NET W65

Generalized Multi-Protocol Label Switching (GMPLS) Unified Control Plane Validation

Dimitri Papadimitriou
Alcatel Bell
Antwerpen, Belgium
dimitri.papadimitriou@alcatel.be

Bela Berde
Alcatel CIT
Marcoussis, France
bela.berde@alcatel.fr

Ricardo Martinez
Centre Tecnològic de Telecomunicacions de Catalunya
Barcelona, Spain
ricardo.martinez@cttc.es

Javier Gonzalez Ordas
Telefònica I+D
Madrid, Spain
javiord@tid.es

Remi Theillaud
ATOS Origin
Paris, France
remi.theillaud@atosorigin.com

Sofie Verbrugge
University of Ghent, IMEC
Ghent, Belgium
Sofie.Verbrugge@UGent.be

Abstract—**Generalized Multi-Protocol Label Switching (GMPLS) has become the protocol suite of choice for unified control plane implementation. However, its adoption is facing major challenges in terms of feasibility, performance, and gain when migrating from legacy packet over optical multi-layer networks driven by overlaid control planes. The ITEA TBONES project aims at tackling both objectives through the development of a platform including all the elements constituting such networks: network dimensioning, management plane and GMPLS control plane. This paper details the control plane architecture of the TBONES software platform, including its realization and applicability for multi-area networks, as well as the Testbed. Several experiments are described, both for validation and for demonstration of its capabilities and future usage.**

Keywords: GMPLS, unified control plane, software

I. Introduction

The benefits of a unified control plane (i.e. maintain a common control plane instance for a network hosting multiple switching layers) has become possible with the emergence of Generalized Multi-Protocol Label Switching (GMPLS) [1]. The latter provides for vendors and carriers, a unique opportunity to deploy a new set of advanced functionality taking advantage of this unified approach in terms of control plane inter-connection models covering diverse data plane switching technologies (from packet to circuit).

The TBONES project aims to demonstrate the feasibility of the implementation of a fully GMPLS-compliant unified control plane for multi-area and multi-layer environments. The project objectives also include the validation of the migration from an overlay control plane interconnection model (requiring a separate control plane instance per data plane switching layer) towards a unified control plane interconnection model where a single control plane instance drives a network comprising more than one switching layer. The latter constitutes the basis for building a unified control plane capable of managing nodes hosting more than one switching layer. Starting from the Internet Engineering Task Force (IETF) GMPLS-compliant User Network Interface (UNI) [2], this project has successfully implemented the mechanisms and protocols that provide support for such environments. Thereby, the present project demonstrates the adequacy of GMPLS unified control plane for multi-layer networks.

The TBONES emulator provided the means for the validating the unified GMPLS control. In particular, the emulator has been used for validating the dynamic provisioning of optical networks performed via a distributed control plane based on the GMPLS protocol suite, as defined by the IETF Common Control and Measurement Plane (CCAMP) Working Group. Besides the verification of the proper operation of the protocols and algorithms, the TBONES emulator allows for quantifying the performance, such as provisioning and recovery speeds, and assessing control plane scalability. Traffic Engineering (TE) and recovery (i.e. protection, rerouting and/or restoration) mechanisms are included. Interactions at the level of the control plane, between the transport network and its clients (with a specific focus on IP/MPLS clients) are investigated. This requires the simulation of both transport and client networks. The emulator focuses on the control plane, with some minimal modeling of the transport plane required for the simulation support: network element functionalities, aggregation of (IP/MPLS packet) flows in wavelengths, and data plane failure simulations. Finally, the TBONES emulator interfaces with two external entities: the management plane emulator and the control plane of the Testbed for validation of the proper inter-working with these two entities.

This paper is organized as follows. In Section II, we introduce the TBONES emulator, the various software components and describe the implementation of the GMPLS control plane software. Section III details the experiments that have been conducted for have been conducted to validate the TBONES emulator capabilities, performance, scalability and

1-4244-0355-3/06/$20.00 (c) 2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

interoperability. Section IV provides an analysis of the experimental results and observations obtained during the execution of these experiments. Finally, we list in Section V the main conclusions drawn from this work in support of GMPLS control plane capabilities and implementation for multi-layer packet optical networks.

II. CONTROL PLANE ARCHITECTURE AND INTERACTIONS

A. Multi-Layer Network Architecture

Operators progressively deploy networks including more than one switching layer. In these environments, the control plane (CP) integration is a key enabler for network resource optimization and more notably operation simplification. A good example of such a network includes IP/MPLS, Ethernet, and lambda switching capabilities under the supervision of a single GMPLS CP instance. For instance, such a system allows MPLS Label Switched Paths (LSP) to be set up on top of Layer 2 LSPs, themselves nested into Lambda LSPs. A unified CP approach allows for a single controller (i.e., a single GMPLS CP instance) to handle multi-layer capable networks. Hence, in the unified model context, the GMPLS protocol suite currently assumes that each of these LSPs can be established using a common instance of the CP.

