Paper

The Realization of NGN Architecture for ASON/GMPLS Network

Sylwester Kaczmarek, Magdalena Młynarczuk, Marcin Narloch, and Maciej Sac

Department of Teleinformation Networks, Gdańsk University of Technology, Gdańsk, Poland

Abstract—For the last decades huge efforts of telecommunication, Internet and media organizations have been focusing on creating standards and implementing one common network delivering multimedia services – Next Generation Network. One of the technologies which are very likely to be used in NGN transport layer is ASON/GMPLS optical network. The implementation of ASON/GMPLS technology using open source software and its results are the subject of this paper. The ASON/GMPLS architecture and its relation to the proposed ITU-T NGN architecture are described. The concept, functional structure and communication among architecture elements as well as the implementation of laboratory testbed are presented. The results of functional tests confirming proper software and testbed operation are stated.

Keywords—*ASON, Connection Control Server, Diameter, GMPLS, IP QoS, NGN, RSVP.*

1. Introduction

The changes that take place in the area of modern community indicate the great value of information. The information has various areas of applications and forms of presentation. Constant information growth and the need for fast availability to the whole public in direct or processed form generate necessity of new telecommunication network architecture proposition. For this reason networks have to be developed to meet the needs of new requirements. The next generation network (NGN) is a proposition of architecture which has a chance to fulfill society requirements.

Standardization of the NGN dates back to the NGN workshop held in 2003 by ITU-T (International Telecommunication Union – Telecommunications) standardization group. The Y.2000 series of recommendations has been given for the NGN specification and requirements. Functional requirements and architecture of next generation networks are described in Recommendation Y.2012 [1]. Conceptually, the NGN architecture consists of service stratum and transport stratum. The transport stratum provides the IP connectivity services to NGN users. The service stratum provides the service control and content delivery functions. From economic aspect, the conversion of current networks to NGN architecture has to proceed evolutionally because of the high costs associated with realization of this architecture.

The ITU-T automatically switched optical network (ASON) [2] concept and generalized multi-protocol label switching (GMPLS) [3] Internet Engineering Task Force (IETF) solution were combined by Optical Internetworking Forum (OIF) into ASON/GMPLS optical network [4], which is one of the most promising solutions for NGN transport layer. Key issue for the ASON/GMPLS proposition is the provision of effective network control servers. An implementation of ASON/GMPLS connection control layer is presented in this paper.

The aim of our work was a realization of selected functionality of ASON/GMPLS network [5]. We decided to carry out the tasks in two stages. The first step was to write software for network elements with respect to ASON/GMPLS standardization and the latest trends in ITU-T NGN architecture. Secondly, implemented software had to be tested in laboratory testbed.

The paper is organized as follows. In Section 2 ASON/GMPLS control plane in context of the NGN architecture is presented. An ASON/GMPLS control plane implementation concept is introduced in Section 3. The realization of ASON/GMPLS architecture, functional architecture of connection control server (CCS) and service control server (SCS) as well as resource terminal (RT) representing transport resources are described in Section 4. The results of functionality and operation tests are reported in Section 5. Conclusions and outlook to future are presented in Section 6.

2. ASON/GMPLS Architecture

The automatically switched optical network (ASON) was proposed by ITU-T and described in Recommendation G.8080 [2]. In fact ASON is only a concept of architecture. It does not specify all protocol details necessary to implement the control plane solution. Optical Internetworking Forum (OIF), a group of international network service providers, made an effort to apply GMPLS protocols to ASON architecture [4]. This solution, an ASON control plane built on GMPLS protocols is known as ASON/GMPLS.

Due to the fact that ASON/GMPLS is a highly complicated architecture, in this section the main aspects regarding the ASON/GMPLS control plane functionality are presented. The main task of the ASON/GMPLS control plane

is to facilitate fast and efficient configuration of connections within the transport layer network to support both switched and soft permanent connections. It consists of different components providing specific functions (including routing and signalling). The interactions between and within domains are defined in terms of reference points: UNI, E-NNI, I-NNI [2].

