A GMPLS Unified Control Plane for Multi-Area Networks

ITEA TBONES Project

• •
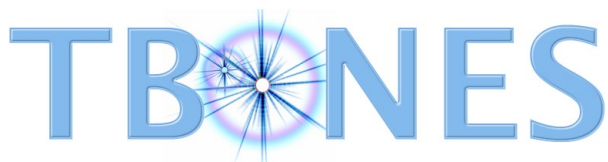
TBONES

ITEA

TBONES: a GMPLS Unified Control Plane for Multi-Area Networks

Dimitri Papadimitriou[1], Alcatel Bell (Antwerpen, Belgium)
Bela Berde, Alcatel CIT (Marcoussis, France)
Remi Theillaud, ATOS Origin (Paris, France)
Koen Casier, University of Ghent - IMEC (Ghent, Belgium)

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 control plane feasibility, performance, and gain when migrating from legacy packet over circuit multi-layer networks driven by overlaid control planes. The ITEA TBONES project aims at tackling both objectives through the development of a platform including network dimensioning and GMPLS control plane elements constituting such networks. This paper presents our methodological approach for the realization of this platform and the capabilities of the TBONES control plane emulator. Several experiments demonstrating its validity and capabilities including its applicability to multi-area networks will be exhibited during the Infocom 2005 demonstration session.**

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 objective of the TBONES control plane emulator is the validation of the control and the dynamic provisioning of multi-layer networks via a distributed and unified control plane based on the GMPLS protocol suite, as defined by the Internet Engineering Task Force (IETF). The TBONES project objectives also include the validation of the migration from an overlay (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 hosting more than one data plane layer. Besides the verification of the proper operation of the GMPLS protocol suite in multi-area networks and Traffic Engineering (TE) algorithms, this project encompasses the quantification of the performance (in terms of resource and speed) of mechanisms such as constraint-based routing, and recovery (i.e. pre-planned and dynamic re-routing). Control plane interactions between the optical and the packet network (with a specific focus on IP/MPLS clients) are investigated. Finally, the TBONES emulator interfaces with the Dimensioning Tool (DT) external entity that calculates an adequate dimensioning for the topology according to an input traffic matrix.

This paper is organized as follows. In section II. we catalogue the TBONES control plane emulator components and describe their implementation. In section III., we detail the TBONES platform operations. Section IV integrates the experimentation aiming at validating the software development and the project technical objectives. Finally, we list in section V the main conclusions drawn from this work.

II. TBONES CONTROL PLANE EMULATOR

The TBONES Control Plane (CP) emulator is implemented as a set of processes running on Linux 2.6. It emulates the behavior of a set of nodes by instantiating for each node, a lower protocol stack and several control plane controllers. Each protocol stack implements the Open Shortest Path First - Traffic Engineering (OSPF-TE) [2] and the Resource Reservation Protocol - Traffic Engineering (RSVP-TE) [3] protocols, and runs in its own process. Each control plane controller set also runs in its own process, they communicate with each other through the protocol stacks. Each control plane controller consists of a set of modules: the Node Emulator (NE), Signaling Controller (SIGC), the TE Controller (TEC) and the Path Computation Controller (PCC).

The SIGC processes the trigger GMPLS RSVP(-TE) [4] signaling messages received from peering controllers. This controller is in charge of the Packet (PSC) and Lambda (LSC) Label Switched Paths (LSP) setup and release, and interacts for this purpose with the TEC. The following signaling procedures are supported: bi-directional LSPs, make-before-break, crankback, failure notification, dynamic and pre-planned re-routing, Soft-Permanent Connections (SPCs) and explicit label control. The SIGC relies on a Signaling Development Kit (SDK). The latter supports the communication with the lower protocol stack and its RSVP(-TE) component, and maintains a database of all LSPs known by this signaling controller.

