE firstly requests for detailed resource information from relevant domains. RCE finds the relevant domains by using the summarized global TE topology. By incorporating the additional resource information into its TEDB, RCE will be able to expand each loose hop into strict hop(s) and returns an all-strict-hop ERO.

- *Recursive per-domain path computation scheme*

Instead of retrieving additional resource/topology information from other domains, the recursive per-domain (RPD) path computation scheme asks the RCE of the next domain to expand the remaining loose hops on the pre-computed routing path. The expanded strict hops are appended to the strict hops in the current domain to compose an all-strict-hop ERO. In the actual process, which is illustrated in Figure 6, the last domain is the first to obtain an all-strict-hop segment and the domain before the last is the second, which appends the strict hops in the last domain to the strict hops in its own domain to obtain an all-strict-hop segment. This expansion procedure is carried out recursively until a complete path with all strict hops from source to destination is obtained. The RPD scheme only needs NARB-to-NARB communication and is therefore independent of the intermediate routers.

- *Forward (or Fast) per-domain path computation scheme*

The forward per-domain (FPD) path computation scheme is an alternate way for multiple RCE's to collaborate in end-to-end path computation. Instead of requesting for expanded routing segment from next domain in a recursive manner, the FPD scheme simply requires the current domain to forward the LSP request with the known strict-hop segment to the border router of the next domain. The border router of the next domain will append the strict hops in its own domain by consulting with its associated RCE and continue to forward the LSP request until reaching the destination. When used with a signaling process, say that FPD path computation is triggered by an RSVP path message, this scheme can help create an LSP very quickly. It can also be used when some domains have no RCE support. The FPD scheme requires that an intermediate border router be either RCE-aware or capable of expanding loose routing hops on its own.

C. Interdomain E2E Service Provisioning Process

In this section, we describe a complete interdomain E2E service provisioning process. We focus on the GMPLS traffic engineering side and leave the advance scheduling and AAA policy based issues to later sections. We describe the process in the following three control sequences.

- *Request preprocess sequence*

An E2E service may request a single LSP or a topology consisting of multiple LSP's. In the latter case, the topology request is decomposed into separate LSP requests at ASTB. Only path computation returns a path successfully for each and every LSP, will the actual resource allocation (signaling) process be kicked off.

The format of an LSP request is shown below:

LSP-REQ = {*Type, Source IP/Port, Destination IP/Port, LSP Bandwidth, EncType, SwType, tWinOpen, tWinClose, tDuration, User ID, <User Profile>, <SLA Parameters>*}

There are four types of LSP requests. They are:

A. LSP with both source and destination in the current domain;

B. LSP with only source in the current domain;

C. LSP with only destination in the current domain; and

D. LSP with both source and destination in foreign domains.

The requests of types A and B are sent to the local NARB server, which proceeds to the LSP query sequence. The requests of types C and D are sent to corresponding foreign NARB servers through NARB-to-NARB communication. A home identifier tag is added to those requests. Upon receiving an LSP request, a foreign NARB server will proceed to the LSP query sequence. When the query is done, the foreign NARB server will return the results back to the home NARB server.

- *LSP query sequence*

Upon receiving an LSP request, the NARB server directs it to the LSP provisioning module. The LSP provisioning module firstly translates time schedule attributes, user profile and SLA parameters into routing constraints understandable for RCE. Then a path request with the translated constraints is passed to the RCE for path computation. RCE will use one of the schemes in section 4.2 to compute an end-to-end routing path for the LSP request.

When a desired LSP path is obtained, the LSP provisioning module sends resource-freezing requests to the resource management module in NARB to request freezing all the resources to be allocated to the LSP. The purpose of the freezing procedure is to avoid inconsistent resource allocation between simultaneous LSP request sessions. To freeze the requested resources, states of related entries in the TEDB (i.e., time slots for resources) are changed into *frozen*, with a tag to indicate which request session has frozen it. If a resource has already been frozen by another request session, a contention resolution procedure is invoked to decide who should get this resource. If a resource is in a foreign domain, the resource management module should forward the request to its foreign peer, asking for freezing that resource.

Only after the resource management module acknowledges that all the requested resources are frozen, can the LSP provisioning module acknowledge the requestor with a *success*. Otherwise, a *failure* is returned and all frozen resources are unfrozen. When the resource freezing request runs out of time, a *failure* should also be returned.