ITU-T standardization group recommended control plane components like: routing controller (RC), protocol controller (PC), connection controller (CC), link resource manager (LRM), termination and adaptation performer (TAP), calling/called party call controller (CCC), network call controller (NCC). These components can be combined in different ways depending on the required functionality.

The architecture of ASON/GMPLS network is presented in Fig. 1. The connection controller is responsible for coordination among the link resource manager, routing controller and other connection controllers for the purpose of setup, release and modification of connection parameters [2]. For this reason the CC components utilize a connection controller interface (CCI) to the transport plane. As stated in [2] the routing controller is an abstract entity that provides routing functions. The link resource manager maintains the network topology. The role of protocol controller is to map the operation of the components in the control plane into messages that are carried by communication protocols between interfaces in the control plane. The termination and adaptation performer holds the identifiers of resources that can be managed using the control plane interfaces. Call components are concerned with call service and implemented in service control server. The main role of the calling/called party call controller is generation of outgoing call request and acceptance or rejection of incoming call request.
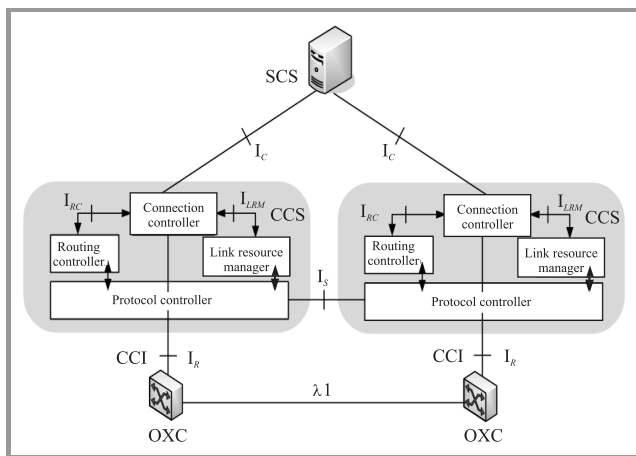

Fig. 1. ASON/GMPLS network architecture. SCS – service control server, CCS – connection control server, OXC – optical cross-connect.

As it has been already mentioned, the ASON/GMPLS architecture is one of the solutions considered as NGN transport layer. ITU-T NGN resource and admission control function (RACF) [1] performs operations similar to

ASON/GMPLS connection control server consisting of connection controller, routing controller, LRM and protocol controller elements. Selected functionality of ITU NGN service control functions is performed by service control server in the ASON/GMPLS architecture. NGN transport functions correspond to resource elements depicted as OXC in Fig. 1. The concept of ASON/GMPLS architecture implementation is presented in the next section.

3. Concept of Implementation

The concept of the proposed ASON/GMPLS implementation is presented in Fig. 2. According to the three-layer architecture depicted in Fig. 1, the implementation includes functionality of service control layer, connection control layer as well as optical resource layer represented by resource terminals (RTs) emulating optical cross-connect operation. Corresponding layers communicate over dedicated interfaces using communication protocols.
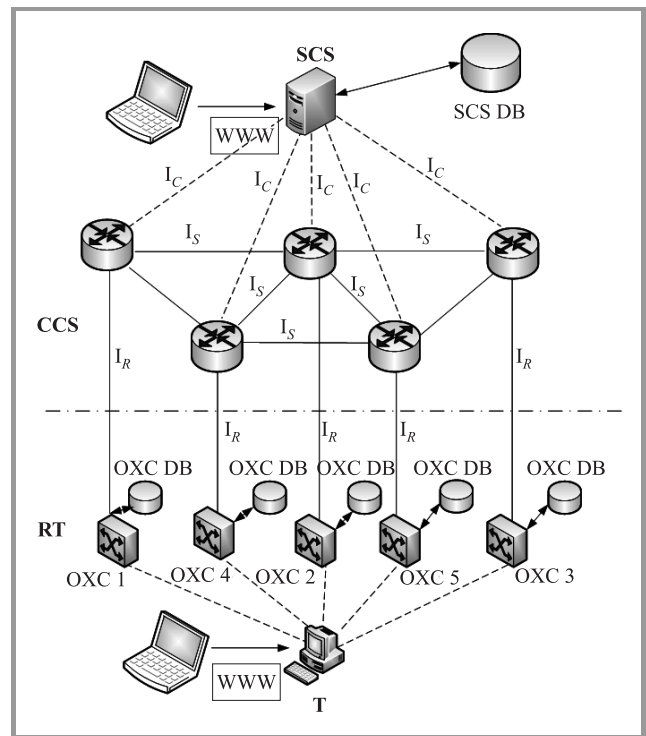

