Network Aware Resource Broker (NARB)

and

Resource Computation Element (RCE)

Architecture

Version 2.1a
December 2007

University of Southern California (USC)
Information Sciences Institute (ISI)

http://dragon.east.isi.edu

Table of Contents

1. Introduction

This document provides an architectural overview of the Network Aware Resource Broker (NARB) and Resource Computation Element (RCE) which are components of the DRAGON GMPLS control plane. It is intended to providing interested readers, users and developers a description of the architectural design work on the NARB and RCE components.

The DRAGON architecture utilizes Generalized MultiProtocol Label Switching (GMPLS) as the basic building block for network element control and provisioning. There are several key functional elements identified in the DRAGON control plane architectures:

- Network Aware Resource Broker (NARB)
- Virtual Label Switch Router (VLSR)
- Client System Agent (CSA)
- Application Specific Topology Builder (ASTB)

This document discusses the architecture of the NARB component. Additional documents which address the other components are identified in the conclusion section of this document.

NARB and RCE are the components responsible for network resource provisioning and management. Because the RCE functionality is a coherent part of the overall NARB functionality, RCE is often considered a subcomponent of NARB. Therefore this document uses NARB and RCE in a combined architectural term NARB/RCE though they can provide services as separate, standalone control plane software. In the DRAGON control plane, NARB/RCE provides path computation, resource management, and Label Switched Path (LSP) provisioning services to the ASTB, CSA and VLSR.

2. Design Philosophy

The goal of the DRAGON project is to leverage the emergence and maturing of optical network technologies to develop and demonstrate the power and flexibility of a “hybrid” packet and circuit switched network infrastructure. Such network infrastructures are complex in nature. They are usually built over multiple layers of network technologies across multiple autonomous domains and probably using equipment by a variety of vendors. In the DRAGON architectural design, we need to address the complexity and provide reliable, flexible, scalable and inexpensive network services to end users. In particular, we propose to use NARB/RCE as the foundation to construct a control plane infrastructure to achieve the above goal. The larger vision is that of a shared infrastructure consisting of multiple network technology layers over which an intelligent control plane can provide for a unified service model.

Our design philosophy is summarized by the following five-S rules.

Rule 1 **S**ynthesis. GMPLS is addressing the 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, particularly in the context of inter-domain and multi-region topologies. Another important topic that is missing from the current GMPLS development activities is the notion of “policy based provisioning.” This implies the application of additional constraints to resource provisioning and allocation decisions. The new constraint dimensions we refer to are AAA and scheduling. In order for efficient and secure service provisioning and resource management, these multiple dimensions of constraints must be handled in a synergized manner. In the NARB/RCE architecture, they are fused into a centralized resource database in NARB/RCE, which is equipped with dedicated computing resources and sophisticated algorithms. Therefore NARB/RCE is sufficiently knowledgeable and powerful to generate efficient solutions and make quick decisions.

Rule 2 **S**egmentation. A general approach to addressing network complexity and scalability is breaking a large topology into small segments or domains. The NARB/RCE should be so designed that each network domain has its autonomous NARB/RCE component to handle the local network segment and manage local network resources precisely.

Rule 3 **S**hare. NARB/RCE in individual domains must be able to obtain a global view of the entire network topology. Such view could be summarized with reduced and/or abstract network resource details. This means that network resource information can be shared by distributed NARB/RCE instances dynamically with reduced overhead. To provision end-to-end services and manage the global resources, the distributed NARB/RCE instances should not only exchange resource information but also other control-plane intelligence that allows them to collaborate smoothly in various global network services.

Rule 4 **S**ervice. Unlike the IP routing service, the services provided by NARB/RCE are in the higher-level, e.g., the end-to-end inter-domain LSP service or the advance path scheduling service with application specific TE and policy constraints. With a unified service model, the NARB/RCE servers and other control plane entities will have the common language to negotiate a service with great flexibility and speed. This also allows flexible reconfiguration of the network control plane by adding, removing or customizing the participating control plane components in various network infrastructures and provides add-on services along with evolution of the network data plane.