- *LSP setup sequence*

After all the LSP requests are responded with *success*, the LSP setup sequence is executed. In this sequence the states of those *frozen* resources will be changed into *reserved*. The LSP provisioning sequence involves the interaction with End System Agents (ESA, see section 2) and the signaling process. In this paper, we skip this part of description.

V. AAA POLICY BASED PROVISIONING

The objective of the policy based provisioning is to incorporate AAA policy into path computation, resource allocation, and signaling functions. This requires high level associations of policy with users (or groups of users) as well as lower level associations of policy with actual network elements at a fidelity sufficient to implement meaningful policy based resource allocations. These two levels were loosely described as call control and connection control in the earlier architecture description. For the higher level AAA functions, there is related work underway or completed in several communities. From the GRID community this includes the Open Services Grid Architecture (OSGA) [11] developed by the Global Grid Forum (GGF) and the associated Global Security Infrastructure (GSI) [12] developed by the Globus consortium. From the Internet2 research community, Shiboleth [13] is an architecture and product which provides similar AAA functions. While these have been generally geared toward grid applications, they are now being reevaluated for application to network provisioning functions. Other related work from the IETF, includes working drafts on integration of AAA functions into the Session Initiation Protocol (SIP) [14]. These efforts are addressing many of the generic issues associated with AAA and include mechanisms to aggregate policies, develop virtual organizations, and include the use of certificate based and/or ticket based authentication mechanisms.

What is missing from much of this work is an extensible and scalable architecture to translate these higher level associations to the network provisioning level. As a result, the focus of our efforts is on the development of architectures and mechanisms to translate the high level AAA information into information which can be utilized directly in the provisioning process. Our approach is the 3D RCM as described earlier in this paper. In this model we synthesis the higher level AAA information into policy information which is associated at the TE resource level. We also utilize the higher level AAA information to develop a set of policy rules. The TE policy data and the policy rules are used during path computation to incorporate AAA policy into provisioning operations. In so doing, AAA policy decision can be synergized with TE based provisioning decision, resulting in fast, precise and simplified control process.

As a research project, we are interested in experimenting with multiple architectures and techniques for AAA policy based provisioning. We intend to evaluate multiple higher level AAA methods including GSI, Shiboleth, and SIP based solutions. As a result, we have attempted to incorporate AAA information into our 3D RCE in manner which is flexible enough to adapt to changes in the higher level AAA architecture.

A description of how we incorporate AAA information into our 3D RCM is provided below.

A. 3D RCM AAA Incorporation for Policy Decision

As described above, there is an information flow which results from the high level AAA architecture. This may be one of several implementations, and the specifics of any one are not presented in this paper. The approach for the 3D RCM is to translate the higher level AAA information into data that can be incorporated into the 3D TEDB. At the global/interdomain topology level, this is inserted into the 3D TEDB in the form of new TLVs which are attached to specific TE resources such as LSR's and TE Links. This higher level AAA information is utilized to develop a set of policy rules. The application of the policy rules against the AAA TLVs allows policy based path computation across domains. At the local/intradomain topology level, AAA policy information can be inserted into 3D TEDB directly by other local modules and be translated into intradomain path computation constraints by 3D RCM.

At the global topology level, the AAA TLV is included as a sub-TLV in the TE-Link Opaque LSA. We are still working on development of a flexible TLV format which can adapt to changes in overall AAA architectures. However, the current TLV format we have defined is shown in Figure 7. Each AAA TLV consists of one or more sub-level AAA TLV's. In this design, we try to avoid an oversized AAA TLV. A sub-level AAA TLV can either describe a simple AAA rule, such as a restriction posed on a specific resource for a specific user (or user group), or provide a reference ID to refer the 3D RCE to the actual policy data that are exchanged through the inter-domain communication channels as shown in Figure 1.

It should be noted that this AAA information is subject to the same set of issues and concerns as that of all the other link state database information. This includes scalability, accuracy, exposure of sensitive data outside ones own domain. The approach to dealing with these issues is the same for AAA information as the other TE information. Techniques such as topology abstraction, limiting distribution of this data to a per domain controller (i.e. NARB), and hierarchal routing techniques should allow individual domains to control the way their topology is viewed externally. In addition, it is not expected that AAA information will be as dynamic as other TE information. For instance, some TE information, such as bandwidth availability, will need adjusted based on provisioning actions. This is not expected to be the case with AAA information, so the issues with dynamic updates should be much less.