Fig. 2. Concept of ASON/GMPLS implementation. RT – resource terminal, T – terminal.

Service control server (SCS) is responsible for handling of user call and call termination request. Call requests are furtherly transformed into connection requests in the connection control layer. SCS provides www interface and stores all necessary information in local database. The database in service control server consists of the following tables: CTRL_TABLE, CALL_STATE and CALL_STAT. CTRL_TABLE table maps addresses from resource layer into addresses of corresponding connection control servers (CCSs). The CALL_STATE table stores the state of processed requests. The CALL_STAT table gather statisti-

cal data regarding performance of handling requests in the system. Particularly, duration of operations regarding call setup and termination requests in the system is registered. Connection in optical layer is established only if there exist enough free resources to allocate. Each resource terminal informs corresponding connection control server about the result of optical resource allocation.

Each connection control server is in charge of dynamic management of optical resources in transport layer by processing requests for establishing (setting) and releasing (deleting) paths. Connection control servers utilize RSVP [6] protocol extended to transport objects regarding resource reservation in optical layer. The design of CCS functionality was based on the following assumptions:

– mapping of elements from control layer to transport layer is one-to-one,

– single reservation session results in reservation of one or more transport units, depending on the bandwidth demand request,

– identifiers of resource layer are transported using mechanisms of LMP protocol [7],

– fixed filter (FF) reservations style is applied [6].

Resource terminals emulate optical resources. For this reason they maintain information regarding state of the emulated device in a local database. Terminal T is used for configuration and verification of reservation state in emulated OXCs. In order to perform these operations dedicated www interface which presents database content of resource terminals is provided.

Communication between cooperating layers is performed over well defined interfaces. Service control layer and connection control layer communicate over I_C interface. Information between connection control server and resource terminal is exchanged over I_R interface. In both cases Diameter protocol [8] is used. In the next section realization of the testbed architecture based on the presented concept is described.

4. Architecture Realization

In this section realization of the proposed ASON/GMPLS architecture concept (Fig. 2) is described. The implementation of ASON/GMPLS network elements limited to selected functionality was based on Linux platform. The realization of the software was performed in two phases (variants). The first step was variant I regarding implementation of connection control layer with basic functionality and the final step was variant II regarding implementation of connection control layer with ability to communicate with surrounding layers.

Functional structure of the implemented ASON/GMPLS software (variant II) is presented in Fig. 3. Service control server (SCS) handling user requests can be managed through www browser, which communicates with Apache