Rule 5 **S**oftware. The DRAGON architecture is mainly designed for research network infrastructures. The target user group is the eScience community. All the control plane components are hardware independent. The NARB and RCE components and their depending modules are open-source software. Most users only need a Linux or FreeBSD box to run them with minimal installation and configuration effort. For most people in the eScience community the deployment of NARB/RCE is both convenient and inexpensive.

3. Functionality

NARB/RCE is designed to provide LSP computation, routing and resource management services in the GMPLS network control plane. Below is a list of features that have been or will be supported by NARB and RCE.

- Dynamic resource state collection (via both intra- and inter-domain OSPF-TE) and resource management
- Collection, formulation and distribution of network-, domain- and user-specific Authentication, Authorization and Accounting (AAA) information
- Domain topology summarization, abstraction and advertisement via OSPF-TE
- Multi-dimensional constraint based path computation
- Inter-domain routing
- LSP scheduling and management
- TE and AAA based policy decision

4. Architecture Description

This section provides a description of the NARB/RCE architecture. Some design details will be presented in next sections.

4.1. Overview

This subsection briefly describes the NARB and RCE in the context of the bigger DRAGON architecture as well as by their interactions with other DRAGON control plane components. Figure 1 depicts the DRAGON network configuration in a single autonomous domain. The inter-domain configuration will be covered in the next subsections.

As illustrated in Figure 1, NARB/RCE accepts LSP queries from CSA, ASTB or LSR/VLSR. NARB/RCE also participates in actual LSP provisioning by reserving, scheduling and updating resource states in its own resource database. Note that although the reservation, scheduling and update are performed in its internal resource database, such 'soft' changes in resource states still pose constraints on allocation of physical resources in the real network. NARB/RCE enforces the constraints by participating in policy decision and signaling authorization and by cooperating with the NARB/RCE peers in other domains.

The DRAGON NARB and RCE are implemented as two standalone software components though they are paired with each other in the DRAGON network. This design separates the computation-intensive RCE, which handles high-volume resource data using sophisticated algorithms, from the protocol-oriented NARB, which is focused on higher-level service provisioning, policy decision and inter-domain communication. This separation allows the two components to be developed and deployed in two parallel threads with higher flexibility.