In a unified GMPLS CP, the set of routing adjacencies, topology and traffic engineering information are maintained by a single routing protocol instance. Therefore, all nodes configured to be part of this instance have a common view of the links (and their Traffic engineering properties) belonging to this network. Being agnostic to the increasing number of data plane switching layers, the unified CP approach allows a single addressing space to be maintained, together with the optimized and automated operations of the multi-layer network. Realization of the multi-layer unified CP is based on concept of Interface Switching Capability (ISC) [3] that refers to the ability of a node to forward data of a particular type. For example, PSC (packet switch capable) is associated with an interface, which can delineate IP/MPLS packets (e.g., a router's interface) while LSC (lambda switch capable), is associated with an interface, which can switch individual wavelengths multiplexed in a fiber link (e.g., an optical cross-connect - OXC's interface). Links in the Traffic Engineering (TE) database are identified by their switching capabilities (at both ends). TE Links are maintained in a single Traffic Engineering Database (TED). The representation, in a GMPLS CP, of a switching technology domain is referred to as a (LSP) region [3]. An LSP region is as a set of one or several switching layers that share the same type of switching technology. Examples of regions are Packet (PSC), Layer 2 (L2SC), and Lambda (LSC). Hence, an LSP region is a technology domain (identified by the ISC) for which data links are represented into the CP as an aggregate of TE information associated to a set of links (i.e. TE links). Since this TED contains the information relative to all the different regions existing in the network, a path across multiple regions can be computed using this TED. Thus optimization of network resources across the multiple regions can be achieved.

B. TBONES Software Architecture

The TBONES emulator consists of four pieces of software: two off-line tools that are run prior to any simulation, and two emulators: a control plane (CP) emulator (depicted in Figure 1) and a management plane (MP) emulator. The CP emulator also embeds a simple transport plane simulator to realistically allocate or free network resources.

The first off-line tool is a *Request Scheduler* that takes as input a network topology and a traffic profile, and generates a set of traffic demands for each emulated node. To compute each node dimensioning, and a virtual topology, the same topology and traffic demands are used by the second off-line tool, the *Dimensioning Tool* (DT). The virtual topology is the initial set of wavelength soft-permanent connections (SPCs) that provides the mean to setup packet LSPs across the network. Additional SPCs will be automatically setup - or teardown, by the CP emulator to accommodate packet traffic changes, based on various policies. All the aforementioned files are Extensible Markup Language (XML) files. The topology and dimensioning files are used during the initialization of the CP emulator, to bring up each emulated node. Each emulated node also embeds a scheduler in order to trigger switched connections (SCs).

Running on Linux kernel 2.6, the GMPLS CP emulator offers, as shown in Figure 1, a centralized GUI to interactively monitor and query each simulated node. Each node is made of two processes: a lower process runs Open Shortest Path First Traffic Engineering (OSPF-TE) [4] and Resource ReSerVation Protocol Traffic Engineering (RSVP-TE) [5] engines, while the upper process runs the set of GMPLS controllers. The lower process exchanges RSVP-TE and OSPF-TE packets through a process that emulates point-to-point sub-networks (software loopbacks). Moreover, each lower process may access to an Ethernet interface to communicate with peering emulator(s), and the management plane emulator. The lower process(es) attached to an Ethernet interface behaves as an IP router in front of the other emulated nodes. The OSPF-TE protocol engine mainly handles OSPF adjacencies maintenance, and LSA reliable flooding. Global and per-interface pacing mechanisms are implemented for scalability purpose. While the OSPF-TE protocol engine maintains a database with all raw LSAs in order to achieve efficient LSA flooding, the upper Traffic Engineering controller maintains a database with all routers, TE links and reachable addresses extracted from received LSAs [3]. The synchronization between these two databases is asynchronous, so that the latter can be locked by a path computation procedure without preventing the former to be updated and LSA flooding to be performed. LSC LSPs are inherited as packet-switching capable links, or forwarding adjacencies, and may be bundled too. The latter database also contains all local data-links that are not directly advertised because they are bundled into a single TE link. The RSVP-TE protocol engine mainly handles Path and Reservation state refresh, and reliable message delivery. The RSVP Summary Refresh extension is implemented for scalability purpose. While the RSVP-TE protocol engine takes care of all states refresh, the upper Signaling controller processes all GMPLS RSVP-TE trigger messages [6], and maintains a database with all LSPs setup across the local node. Bidirectional LSP setup can lead to resource allocation collisions; burst of LSPs setups may lead to path computation being performed on transiently not up-to-date TE database; end-to-end and boundary crankback [7] is therefore enabled. In addition to signaling and routing updates, and traffic demands triggers, the upper process receives inputs from the management plane emulator, and from the centralized console. A dedicated thread handles each of these five events sources.