HTTP embedded server. The www server uses PHP hypertext preprocessor compiled to shared object library libphp5.so loaded as a module at the beginning of its initialization. SCS server configuration and description of connection control layer are stored in local Oracle database. Communication between database and www server uses SQL queries. PHP OCI8 packet provides functions to communicate with the database. That allows to update the database content by WWW/PHP scripts of user interface.
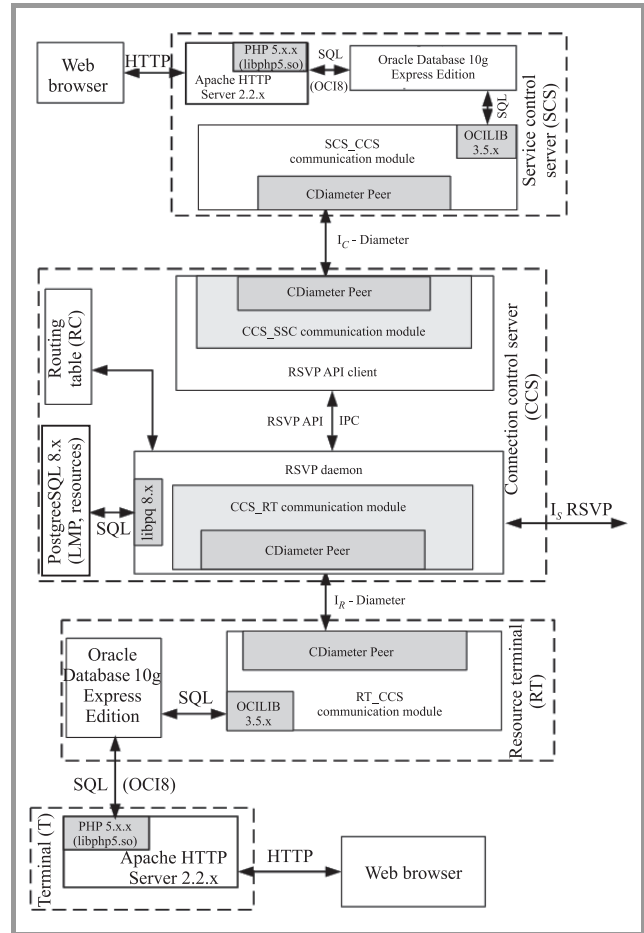

Fig. 3. ASON/GMPLS network realization.

The most important part of service control server (SCS) is the communication module responsible for exchanging information between SCS and CCS. In the communication module OCILIB library is used to provide communication with database including database change notification (DCN) mechanism. DCN allows to asynchronously notify the communication module of the SCS about changes in the database generated by the user through the WWW/PHP interface. Moreover, OCLIB allows to modify the content of the database according to the results of user request processing in connection control servers and resource terminals. Requests results, particularly regarding processing performance in the architecture can be statistically analyzed in SCS and presented by the WWW/PHP interface. Furthermore, service control server communicates with connection control layer using Diameter protocol. In the re-

alization of SCS an open source implementation CDiameter Peer was used. Thus, communication module of the SCS has the functionality of Diameter Peer, which allows to provide connection control layer with the following parameters of user request: Call-ID call identifier [9], source and destination address in connection control and resource layers, bandwidth demand, reservation priority. The above mentioned request parameters are carried in Diameter messages as appropriate attribute value pairs (AVP) elements. Detailed description of Diameter application used in the architecture is described in the last part of the section.

The core component of the proposed and implemented ASON/GMPLS architecture is connection control layer. The following ASON/GMPLS functionalities: connection controller, protocol controller, routing controller and link resource manager were implemented in each connection control server. The functionality of protocol controller and connection controller is performed by the RSVP daemon element, which is the part of KOM RSVP implementation [10] of RSVP protocol. Functionality of KOM RSVP project was appropriately extended to transport information for control of optical network and to communicate with service control layer as well as resource terminal layer [11], [12]. Thus, some limited functionality of RSVP-TE [13] signalling protocol was achieved. Extension of KOM RSVP preserved original structure of KOM RSVP implementation, with division into RSVP daemon and RSVP API client. The role of API client is to invoke API functions provided by RSVP daemon with respect to Diameter messages received from service control server in order to control the process of setting-up and releasing connections in the transport layer. Apart from RSVP API functions, due to distributed nature of CCS processing, POSIX signals were also used as another way of interprocess communication (IPC). The main functions of RSVP daemon are to send, receive and process RSVP messages as well as to allocate and release optical resources emulated by resource terminals. Information necessary to affect RTs are carried by Diameter protocol. CDiameter Peer implementation was used in RSVP daemon as well as in RSVP API client. Each resource terminal has its single representation in the server from the connection control layer. The state of OXC resources under the control of connection control server is maintained in resource database which plays the role of LRM ASON component. To identify physical resources for each RSVP session some additional information was introduced to KOM RSVP code including resource ID in PSB and RSB blocks interchangeably defining physical resource in the transport plane. A functionality of Call-ID object representing call identifier according to [9] was also added. RC ASON element is based on the routing table in the operating system kernel and managed by iproute package. For resources discovery purpose link management protocol (LMP) procedures are implemented. All information necessary for proper operation of CCS server are stored in local PostgreSQL database which is accessible thanks to the mechanisms provided by libpq library.