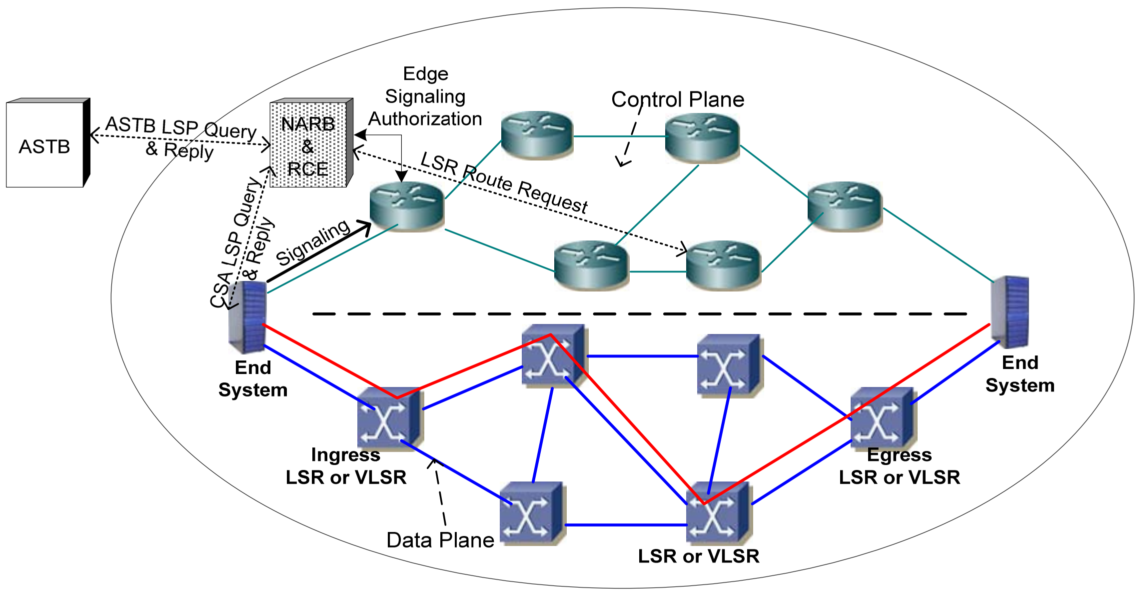

Figure 1: Example NARB/RCE configuration in the DRAGON network.

4.2. Resource Computation Model

The Resource Computation Model (RCM) is the theoretic foundation of the NARB/RCE architecture. Resource computation is the key phase that transforms raw resource and policy information into policy based GMPLS routing and signaling decisions. In this section, we present a three-dimensional (3D) constrained 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., TE constraints, time schedule constraints and AAA policy constraints, respectively, which have also been identified in the Design Philosophy Rule 1 (see Section 2). This model introduces the notion of a 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 LSRs to process provisioning requests rapidly.

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.

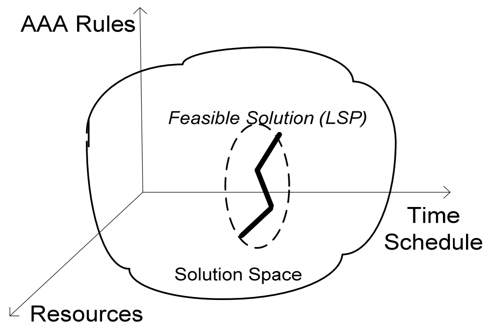

Figure 2: 3D constrained Resource Computation Model (RCM).

4.3. Inter-domain Path Computation

4.3.1. Inter-domain Routing Architecture

In multi-domain GMPLS networks, an 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. The NARBs 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 mining 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.

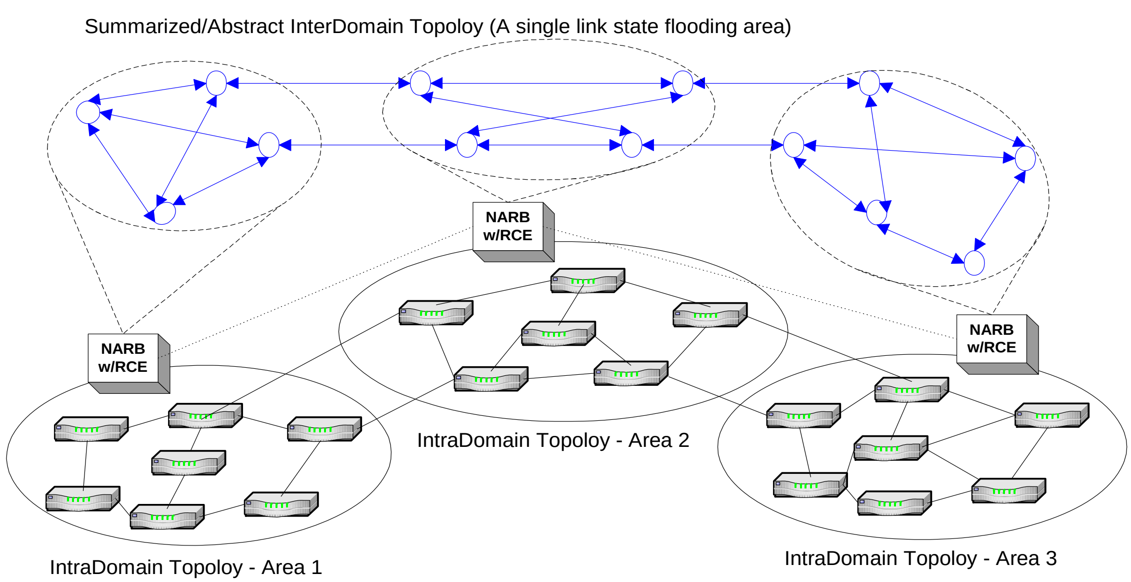

Figure 2: Illustration of the DRAGON inter-domain routing architecture.