A CAC function of the CP performs checking of LSP requests. Connection Admission Control (CAC) policies are pre-provisioned on a per network edge node basis. These policies comprise a set of rules that applies to the incoming traffic requests. The following usage of policies implies a local decision making process. The policy condition is checked locally in individual CAC controllers. According to the LSP and configuration information, the CAC accepts or rejects LSP requests. More specifically, we have to configure how each LSP request is filtered based for instance on the user id or on the IP address. For example, the following rules are maintained on per identified client basis:

- number of LSPs N an identified client can request
- number of LSPs n (n < N) an identified client can request during a period of time t
- number of LSPs n (n < N) an identified client can request during a period of time t for a working hour H
- number of LSPs n[d], n[D] an identified client can request toward known destination d or set of destinations D (destination are identified as the receiver's IP address)

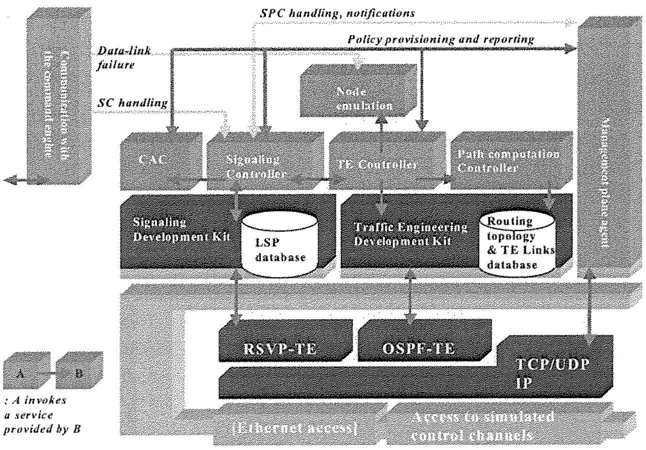

Figure 1. TBONES CP Software Architecture

The POLICY_DATA object [8] is included as part of the incoming requests to allow user identification and (optionally) determine the application for which the resources are requested. Then in order to take a decision i.e. accept or reject the LSP request, statistic information is computed. For this purpose, several statistics on number of LSPs (i.e. per user, per class of LSP i.e. TDM or PSC for example, per source address or destination address) where used.

C. *Interactions*

An XML-based interface between the Network Management System (NMS) and the CP emulator supports configuration management (CM) and fault management (FM) functionality in TBONES. Detailed information about LSPs and alarms is exchanged across this interface to facilitate the emulation of a range of management approaches. The NMS uses two interactions patterns to communicate with the CP. The provisioning, rerouting and release of SPCs follow a request-response model, initiated by the NMS, whereas for SCs – autonomously setup by the CP-, restorations and alarms unsolicited notifications are sent upstream. The interface primitives are carried as XML messages over TCP/IP sockets. XML was chosen over management specific protocols – such as Simple Network Management Protocol (SNMP)- for its flexibility, extensibility, ease of integration and simple protocol stack. The NMS establishes two communication channels –one for requests and the other for notifications- with each emulated CP node. When the NMS starts the creation of an LSP, it constructs an XML message containing:

- The source and destination node IP addresses.
- The type of LSP: supported values are PSC (for packet switched LSPs) and LSC (for lambda LSPs).
- The partial or entire end-to-end route, expressed as a set of loose or strict hops that must be traversed by the LSP. Each hop may represent a node, an interface or a label within an interface (e.g. a specific wavelength).
- A set of nodes or links exclusion constraints.
- The re-routing scheme to use: dynamic, pre-planned or no re-routing [9].
- The crankback scheme to use: end-to-end, source or no crankback [7].

The NMS sets up LSPs upon operator request or automatically to build the initial virtual network topology, which is read from a dimensioning file. Depending on a configuration option and on operator preferences, the request may contain a full or partial path, or only the end nodes. Restoration and crankback parameters are selected per LSP. This enables to experiment with various path computation and restoration strategies. The NMS sends the LSP provisioning message to the source node. If it does not contain the entire path (i.e. if there is any loose hop in the request), the source node computes the path to the first loose hop, and triggers RSVP-TE signaling. The entire path is completed by the loose hop nodes and ABRs until the destination is reached. Unless the incoming request specifies which labels to use, nodes along the path perform local label selection. If the LSP setup fails over this first path, the CP will try alternate routes. The interaction eventually ends with a response to the NMS from the source node, reporting the unique LSP identifier (for the source-destination address pair) and its actual path (including labels), or a failure condition. The NMS will use the identifier provided by the CP when performing any additional action on the LSPs, such as rerouting or removing it. When the CP restores a LSP, its head-end node issues a notification towards the NMS. To allow the NMS to permanently keep track of the state of the LSPs and of the resources allocated to them, the notification conveys the new path of the LSP after restoration. Although exchanging this data adds significant load to the interface, its availability enables to study management approaches where the NMS always has full view of the network state. The notifications channel also carries the switched connection setup and release messages. Again, the full path is reported to the NMS as part of the setup notification. Finally, alarms regarding failing links are sent from the nodes adjacent to the fault to the NMS using an X.733 based structure.

The PBM solution, used in TBONES, consists of two main components: a policy server and an adaptation layer. The policy server supports policies compliant with Policy Core Information Model (PCIM) [10]. The adaptation layer adapts the policy server rule(s) into a configuration understandable by the target network element. This configuration is downloaded into the node using a specific XML over socket communication. The PBM also provides a monitoring tool for visualizing the repository contents, the state of each rule and log information on policy server execution. For validation and demonstration purpose, this adaptation layer provides a graphical view of the rules being enforced and where, in the network, these rules have been enforced. More specifically, the view illustrates the XML-based set of rule (or decision from the policy server point of view) sent to each node. Once the user clicks on a rule, automatically the network topology view is updated in order to highlight the enforcement point.