The TEC processes signaling information such as explicit, record and exclude routes, or constraints (suggested label, label sets, etc.) to choose the component TE link for each LSP. The TEC relies on a generic TE Development Kit (TEDK) that supports the communication with the lower protocol stack and its OSPF(-TE) component, and maintains a database of all TE links (bundles and component links) advertised through OSPF(-TE). This database is the main input to the PCC, other inputs include crankback-related information [5] and signaled exclude route [6]. The OSPF(-TE) component maintains the routing adjacencies with peer nodes and the flooding of OSPF(-TE) Link State Advertisements (LSA) including global and per-interface mechanisms to limit the bandwidth consumed by such a flooding. The TEC updates information, e.g., per-priority Maximum LSP bandwidth advertised by GMPLS OSPF(-TE) [7], for the TE link(s) used by an LSP. It interacts with the NE for the reservation and allocation of local resources, and therefore for label allocation. The following TE procedures are supported:

---

[1] Corresponding Author: Alcatel Bell, Fr. Wellesplein, 1, B-2018, Antwerpen, Belgium - E-mail: dimitri.papadimitriou@alcatel.be

- *Multi-area Traffic Engineered (TE) LSP signaling* (using loose explicit routing)
- *PSC LSP over LSC Forwarding Adjacencies (FA) LSP*: the TEC can trigger the setup of additional LSC FA LSPs to fulfill new packet LSP requests; initial FA LSPs setup follow the topology computed by the DT from a given topology and traffic matrix, prior to the initialization of the simulation. The other LSC FA LSPs are dynamically triggered and setup on demand as needed.

III. TBONES PLATFORM OPERATIONS

The different modules of the TBONES platform as well as the inputs/outputs and the information flows are depicted in Fig. 1. A network topology is used as input by the data and control plane modules. The traffic matrix is used by the DT that calculates an adequate routing and dimensioning for the network topology according to this matrix. The resulting output can be loaded by the control plane. The data plane gives to the control plane the scheduled LSC LSP requests, which are deduced from the traffic matrix. The TBONES emulator is implemented as a set of Linux processes that imitate a set of nodes by instantiating, for each node, a protocol stack and the set of GMPLS controllers. The former process provides a protocol stack that includes OSPF(-TE) and RSVP(-TE), and an IP stack to forward messages across the simulated IPv4 control channels. The protocol stacks exchange RSVP(-TE) and OSPF(-TE) packets through a process that emulates point-to-point sub-networks (software loopbacks). Moreover, each protocol stack may access to an Ethernet interface to communicate with the peering emulator(s). The protocol stack(s) attached to an Ethernet interface behaves as an IP router compared to the other protocol stacks. The second process, running on top of the protocol stack, implements the GMPLS controllers and a command engine agent that handles the communication with the emulator command engine.
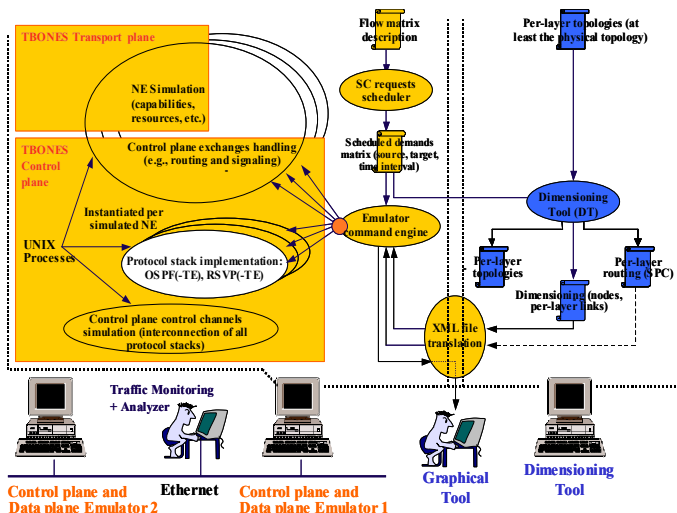

Figure 1.   TBONES Information Flow and Processing

A protocol stack process never performs any blocking operation and is single-threaded. On the other hand, each control plane process is multi-threaded, one thread being used for each event: signaling messages, routing updates, and internal scheduler events (trigger of end-to-end LSP setup and teardown). Once a thread wakes up, it may run any of the controllers. The control plane emulator provides also a GUI command-engine, which allows its user to interactively query the different emulated nodes (dump of the signaling and traffic engineering databases; retrieval of statistics), and to generate data plane failures. The communication with the command-engine uses local sockets. Thus, there are three event sources: routing updates from the TEDK; signaling messages exchanged with peer signaling controller and received through the SDK, and commands from the emulator command engine.

IV. TESTBED AND EXPERIMENTS

The TBONES control pane emulator provides some minimal data plane modeling as required for experimental support. The testbed includes, as depicted in Fig.1, two emulators hosted by two servers (running on Linux 2.6) that are interconnected by an Ethernet LAN segment. A special port is configured on the LAN switch to analyze the traffic flowing through this segment and graphically represent the OSPF routing adjacencies between nodes as well as the LSPs setup using GMPLS RSVP(-TE) signaling. A dedicated host system runs the DT and exchanges output results with the emulator using XML files. This system also provides graphical representation of the LSPs established in the network topology.