The inter-domain routing architecture is illustrated in Figure 2. The NARB in each domain summarizes its intra-domain traffic engineering information and advertises an abstract topology to other domains. An instance of OSPF-TE is responsible for flooding the abstract topology information at the inter-domain level. RCE in each domain can therefore construct a global topology with summarized traffic engineering information. Meanwhile, one instance of OSPF-TE runs inside each domain, responsible for flooding physical traffic engineering information in an intra-domain 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 inter-domain OSPF-TE while global AAA policy information is exchanged via NARB-to-NARB communication. This architecture provides an effective inter-domain routing information exchange mechanism with reasonable scalability.

4.3.2. Inter-domain Path Computation Scheme

The NARB/RCE architecture supports two kinds of inter-domain path computation schemes. The first kind uses a centralized path computation mechanism, in which the source-domain RCE computes a complete end-to-end routing path for each LSP request. The other uses a per-domain path computation mechanism, which involves the RCEs of all the domains along the routing path. In this document, we firstly describe the Recursive Per-Domain (RPD) path computation scheme, which is the default per-domain path computation scheme in the DRAGON network.

The RPD scheme is illustrated in Figure 3. In the scheme, RCE synchronizes its resource database to both the inter-domain and intra-domain OSPFd 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 a explicit routing path 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 explicit route with all strict hops, is desired, the source-domain NARB/RCE asks the next-domain NARB/RCE 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 explicit route. In the actual process, 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.

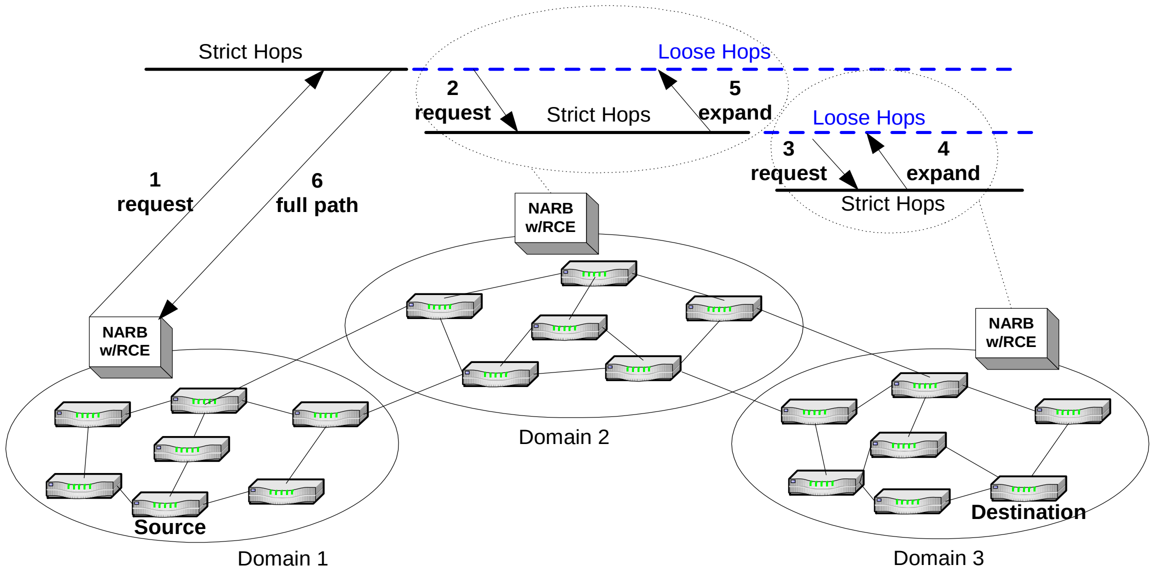

Figure 3: Illustration of the Recursive Per-Domain (RPD) path computation scheme.