III. EXPERIMENTS

Several experiments (including measurement of the CP emulator performance) have been conducted to validate the TBONES software capabilities, performance, scalability and interoperability.

A. Cost-based evaluation experiments

The dimensioning tool developed within the TBONES project allows studying three network evaluation cases. A first case compares the required capacity for a GMPLS optical transport network (OTN) to that for the corresponding statically configured OTN. A second case explores the optimal evolution scenario from static to GMPLS controlled OTN. A third case studies the problem of a logical topology design that fully exploits the flexibility of GMPLS CP for reusing resources in a multi-layer recovery scenario.

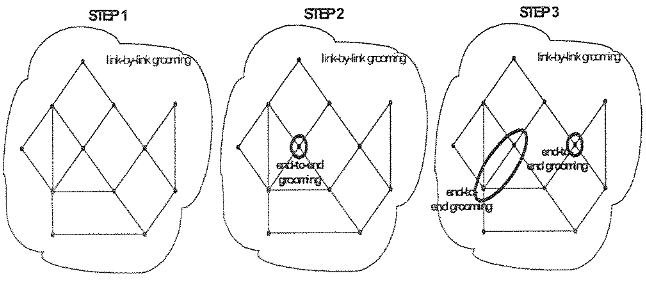

Figure 2. Introduction of end-to-end grooming islands in link-by-link grooming network

Concerning the second case, it should be noted that optical cross-connects (OXC) are still very expensive today, so that it is important for the operator to find the optimal introduction time. By dimensioning and optimizing the cost for increasing traffic, we suggest the following scenario in which the cost of this evolution is minimized. Island based grooming introduces OXCs only in some nodes of the network. Those nodes are transformed into little islands where we use end-to-end grooming (all transit traffic can be sent end-to-end on the optical layer). Gradually more of these end-to-end grooming islands get introduced. An example is given in Figure 2. In the beginning (step 1) link-by-link grooming (where traffic cannot be switched on the optical layer and the IP layer topology equals the physical layer topology) is used throughout the whole network. If a certain node gets too heavily loaded, we install an OXC in that node so that it becomes an end-to-end grooming island. In step 2 we notice one such node. As traffic grows, more of these islands get introduced and islands can be merged to become a bigger island. This is what we see in step 3, the island of the previous step has grown and a new one has appeared. Eventually (if the traffic keeps growing) the whole network can become end-to-end grooming. This intermediate (island) step in the migration from link-by-link grooming towards end-to-end grooming allows spreading the expenses for OXC introduction.

	2004	2006	2008	2010	2012	2014
Number of end-to-end grooming nodes	0	7	13	15	19	20
Number of islands	0	2	2	2	2	1
Gain compared to link-by-link grooming	0%	-5%	9%	23%	31%	34%
Gain compared to end-to-end grooming	75%	53%	24%	10%	5%	3%

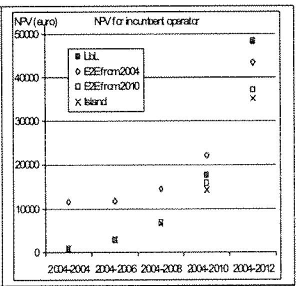

Figure 3. Comparison of different grooming approaches

We have considered a reference pan-European network topology with associated traffic demand model [11]. A 100% IP traffic growth is assumed and the time frame 2004-2014 is studied. We have dimensioned the network and evaluated the equipment costs in different scenarios for 30% of the total traffic demand predicted by the model (assume that big operator is able to attract 30% of the overall traffic). We use an IP-over-Optical equipment cost model with decreasing costs over time [12]. The left part of Figure 3 shows the evolution of the number of end-to-end grooming islands installed in the network and compares the CapEx (total equipment cost) to dimension the network using link-by-link, end-to-end and island based grooming. As there are no islands installed in 2004, the expenses for the island based grooming approach are equal to those for the link-by-link grooming approach. In this case, the gain obtained compared to end-to-end grooming is 75%, so that introducing OXCs would definitely not be a good idea here. As the traffic grows, more islands are suggested by the island based grooming approach, so that the gain compared to end-to-end grooming (with OXCs in all nodes) decreases, towards 3% for 2014 (20 of the 28 nodes have OXCs). On the other hand, the gain to be made compared to link-by-link

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

grooming increases with growing traffic, from 0% in 2004 to 34% for 2014. Note that the gain of -5% in 2006 follows from the particular calculation method and is only a transition phenomenon. The right part of figure B compares the NPV for island based grooming to that for link-by-link and end-to-end grooming, for several values of the economical lifetime of the project (planning interval). We see that, for N up to 4 years (considered interval 2004-2008 or smaller), three of the considered options have similar NPV: expansion using link-by-link grooming, island based migration and network-wide migration in 2010 (which is actually equal to link-by-link grooming in the considered interval). Network-wide migration in 2004, leads to significantly higher NPV and therefore needs to be avoided. When considering a planning interval of 6 years (2004-2010), it becomes clear that network-wide migration in the year 2010 is a better option than continuing to keep expanding the network using link-by-link grooming. Island based grooming is clearly the best option. Enlarging the planning interval even further makes clear that migration towards end-to-end grooming definitely is to be preferred over the continued use of link-by-link grooming. With a planning interval of 8 years (2004-2012), network expansion based on link-by-link grooming is the worst option. Even immediate migration towards end-to-end grooming (in 2004) is better.