Several experiments aiming validation of the TBONES emulator implementation will be exhibited during the Infocom 2005 demo session. These experiments mainly include:
- The TBONES software validity including OSPF(-TE), RSVP(-TE) stacks and the different GMPLS controllers.
- The TBONES software supported load and performance (i.e. benchmarking) including OSPF(-TE), RSVP(-TE) stacks and the different GMPLS controllers. For instance, OSPF(-TE) implementation performance implies:
  (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 Fig. 2. The backbone Area 0, the Area 6, 66 and 77 belong to the same Autonomous System (AS), as depicted in Fig. 2. 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, 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 LSA (Opaque Type 1) are exchanged within each area to describe the TE attributes of their internal links (in particular, the links interconnecting the Area 0 ABRs). The 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.
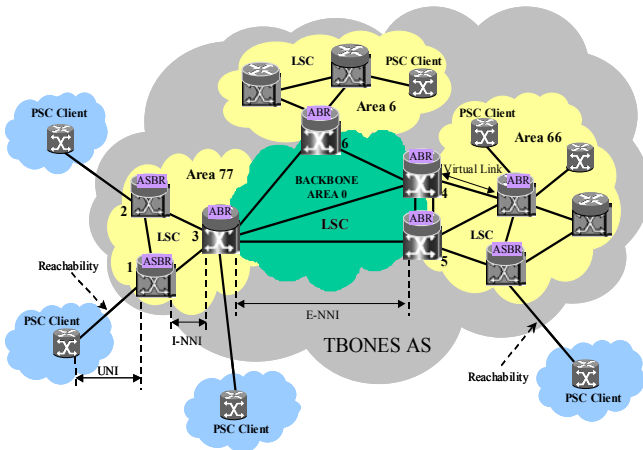

Figure 2.   TBONES Multi-Area Routing Topology

- Pre-planned and dynamic end-to-end LSP re-routing. The former implies that the protecting LSP resources are allocated at the control plane level only and explicit action is required to activate (i.e. commit resource allocation at the data plane) during the recovery phase. Dynamic re-routing switches traffic to an alternate LSP that is fully established only after failure occurrence. The new alternate route is selected at the LSP head-end node, it may reuse resources of the failed LSP at intermediate nodes and may include additional intermediate nodes and/or links.
- The collaboration between the emulator and DT. This involves validation of the interface between the emulator and the DT and the capability to transparently exchange topological attributes through the dimensioning tool.
- The validation of the migration from an overlay towards a unified control plane interconnection model. For this purpose, the GMPLS-compliant User Network Interface (UNI) [8] for the overlay model is used for comparative purposes. In this model, no routing adjacencies are established between network edge and client nodes, but only between peering client nodes using the server layer LSP for the client routing adjacency establishment. Performance results are compared with those obtained with respect to the target models of the TBONES routing topology: 1) *Augmented model*: routing adjacencies between network edge and client nodes are used to exchange reachability information only. Depending on the addressing space two cases can be considered: separate control plane addressing space (between the client and the network): and common control plane addressing space: the control plane shares its address space with the network (at least its edges). 2)

Unified model: routing adjacencies between network edge and client nodes are used to exchange reachability, topology and TE information.

It is also the objective of the TBONES project to assess the scalability (in terms of network size, traffic throughput and variations, as well as failures) of a distributed GMPLS control plane for PSC + LSC multi-layer networks. This experiment includes the evaluation of how multi-layer provisioning in a network undergoing traffic variations can reduce the LSP request blocking. For this purpose, a scheduled demand matrix is provided to the control plane, that triggers the setup and teardown of LSPs. Newly setup LSPs will accommodate increasing traffic demands, while LSP teardown happens for decreasing (or otherwise changing) traffic patterns to free capacity that can be used in other parts of the network. There exist multiple approaches to decide on the triggering of LSP establishment that translates a scheduled demand into a logical topology. The demand itself is presented as a collection of LSPs to be aggregated in this configurable logical topology.