Resource terminal is responsible for emulation of optical cross-connect (OXC) device. Resources of the emulated OXC are mapped to the content of the Oracle Database 10g Express Edition database. The state of the resources can be checked and each OXC can be configured through www interface at any time. That operation is possible through terminal T with Apache www server and PHP hypertext preprocessor which generates SQL queries to databases in resource terminals. Similarly to service control server PHP OCI8 packet is used. The main part of resource terminal is the communication module exchanging information with connection control server. Like in SCS, in resource terminal OCILIB library provides database communication mechanisms allowing to retrieve and modify the content of the Oracle database. In resource terminal CDiameter Peer is also utilized to provide communication with connection control server using Diameter protocol. Diameter messages at I_R interface transport parameters necessary to establish and release connection in the transport layer.

Service control server as well as resource terminal communicate with connection control server using Diameter protocol [8]. For this reason a new Diameter application PBZ_App and a new Diameter vendor identifier PBZ_vendor_id have been defined. Moreover, new types of Diameter messages and AVPs have been proposed. Diameter messages sent between service control server and connection control server (I_C interface) are listed in Table 1. Messages exchanged between resource terminal and connection control server (I_R interface) are presented in Table 2. Both tables contain full description of used messages and their contents. According to Diameter specification [8], all requests and corresponding answers have the same command codes. The type of the message (request or answer) is determined by "R" flag in header field, which is set to 1 for requests.

Message flow for connection creation scenario in the realized ASON/GMPLS architecture is presented in Fig. 4. Arrows with numbers correspond to consecutive stages of exchanging information. Particular messages have the following meaning:

1. Asynchronous DCN notification of changes in the CALL_STATE table of Oracle database. Changes were caused by generating request for connection creation through www/php interface.

2. Diameter CCR (Connection-Create-Request) message.

3. Execution of RSVP API createSender() function.

4. RSVP PATH message to RSVP daemon of the next connection control server (according to the routing table).

5. RSVP RESV message in case of successful resource reservation in the remaining connection control servers on the path or RSVP PATH ERROR message otherwise. Steps 6 and 7 are skipped when receiving PATH ERROR message.

Table 1
Diameter messages exchanged over I_C interface

Abbr. name	Full command name and meaning	Code	Direction	Carried parameters
CCR	Connection-Create-Request (*Request for creating connection*)	10000	SCS→CCS	– call identifier (Call-ID) – IP address of source connection control server – IP address of destination connection control server – IP address of source resource terminal – IP address of destination resource terminal – bandwidth – priority
CCA	Connection-Create-Answer (*Answer to CCR message*)	10000	CCS→SCS	– call identifier – result of connection creation
CDR	Connection-Delete-Request (*Request for terminating connection*)	10001	SCS→CCS	– call identifier
CDA	Connection-Delete-Answer (*Answer to CDR message*)	10001	CCS→SCS	– call identifier – result of connection termination

Table 2
Diameter messages exchanged over I_R interface

Abbr. name	Full command name and meaning	Code	Direction	Carried parameters
RCR	Resource-Create-Request (*Request for resource allocation – allocation of optical transport units*)	20000	CCS→RT	– call identifier – identifiers of incoming transport units – identifiers of outgoing transport units
RCA	Resource-Create-Answer (*Answer to RCR message*)	20000	RT→CCS	– call identifier – result of resource allocation
RDR	Resource-Delete-Request (*Request for resource release – release of optical transport units*)	20001	CCS→RT	– call identifier – identifiers of incoming transport units – identifiers of outgoing transport units
RDA	Resource-Delete-Answer (*Answer to RDR message*)	20001	RT→CCS	– call identifier – result of resource release