4.3.3. Accommodations for Domain Privacy and Path Determinability

Using domain summarization/abstraction in inter-domain topology exchange is one way to keep private domain resources from being exposed to other domains. However, the RPD path computation scheme lets one domain obtain all-strict-hop path segments from other domains, which violates inter-domain privacy. The NARB/RCE architecture provides extra protection on such privacy by supporting path computation schemes that do not require strict-hop path information being passed across domains. One scheme is called forward per-domain (FPD) computation. In this scheme, NARB simply returns strict hops in the current domain with loose hops in next domains. The loose hops will let signaling proceed to the border routers in the next domain. Then the next domain border router will resolve the loose hops into local strict-hop path with its own NARB server (see Figure 4).

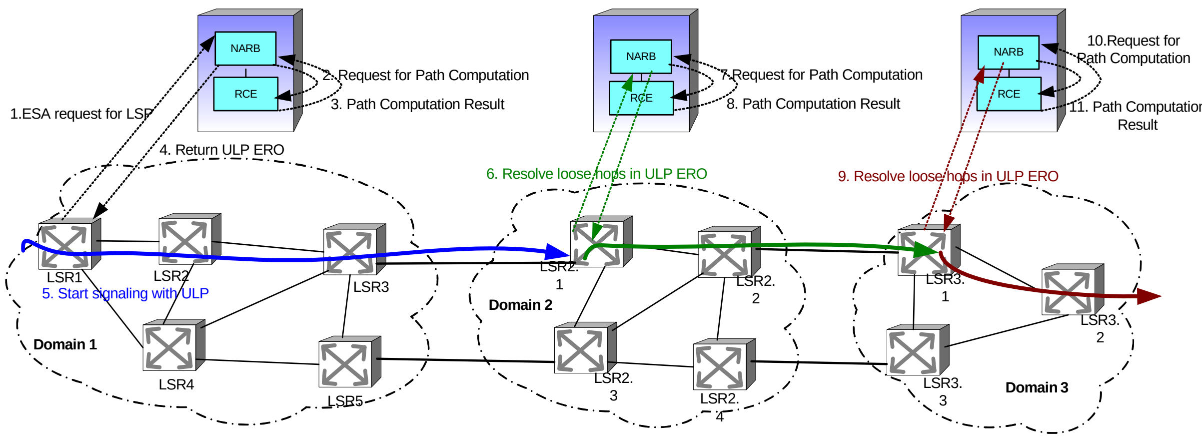

Figure 4: Illustration of the ULP-based Forward Per-Domain (FPD) path computation scheme.

The path provisioning based on the above FPD scheme is called an open-loop loose-hop solution. As an open-loop solution, the success of provisioning cannot be guaranteed since no end-to-end strict-hop path is computed before the signaling. Even if the strict-hop path does exist, there could be contention from simultaneous provisioning sessions during signaling time. To address the path determinability problem, two add-on features are introduced to NARB/RCE design. One is called path Query Confirmation (Q-Conf). Q-Conf allows a next domain to return a Confirmed Loose-Hop (CLP) explicit route as against the Unconfirmed Loose-Hop (ULP) explicit route in the open-loop solution. Following a recursive path computation procedure, the source domain receives a CLP with guaranty that an end-to-end strict-hop path exists (see Figure 5). The other feature is path Query Holding (Q-Hold). Q-Hold allows NARB/RCE to hold resources along the routing path in a recursive procedure. The signaling process will 'pick up' the resources being held using the obtained CLP. Q-Hold expires after a predefined period of time. Q-Conf and Q-Hold combined provide a closed-loop loose-hop path computation scheme to the multi-domain environments where both domain privacy and path determinability are desired.