V.   CONCLUSION

The TBONES project aims at demonstrating the feasibility of a unified control plane using the innovative GMPLS protocol suite and mechanisms. Its complete validation (in terms of compliance and interoperability) and performance assessment experiments (benchmarking) are ongoing that will conclude a first development phase. This project also aims at demonstrating the relevance, the scalability and the gain obtained from the deployment of a unified control plane for multi-layer networks. Further experiments are currently conducted to validate different scenarios ranging from cost analysis of grooming strategies to the migration from overlay to unified control plane interconnection models.

REFERENCES

[1]   E.Mannie (Editor) et al., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3945, October 2004.
[2]   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.
[3]   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.
[4]   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.
[5]   K.Kompella et al., "Routing Extensions in Support of Generalized Multi-Protocol Label Switching", draft-ietf-ccamp-gmpls-routing-09.txt (work in progress), October 2003.
        K.Kompella et al., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching", draft-ietf-ccamp-ospf-gmpls-extensions-12.txt (work in progress), October 2003.
[6]   A.Farrel et al., "Crankback Signaling Extensions for MPLS Signaling", draft-ietf-ccamp-crankback-03.txt (work in progress),October 2004.
[7]   C.Y. Lee et al., " Exclude Routes - Extension to RSVP-TE", draft-ietf-ccamp-rsvp-te-exclude-route-02.txt (work in progress), July 2004.
[8]   G.Swallow et al., "GMPLS User-Network Interface (UNI): RSVP-TE Support for the Overlay Model," draft-ietf-ccamp-gmpls-overlay-05.txt (work in progress), October 2004.

ABOUT **TBONES**

The objectives of the project (started in May 2003) require the federation of a variety of expertise. The Europe-wide consortium regroups partners from four countries with different status, complementary experiences, and know-how.

The two-year project is a collaboration of Alcatel Research&Innovation (France), Atos Origin (France), Alcatel Bell (Belgium), University of Ghent – Interuniversity MicroElectronics Centre (IMEC) (Belgium), Telefonica I+D (Spain), and Centre Tecnològic de Telecomunicacions de Catalunya (CTTC) (Spain).

The intellectual merit of the project includes greatly expanding the knowledge and technologies relating to cost-efficient, highly flexible, revenue generating optical transport networks that are driven by a GMPLS control plane.

The project duration is expected to be two years. At the end of the second year, TBONES project will demonstrate a capability set of core GMPLS driven dynamic provisioning.

ABOUT **ITEA**

ITEA – Information Technology for European Advancement – is Europe's premier cooperative program for pre-competitive research and development in embedded and distributed software. It was launched in 1999 as an eight-year EUREKA strategic cluster program to stimulate and support projects that will give European industry a leading edge in software-intensive systems.

EUREKA is a pan-European network for market-oriented, industrial R&D. It was created as an intergovernmental Initiative in 1985. EUREKA aims to enhance European competitiveness through its support to businesses, research centers and universities who carry out pan-European projects to develop innovative products, processes and services.

The internationally recognized EUREKA label adds value to a project and gives participants a competitive edge in their dealings with financial, technical and commercial partners.

ITEA supports coordinated national funding submissions within the EUREKA framework, issues annual Calls for Projects, evaluates projects and helps bring research partners together.

ITEA-labeled projects build vital middleware and prepare standards, laying the foundations for the next generation of products, systems, appliances and services. ITEA projects are industry-driven, involving complementary R&D from at least two companies in two countries (EUREKA rules). The program is open to partners from large industrial companies, Small and Medium-sized Enterprises (SMEs), as well as public research institutes and universities.

The founding companies, which include Alcatel, Barco, Bosch, Bull, DaimlerChrysler, Italtel, Nokia, Philips Electronics, Siemens, Thales, and Thomson, were later joined by the European Federation of high-tech SMEs. These firms are all leaders in their field, whether in automotive, telecommunications, mobile communication, consumer electronics or information technology.

ITEA TBONES Project

Project Leader: Mr Bela Berde
Alcatel CIT
Route de Nozay
91460 Marcoussis
FRANCE

Tel: +33 1 6963 4731
Email: bela.berde@alcatel.fr
Web: www.tbones.be

ITEA Office

Laplace Building 0.04
PO Box 513
5600 MB Eindhoven
THE NETHERLANDS

Tel: +31 40 247 5590
Fax: +31 40 247 5595
Email: itea@itea-office.org
Web: www.itea-office.org

TBONES

ITEA
INFORMATION TECHNOLOGY
FOR EUROPEAN ADVANCEMENT