B. *Performance and Scalability GMPLS Control Plane*

Several performance experiments have been conducted using the TBONES GMPLS CP software using the network topology depicted in Figure 4 that includes 28 nodes and 42 links. These experiments aim to demonstrate GMPLS CP supported load and performance including OSPF(-TE), RSVP(-TE) stacks and the different GMPLS controllers.

Figure 4. Pan-European Network and Routing Topology

For instance, OSPF(-TE) performance experiments include:

a) LSA/opaque TE LSA processing time: verify dependency on LS update packet size.
b) LSA/opaque TE flooding (to neighbors) time: verify dependency on pacing (intervals).
c) SPF/CSPF computation time: verify dependency on the number of links and nodes.
d) RIB/FIB update (CP level): verify de-correlation from number of link and nodes.
e) Scalability enhancement delivered using link bundling on (a), (b) and (c).
f) Impact of multi-area exchanges on performance:
   - Type3_LSA: using an increasing number of inter-area prefixes until reaching saturation.
   - Type4_LSA: using an increasing number of Autonomous System Boundary Routers (ASBR) with an increment of 1 until reaching saturation.
   - Type5_LSA: from the previous increasing number of ASBRs, inject an increasing number of external prefixes per ASBR.

The capability to emulate multi-area TE environments as depicted in Figure 4 has also been experimented. The backbone Area 0 is (among other) responsible for distributing routing information between non-backbone areas. Each Area Border Router (ABR) has complete topological information concerning the backbone, AS-External prefixes, and routes to ASBRs and summarized information from each area connected to the other ABRs. In their turn, the ABRs by flooding Link State Update packets populate their locally attached area Link State Databases (LSDB). Type10 TE LSAs are exchanged within each area to describe the TE attributes of their internal links (in particular, the links interconnecting the Area 0 ABRs). The path computation controller (PCC) uses this reachability information and the local area TE information, to compute loose routes from the ingress to the egress node (as determined by the request scheduler) associated to another area. Then, the SIGC initiates signaling of the multi-area LSPs.

Several experiments were also conducted to determine the supported number of LSPs as well as the number of RSVP-TE states and the sustainable state refresh rate. For this purpose, using the topology depicted in Figure 4, 10000 LSPs were simultaneously and successfully setup. However, their maintenance is not ensured, as some states are progressively lost. An additional experiment using 5000 LSPs shows that the signaling controller can safely maintain all LSPs without loosing any state. Therefore, to achieve maintenance of O(10k) LSPs further refinement is required at the signaling controller level. Note however that use of fast processing of Refresh messages becomes critical in such highly loaded environments; thus, the use of SRefresh message [5] is highly recommended. In summary, in terms of number of LSPs the order of magnitude that the GMPLS CP software can currently support is O(1k). To reach a higher order of magnitude i.e. O(10k) several recommendations have been devised - see Section IV.

C. *Interoperability*

Interoperability of the developed GMPLS CP software was demonstrated by interfacing the TBONES emulator with an external GMPLS controlled entity (Testbed). The Testbed is an experimental transparent optical network composed by a set of real and emulated optical nodes and links in a bidirectional ring network topology. The objective of using the Testbed control plane (CP) to interact with the developed TBONES emulator supporting dynamic LSP provisioning in a multi-domain optical network is the validation of the inter-working between domains.

The routing topology considered for the interconnection between the TBONES emulator and the Testbed is a multi-area single AS network (similar to the one depicted in Figure 4). In such routing topology, the OSPF-TE routing information exchange through the backbone area 0 serves as the baseline scenario wherein end-to-end LSPs are established across the two entities. For this purpose, RSVP-TE signaling protocol that considers Summary Refresh extension and reliable message delivery is employed to set up both soft-permanent and switched connections. These LSPs are established across multiple areas by using the routes computed by the path controller. The role of this controller placed on every network node is to compute explicit routes on a per-domain basis by using the information populated by OSPF-TE such as summarized network topology, reachable client end-points and TE link properties (e.g. unreserved bandwidth). Hence, a Constrained Shortest Path First (CSPF) algorithm is utilized to calculate the shortest path for a pair of nodes with non-zero unreserved bandwidth on every route link. By doing so each entry boundary node is responsible for computing the path to the next exit boundary (ABR or ASBR) until reaching the destination node using the intra-area TE link information by the described CSPF-based algorithm. Finally, the computed explicit route (with loose objects) is passed to the RSVP-TE signaling Path message with the aim to set up the bidirectional LSPs between any two points of the whole network.