6. Diameter RCR (Resource-Create-Request) message.

7. Diameter RCA (Resource-Create-Answer) message.

8. POSIX signal of number 50 in case of successful resource reservation in all connection control servers (including source connection control server) or POSIX signal of number 51 otherwise.

9. Diameter CCA (Connection-Create-Answer) message.

10. Update of connection state in Oracle database.

Message flow for connection termination scenario in the realized ASON/GMPLS architecture is presented in Fig. 5. Arrows with numbers correspond to consecutive stages of exchanging information. Particular messages have the following meaning:

1. Asynchronous DCN notification of changes in the CALL_STATE table of Oracle database. Changes were caused by generating request for connection termination through www/php interface.

2. Diameter CDR (Connection-Delete-Request) message.

3. Execution of RSVP API releaseSession() function.

4. RSVP PATH TEAR message to RSVP daemon of the next connection control server (according to the routing table). Simultaneously Diameter RDR (Resource-Delete-Request) message to the corresponding resource terminal is sent.
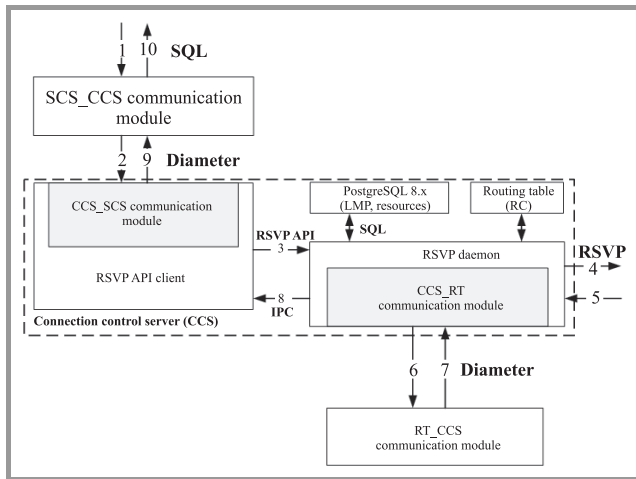
Fig. 4. Connection creation scenario in the realized ASON/GMPLS network.
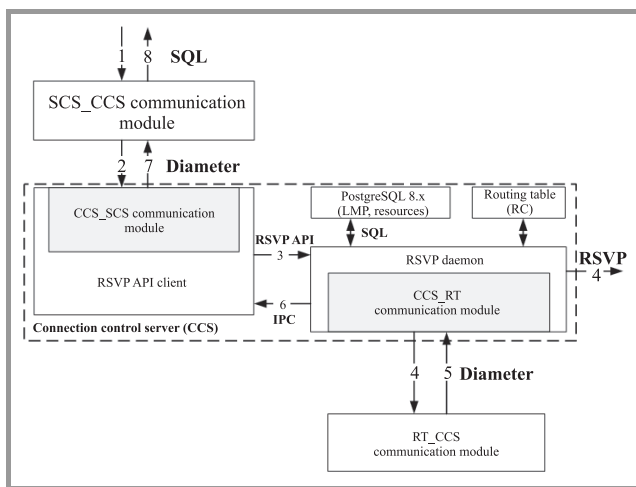

Fig. 5. Connection termination scenario in the realized ASON/GMPLS network.

5. Diameter RDA (Resource-Delete-Answer) message.

6. POSIX signal of number 52.

7. Diameter CDA (Connection-Delete-Answer) message.

8. Update of connection state in Oracle database.

5. Results and Tests

The implemented software for the ASON/GMPLS network elements has been installed and validated in laboratory testbed. Testing system was created to study not only the basic functionality, but also to investigate communication of connection control layer with the whole architecture. The structure and configuration of the testbed is described in Subsection 5.1. The results of performed functional tests are presented in Subsection 5.2.