B. Policy Enforcement

Incorporation of AAA information into the 3D TEDB and 3D Path Computation does allow calculation of a path based on policy. These functions of the NARB and 3D RCE are not intended to be located in every LSR in a domain. Therefore there needs to be a simple mechanism to communicate policy decisions to LSR's such that provisioning can occur uninterrupted. A key goal is to be able to preload these policy directives in advance of provisioning actions. We plan to utilize the Common Open Policy Service (COPS) [15] protocol for support of policy provisioning (COPS-PR) [16] for this purpose. In order to be compatible with vendors who have implemented this protocol, we hope that we can achieve policy enforcement without needing changes to the current IETF specification of COPS-PR. However, we have not completed this analysis.

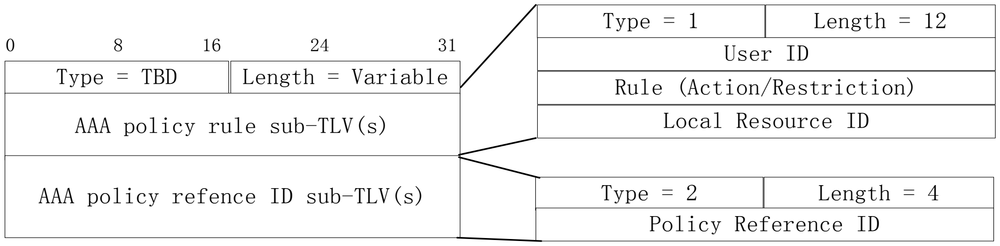

Figure 7: Format of AAA TE Link TLV.

VI. ADVANCE SCHEDULED SERVICE PROVISIONING

An important need for those who desire types of deterministic network services, is the ability to specify a time period for which an end-to-end path is needed. In response to this requirement, we have developed an approach to incorporate time windows into the provisioning architecture. The notion of time is associated with a TE resource via a new TE Link sub-TLV which is inserted into the 3D TEDB. This allows indication of a TE Link availability in increments of 15 minutes. Incorporation of this data into the path computation and provisioning process allows a path computation result which includes a scheduled start time and duration. We designate this a Schedule TLV and the current format we have defined is shown in Figure 8.

The enforcement of provisioning actions which have time associated with them is similar to that described above for AAA policy information and uses COPS-PR. However, time based provisioned paths will likely require more updates from the COPS-PR policy server. This is because as time periods begin or end, LSR's will need to treat provisioning messages differently. This is no different from a situation where AAA policy information has changed. However, it is expected that time schedule based provisioning will result in more frequent updates than standard AAA policy information.

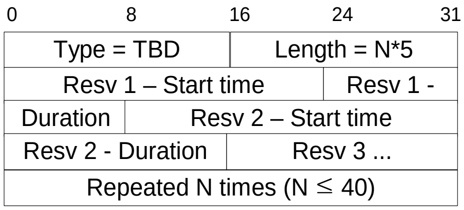

Figure 8: Format of Schedule TE Link TLV. (A Schedule TLV consists of up to 40 reservations. Start time (3 bytes) and duration (two byes) use timestamps measured in time slots.)

VII. IMPLEMENTATION STATUS

As part of the DRAGON project we are developing software based on the architecture presented in this paper. We have completed initial implementations of NARB, RCE, ASTB, ESA, and VLSR as identified in this document. Source code, related documents and software release information can be found at [17]. We are still working on some design and implementation issues associated with the integration of the AAA, scheduling, and policy information into the TEDB and path computation algorithms.

We have modified open source for some of our implementation efforts. For GMPLS routing we have extended the open source ZEBRA software (www.zebra.org). For GMPLS signaling, we have extended the KOM RSVP implementation from Darmstadt University of Technology. This includes extending these software components to include the MPLS/GMPLS RFC's and drafts such as RFC3630, RFC3477, RFC3471, RFC3473, draft-ietf-ccamp-ospf-gmpls-extensions-12.txt and others.

Our project activities also include building of on an experimental infrastructure in the Washington D.C. metropolitan area. This network consists of optical switching nodes and edge devices configured in a hybrid ring/mesh topology. The optical nodes are provided by MOVAZ Networks, are GMPLS capable, and include some early deployment of prototype equipment. The metro area network instantiated in the Washington D.C. area is architected around a multi-protocol all optical metro area WDM infrastructure. This allows maximum flexibility in terms of the type of end systems and applications which can be supported. Testing of our provisioning architecture and software components is currently underway.