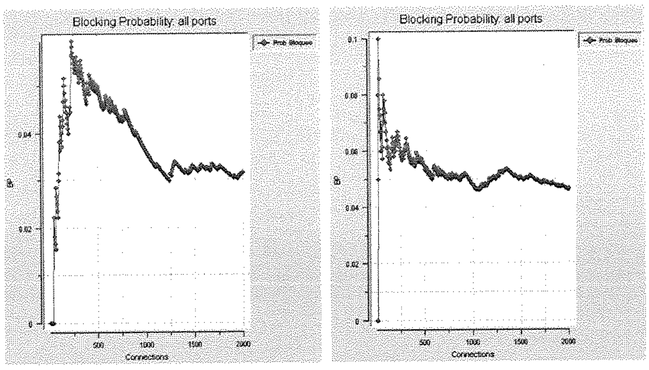

Figure 5. Blocking Probability for 1 Erlang: a) uniform and b) non-uniform traffic for test 1 (SCs requests)

In order to demonstrate the signaling, routing and path computation procedures described above two provisioning tests have been considered besides the protocol (i.e. OSPF-TE and RSVP-TE) interoperability and validation. The former test here referred as tests 1 deals with the establishment of SCs between the Testbed and TBONES emulator and the later referred as test 2 is addressed for setting up SPCs between Testbed and TBONES emulator. Firstly, we describe the most important assumptions adopted during the experimental process of the above tests: all LSC (Lambda) LSP requests to be established are bidirectional; the dynamic traffic generation of each bidirectional LSP requests arrive according to a Poisson process modeled with a mean inter-arrival time of 60s and the LSP holding time is negative exponentially modeled with a mean holding time of 60s as well; then the total offered load to the whole network is 1Erlang; the traffic is supposed to be uniformly and non-uniformly among all node pairs for test 1 and test 2; and each data point result was obtained over a generation of 2000 LSP requests. Finally, as an example of the interoperability between the TBONES emulator and the Testbed, Figure 5 plots the blocking probability performance of the test 1 (i.e. SCs requests) for uniform (Figure 5a) and non-uniform (Figure 5b) traffic.

IV. Experiment Results and Observations

The TBONES dimensioning tool allows assessing from a CAPEX point on view different scenarios for IP over optical networks, to quantify the following points:

1. Is there an interest to introduce wavelength cross-connect under IP routers to reduce the amount of transit traffic handled at layer 3. Our studies have confirmed that wavelength cross-connects to bypass IP routers can result in a cost benefit when the traffic volume (and more precisely the amount of transit traffic) becomes large enough. In a realistic case, where a pan-European network operator is able to achieve a market share of 30% (of the total volume of forecasted European traffic), the year 2010 seems the most suited time-period to shift from a pure packet-switched (link-by-link grooming) to a pure wavelength cross-connected (end-to-end grooming) network solution. In these studies, two major aspects have been demonstrated. First, by deciding on a node-per-node basis to introduce a wavelength cross-connect (i.e., creating/extending an end-to-end grooming island) helps in significantly reducing the overall network cost (over the complete planning horizon) and favors an earlier introduction of the wavelength cross-connects. Secondly, the importance of considering a sufficiently large planning horizon has been emphasized: the longer the planning horizon, the more opportunities there are to incorporate the benefits a wavelength cross-connect represents after the moment it becomes more cost-efficient.

2. Do these cross-connects present an interest from a resilience perspective thanks to the dynamic configuration capabilities of the transport network they are allowing. Our studies have demonstrated the benefits of flexible/dynamic optical networks in comparison to static OTNs for the recovery against router failures. This is because a better re-use of resources is possible, due to the dynamic reconfigurability of the logical topology, allowing the setting up and tearing down of LSPs at the time of the failure event. Also, it has been shown that GMPLS for survivability compared with the static IP re-routing approach results in lower capacity requirements roughly in the range of 5% to 15% in total cost. If in addition the logical topology design is optimized in an integrated way, instead of using a manual design heuristic, an additional 10% cost improvement was observed.

3. Does the flexibility provided by the cross-connects bring gains in presence of dynamic traffic. Our studies have shown, for some typical pan-European backbones, that the flexibility allowed by the cross-connects can bring significant overall network capacity-savings in the presence of a dynamic traffic. This is due to the statistical multiplexing which is possible in the more flexible GMPLS-capable network. The effect of this statistical multiplexing increases when the traffic becomes more dynamic, resulting in more capacity savings. Our studies have also shown that these capacity savings increase when the meshedness of the considered network decreases. Finally, it is shown that an increase in the multiplexing value between the

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

two considered network layers also results in an increase of the overall network capacity-savings.

4. Is transparent (e.g. without opto-electronic conversion) switching technology of interest for implementing the cross-connects. Our studies have shown on a typical pan-European backbone that a hybrid cross-connect (transparent switching in association with a limited set of opto-electronic regenerators) can bring significant overall network element cost savings (more than 15%) as long as the ratio between the cost of a opto-electronic switching port and the cost of a full optical port is at greater or equal to 5. This later figure should give a target for optical switching element designers; it should be reachable unless price of high-speed opto-electronic devices dramatically falls. The overall network element cost savings depend upon the maximum achievable non-regenerated transmission length, but it is bounded to about 30% for transparent lengths of practical interest. Depending on the adopted technology, a Maximum Transparency Length (MTL) of 1800 a 2800 km seems achievable: such MTL values are sufficient to obtain a significant cost reduction by introducing transparent cross-connects in the considered European backbone. Of course, the smaller the network, the less critical the MTL-constraint becomes.