4.4. Interaction with Other IDCs

The control plane entities that have NARB-like inter-domain topology exchange and path computation functionalities are called Inter-Domain Controllers (IDCs). NARB needs to interact with other implementations of IDCs in bigger and more heterogeneous environments. One type of unified IDC-IDC communication protocols under development is based on Web Services. The NARB/RCE architecture design covers two important aspects of the unified IDC-IDC interactions.

1. A more generic topology description method has been designed to facilitate the IDC-IDC topology exchange. This is a link state based description, which is not limited to be used by OSPF-TE.

2. Collaborative IDC-IDC path computation can exchange both CLP and ULP across domains but favors the closed-loop loose-hop path scheme using CLP. The all-strict-hop RPD scheme has been excluded. Unified format of the CLP and ULP explicit route object has been designed.

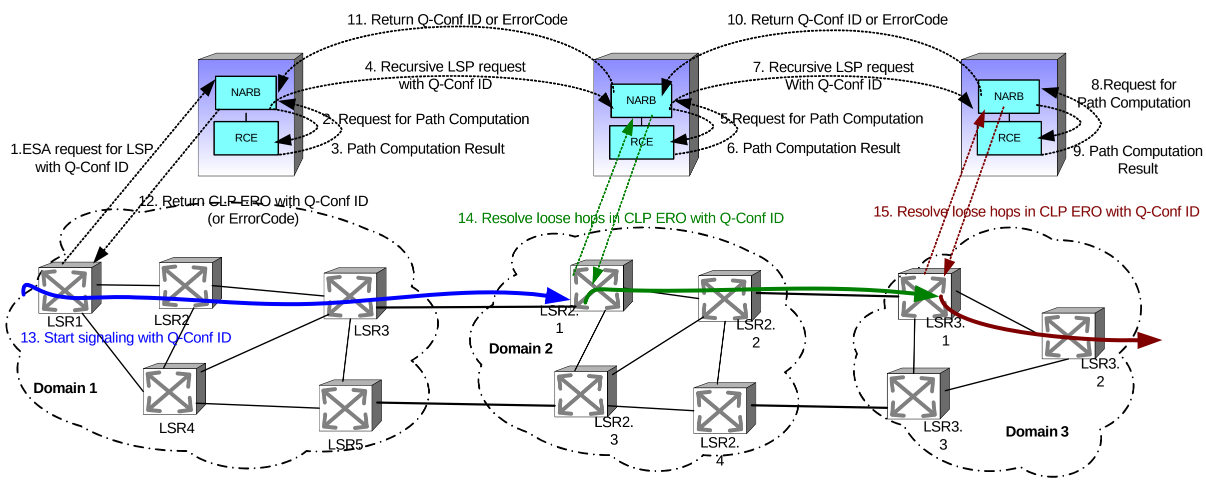

Figure 5: Illustration of the CLP-based Closed-Loop-Loose-Hop path computation scheme.

4.5. AAA and Scheduling Integration

With the emphasis on a unified service model, we need to translate these high 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 and scheduling information into information which can be utilized directly in the provisioning process. Our approach is the 3D RCM as described earlier in this document. In this model we synthesize the higher level AAA information into policy information which is associated with other information at the TE resource level. We also utilize the higher level AAA information to generate 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 addition to AAA policy, 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. In so doing, both AAA policy decision and LSP scheduling can be synergized with TE based provisioning decision, resulting in fast, precise and simplified control process.

Work on AAA policy and LSP scheduling related architectural design is currently in progress.

5. Conclusion

This document has provided a description of the NARB/RCE architecture design. While this document mainly elaborates the design of two control plane key components in the DRAGON architecture, it can also be used as a reference for architecting more generic network control planes that are based on the GMPLS technology. Those interested in deploying the NARB and RCE components in their networks or in the DRAGON control plane in general should further refer to the following documents available at http://dragon.east.isi.edu.