VIII. CONCLUSION

In this paper we have described an architecture for GMPLS based multi-domain provisioning of end-to-end network paths which incorporates AAA policy and time schedule constraints. We strongly believe that development of control planes and techniques to allow this type of provisioning is absolutely critical in order to realize the full promise of optical network technologies. While we have focused on GMPLS environments, we note that many of the major issues in this space revolve around incorporation of the higher level information such as AAA and scheduling into the GMPLS provisioning mechanisms. In addition, many issues remain unresolved with basic GMPLS. This includes issues such as inter-domain routing, multi-region path computation, and signaling across heterogeneous network technologies.

In this environment, we do believe that link state protocols which incorporate topology abstraction and hierarchical routing techniques are the best method to incorporate higher level information into provisioning as well as address the other GMPLS issues. The combination of intelligent link state topologies with advanced multi-dimension path computation algorithms appears to be the best candidate for addressing the extreme heterogeneity in current (and likely future) end-to-end network paths.

An important goal of our architecture and implementation work is to test and evaluate various methods for achieving the objective of rapid provisioning of network resources in direct response to user requests. Our next step will be to evaluate several of these techniques from a performance, resource usage, and scalability perspective.

REFERENCES

[1] C. de Laat et al., “Generic AAA Architecture”, RFC 2903, August 2000.

[2] E. Mannie ed., “Generalized Multi-Protocol Label Switching (GMPLS) Architecture,” IETF RFC 3945, October 2004.

[3] D. Awduche el al., “Overview and Principles of Internet Traffic Engineering,” IETF RFC 3272, May 2002.

[4] J. Vollbrecht, “Lambda Grid, Grid Computing and AAA,” Merit Techical Report, April 2004.

[5] D. Katz, D. Yeung and K. Kompella, “Traffic engineering extensions to OSPF version 3,” IETF Internet Draft, Work in Progress, draft-ietf-ospf-ospfv3-traffic-05.txt, March 2005.

[6] G. Swallow, et al., “Generalize Multiprotocol Label Switching(GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model,” IETF Internet Draft, Work in Progress, draft-ietf-ccamp-gmpls-overlay-05.txt, October 2004.

[7] L. Berger (ed), “Generalized MPLS Signaling - RSVP-TE Extensions,” IETF RFC 3473, January 2003.

[8] A. Farrel, J-P, Vasseur and J. Ash, “Path Computation Element (PCE) Architecture,” IETF Internet Draft, Work in Progress, draft-ietf-pce-architecture-00.txt, March 2005.

[9] J. Vasseur, A. Ayyangar and R. Zhang, “Inter-domain Traffic Engineering LSP Path Computation Methods,” IETF Internet Draft, Work in Progress, draft-vasseur-ccamp-inter-domain-path-comp-00.txt, January 2005.

[10] A. Farrel, J-P. Vasseur and A. Ayyangar, “A Framework for Inter-Domain MPLS Traffic Engineering,” IETF Internet Draft, Work in Progress, draft-ietf-ccamp-inter-domain-framework-03.txt, June 2005.

[11] I. Foster, C. Kesselman, et al., “The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration,” Open Grid Service Infrastructure WG, Global Grid Forum, June, 2002.

[12] Globus Alliance, “Overview of the Grid Security Infrastructure,” Available on World Wide Web at: http://www.globus.org/security/overview.html.

[13] M. R. Gettes et al., “Internet 2 Shibboleth Web Authentication Project,” Available on World Wide Web: http://www.immagic.com/TOC/elibrary/TOC/meteor/downloads/shibblth.pdf.

[14] M. Handley et al., “SIP: Session Initiation Protocol,” IETF RFC 2543, March 1999.

[15] D. Durham et al., “The COPS (Common Open Policy Service Protocol),” IETF RFC 2748, January 2000.

[16] K. Chan et al., “COPS Usage for Policy Provisioning (COPS-PR),” IETF RFC 3084, March 2001.

[17] DRAGON Project, “Dynamic Resource Allocation via GMPLS Optical Networks,” available on WWW at: http://dragon.maxgigapop.net.