To sustain the deployment of a multi-layer network, a unified GMPLS CP was initially assumed to be the most suitable in delivery of sufficient performance, flexibility and scalability. The conducted experiments using the TBONES software demonstrates that the GMPLS signaling and routing specifications developed at the IETF CCAMP WG address most carrier requirements in terms of performance, scalability and flexibility of a unified CP. The key mechanisms to provide a unified GMPLS CP are:

1. *Separation between protocol-specific and application-specific mechanisms.* GMPLS unified CP architecture is realized by separating protocol-generic mechanisms from application-specific mechanisms. In the unified CP context, the term application refers to GMPLS RSVP-TE and OSPF-TE. As a consequence, soft-state protocol-generic mechanisms (such as OSPF neighbor relationship maintenance, OSPF LSA reliable flooding but also RSVP acknowledgement, RSVP Path/Resv states refresh and monitoring, etc.) are implemented once and for all in the lower part of the software (referred to as "protocol stack"). Hence, all the GMPLS RSVP-TE and OSPF-TE applications (and potentially others for future GMPLS developments) share the same protocol stack mechanisms.

2. *Two-stage OSPF architecture and database.* The OSPF database includes a topology LSA database that contains the received raw LSA packets, for flooding purpose by the OSPF-generic protocol stack and a TE LSA database (also referred to as Traffic Engineering Database or TED) that includes application-specific TE link data (i.e., therefore preventing from re-processing every TE link data information whenever a CSPF is run). The update of the TED from the LSA database is performed asynchronously with a flow control. As the TE database that contains application-friendly data and its updates are performed asynchronously, the CSPF execution is free to lock the TE database (preventing any update) without any impact on the OSPF flooding mechanism itself. When the CSPF run is completed, the TED can be updated and synchronized with the LSA database.

3. *TE link as unique application-specific entity.* The GMPLS CP handles any TE entity as a TE link in the TED that makes use of a fully recursive definition of TE links. The TED is also used to store local component TE links comprising a set of one or more data links (even though they are not advertised; only the TE link bundles are advertised). The same mechanism is also considered for forwarding adjacency LSP (FA-LSP) stored as FA links in the TED. The simplification and flexibility that result from this implementation is such that the only processed entities for the TE controller and Path computation module are TE links, defined as resource aggregates that are encoded as links with TE attributes.

4. *Unified Traffic Engineering (TE) information processing.* A unified GMPLS CP must allow for maintaining technology agnostic traffic engineering information processing. Such powerful processing allows for maintaining efficient, scalable and flexible processing of information pertaining to different switching capabilities. It eases the addition of a new switching capability as part of the controlled network while preventing for modifying any fundamental TE link operation. The resulting recommendations are 1) maintain TE link and related information processing as technology agnostic as possible 2) when applicable process the switching capability information for pruning TE database in order to reduce the set of TE links on which path computation has to be operated and 3) process uniformly any other TE attribute.

5. *RSVP state maintenance and LSP Database.* In order to improve processing of GMPLS RSVP-TE, signaling, the lower RSVP protocol stack maintains a Path/Resv states database, that support summary state refresh mechanism (only a message ID is stored and processed). An LSP database is offered to the application (this database contains most RSVP TE objects, pre-processed and decoded is order to simplify application specific processing).

6. *Asynchronous, prioritized and multi-threaded Path Computation.* The GMPLS CP has the capability to asynchronously perform path computation, to prioritize and preempt path computation requests, execute several path computation requests in parallel but also for a particular path computation request to execute multiple runs. Prioritization of the path computation provides the capability to give precedence to high priority over low priority LSP requests. For instance, in case a burst of LSPs has to be dynamically re-routed, this prioritization mechanism allows for giving precedence to higher priority LSPs.

Another fundamental outcome from these experiments is the stringent need to provide for an efficient policy based management (PBM) of the CP operations in particular for multi-layer packet-optical networks as considered in the TBONES context. The experiments led in this project have demonstrated the need for an efficient policy-based connection and resource admission control, an efficient policy control on TE routing (e.g. TE link attribute flooding) and CSPF operations (e.g. path computation) and finally on efficient policy control on Signaling Controller (SIGC) operations involving of GMPLS RSVP signaling parameters. In particular, experiments have shown that in order to achieve efficient TE control of network resource usage the specific classes of policies must be considered [13].