- DRAGON Control Plane Overview
- NARB and RCE Architecture
- NARB Design and User Manual

- RCE Design and User Manual
- VLSR Implementation Guide
- DRAGON and HOPI Network Control Plane Guide

6. DRAGON Project

This document was generated by the University of Southern California (USC) Information Sciences Institute (ISI) as part of the National Science Foundation funded Dynamic Resource Allocation via GMPLS Optical Networks (DRAGON) project. Additional details regarding this project are available at http://dragon.east.isi.edu.

Appendix A: Subnet Control Model

Here we discuss a special type of networking model that allows a subnet to be embedded into a domain. The purpose of this subnet control model is to harness the native control capabilities of vendor-specific subnets. The advantage in so doing is that we can use the vendor's native control for reliability and simplicity while still maintaining the integrity of the overall GMPLS control plane. For example, the Internet2 Dynamic Circuit Network (DCN) has a subnet of Ciena CoreDirector SONET switches, which provide flexible Ethernet-to-SONET mapping at every switch. The CoreDirector switches have robust Ethernet service provisioning capability between two edge ports via TL1 or GMPLS based OIF UNI interfaces. We want to make use of such native provisioning capability without going through the hassles to control the subnet from inside. By overlaying one or more Subnet Controlling VLSRs on top of the subnet, we can simply control it from edge to edge via standard interfaces such as OIF UNI or via vendor's native control API. Diagram of the Subnet Control Model is shown in Figure 6.

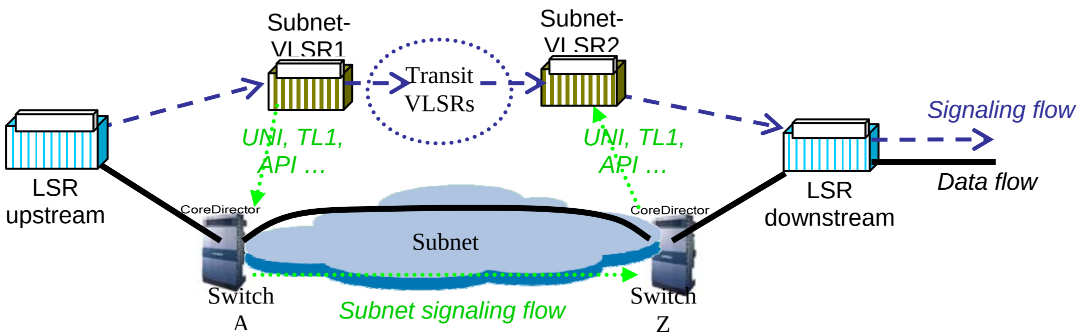

Figure 6: Illustration of Subnet Control Model.

The implications of this model for NARB/RCE design include the following. Firstly, RCE needs to have a detailed view of the Subnet topology so it can compute a deterministic path across the subnet. Secondly, RCE needs to incorporate the Subnet topology into the domain topology since the two topologies come from different sources and bridging links need to be identified or created in order to carry out crossing-layer path computation. Thirdly, NARB/RCE needs to map the data-flow path into a signaling-flow path because the actual signaling flow traverses the VLSR level instead of the Subnet level. Note that the number of Subnet VLSRs can range from one to the number of Subnet switches, but usually less than the latter number so one VLSR may correspond to multiple subnet switches. This separation of signaling flow from data flow is not only the nature of GMPLS control-plane separation but also a separation of DRAGON VLSR level control from vendor-specific Subnet level control, which makes the Subnet Control Model unique.

In addition to the VLSR-level ERO, the NARB/RCE path computation also generates a subnet ERO for a domain with a subnet. The subnet ERO can be returned to a client upon request. The client can therefor learn subnet-level path details. Also, a subnet ERO can be used to guarantee a deterministic, explicit path across the subnet during signaling time. This is particularly important for book-ahead or scheduled services. Subnet ERO can be converted into any vendor-specific explicit route representation, such as the Designated Transit List (DTL) for Ciena CoreDirector subnets. When combined with vendor-specific subnet signaling features, a subnet ERO or its converted form can help eliminate inconsistent routing paths between scheduling and signaling.