The management plane (MP) experiments focused on the validation of the system and of the MP-CP interface. Although MP implementation performance was not an objective, processor load and provisioning time measurements were taken to make a first assessment of the performance of the selected architecture. Also, as the MP is keeping track of all actions performed by the CP, this creates a side effect in terms of information the latter has to exchange with the management plane. This occurs in particular when the management does not make use of any abstraction of the exact CP information elements processed by the CP. Moreover, given that the CP brings a new degree of flexibility to transport networks, the MP must mirror such flexibility for an operator to be able for leveraging all the CP capabilities, and tuning it to each specific scenario. The high number of options that GMPLS protocols support, along with its different deployment models, increases the need for a configurable MP. This means that during the development of a network management system for next generation transport networks, flexibility to accommodate a variety of deployment models (unified, overlay, augmented) and provisioning approaches (soft permanent or switched connections, etc) is an essential success factor. The management system must not impose how the network is operated, but should be able instead to gracefully adapt itself to different operation, provision and maintenance processes. The CP ability to take autonomous decisions and trigger network changes is another new challenge for management systems of transport networks, which previously had to deal with dumb network devices that were fully commanded from the centralized system. CP-MP integration suffers as a result, since there are much more interaction cases, of no trivial complexity, that need to be tested and validated. Interoperability aspects should not be overlooked during the initial development phases, tests should be conducted as early as possible and an appropriate integration phase should be planned.

V. Conclusion

The TBONES project and experiments conducted using the developed platform have lead to significant improvements both in terms of GMPLS CP capabilities for multi-layer packet optical networks and GMPLS CP implementation. This project has among others implemented the foreseen performance advantages and capabilities of a distributed CP for packet-optical environments. Even if several improvements can still be provided, the performance, scalability and flexibility achieved when using a GMPLS distributed CP are significant enough to position this approach as suitable for controlling packet-optical environments. Indeed, one of the main results is the need to improve the GMPLS CP capabilities for multi-layer networks in order to attain the full performance and efficiency gain. Another substantial result obtained from running a distributed CP is the essential need for delivering a cooperative policy-based management well integrated with the CP software architecture. This can be indeed achieved through the use of a dedicated, stand-alone policy-agent controller as part of the GMPLS CP software architecture.

The TBONES project has also demonstrated that the use of a fully IETF compliant GMPLS CP protocol suite delivers the most suitable response to the operational concerns in terms of evolution capability. As such, the TBONES project has demonstrated that starting from a GMPLS implementation allows to easily proceed with any CP interconnection model (from overlay to a unified model) and progressive deployment. Therefore, the GMPLS CP protocol suite as defined by the IETF delivers the most future safe solution to operators willing to deploy a distributed CP. By providing for upgradeable and tighter CP interactions capabilities and collaborative mechanisms, the GMPLS CP protocol suite allows operators and carriers to better tackle the challenges of optimally integrating their packet and optical environments.

Acknowledgment

This work was carried out within the framework of the TBONES project sponsored by the ITEA. Part of this work has been supported by the Flemish government for Innovation through Science and Technology in Flanders (IWT).

References

[1] E.Mannie (Editor) et al., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3945, October 2004.

[2] G.Swallow et al., "Generalize Multi-Protocol Label Switching (GMPLS) User-Network Interface (UNI): RSVP-TE Support for the Overlay Model," RFC 4208, October 2005.

[3] K.Kompella et al., "Routing Extensions in Support of Generalized Multi-Protocol Label Switching", RFC 4202, October 2005.
K.Kompella et al., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching", RFC 4203, October 2005.

[4] J.Moy, "OSPF Version 2", RFC 2328, April 1998.
R.Coltun, "The OSPF Opaque LSA Option", RFC 2370, July 1998.
D.Katz et al., "Traffic Engineering (TE) Extensions to OSPF Version 2, RFC 3630, September 2003.

[5] R.Braden, et al., "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.
L.Berger, et al., "RSVP Refresh Overhead Reduction Extensions", RFC 2961, April 2001.
D.Awduche, et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[6] L.Berger (Editor), "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, January 2003.
L.Berger (Editor), "Generalized Multi-Protocol Label Switching (GMPLS) Signaling - RSVP-TE Extensions", RFC 3473, January 2003.

[7] A.Farrel et al., "Crankback Signaling Extensions for MPLS Signaling", draft-ietf-ccamp-crankback-05.txt (work in progress), May 2005.

[8] S.Herzog, "RSVP Extensions for Policy Control", RFC 2750, January 2005.

[9] Y.Rekhter, J.Lang and D.Papadimitriou, " RSVP-TE Extensions in support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery", draft-ietf-ccamp-gmpls-recovery-e2e-signaling-03.txt (work in progress), April 2005.

[10] B.Moore (Editor) et al., "Policy Core Information Model (PCIM) Extensions", RFC 3460, January 2003.

[11] http://www.ibcn.intec.ugent.be/css_design/research/projects/IST_FP5/NRS/index.html

[12] J. Derkacet al.: IP/OTN Cost Model and Photonic Equipment Forecast – IST Lion Project, Proc. of Workshop on Telecommunication Techno-Economics, (Rennes, France, May 2002), pp. 126-138.

[13] B. Berde, et al. "Improving Network Performance through Policy-based Management applied to Generalized Multi-Protocol Label Switching", ISCC 2005, Tenth IEEE Symposium on Computers and Communications, June 2005.


www.icc2006.org
IEEE International Conference on Communications
11-15 June 2006
IEEE Catalog Number: 06CH37799C
ISBN : 1-4244-0355-3
Library of Congress : 81-649547
IEEE
IEEE COMMUNICATIONS SOCIETY
ICC 2006
ISTANBUL
EUROPE
ASIA
Bridging Continents Through Communications
Copyright © 2006 IEEE