5.1. Testbed Architecture and Configuration

Network architecture from Fig. 2 has been implemented for the purpose of ASON/GMPLS software testing. Connection control server software has been installed on the NTT TYTAN computers with the following hardware parameters:

– Supermicro X8DTL-3F motherboard,

– Intel XEON E5506 (2,13 GHz) quad core processor,

– 4GB DDR3 ECC R RAM memory,

– 2x500GB SATA HDD.

Resource terminal software has been installed on NTT computers with the following hardware parameters:

– Gigabyte GA G31M-ES2L motherboard,

– Celeron E3300 (2,5GHz) dual core processor,

– 2GB DDR2 DIMM memory,

– 250GB SATA HDD.

The architecture and the configuration of the realized ASON/GMPLS testbed (along with IP addresses of all network equipments) are described in Fig. 6. Execution of
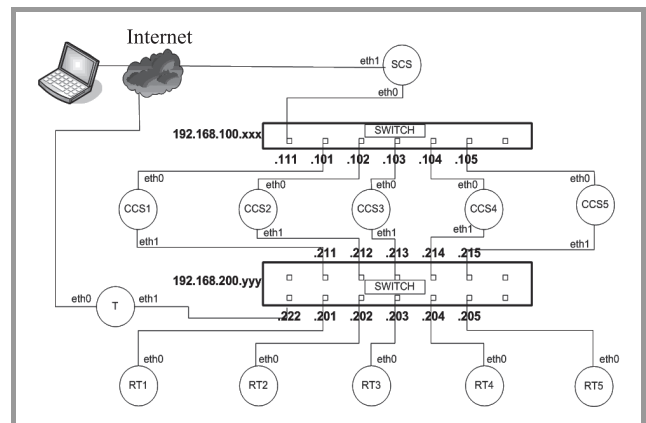

Fig. 6. Architecture and configuration of the implemented ASON/GMPLS testbed.
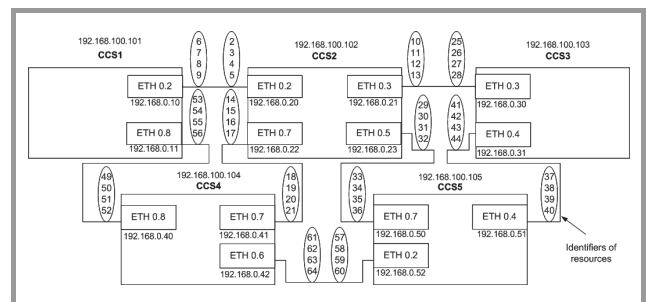

Fig. 7. Architecture of the implemented ASON/GMPLS connection control layer. ETH *x.y* stands for virtual network interface *y* based on physical network interface eth*x*.

all functional tests has been preceded by proper configuration of Debian Linux operating system and implemented software on all computers in the testbed.

Due to limited number of physical network interfaces in hardware platform, implementation of the connection control layer from Fig. 2 required configuration of virtual interfaces. The structure of the used virtual local area networks (VLANs) and the sets of optical resources identifiers assigned to particular interfaces are presented in Fig. 7.

5.2. Functional Tests

In order to verify the implemented and installed ASON/GMPLS software a set of functional test has been executed. Performed test scenarios included validation of:

– handling of single connection creation request, including the situation when there is not sufficient amount of resources to fulfill the request,

– handling of single connection termination request,

– handling of multiple connection creation requests,

– handling of connection termination request in case there are other established connections.

During execution of the scenarios the following aspects of the implemented ASON/GMPLS network operation have been checked:

– communication between service control server and connection control server using Diameter protocol,

– communication between API client and RSVP daemon,

– RSVP daemon operation:

– PATH message handling and processing of RSVP objects (particularly RSVP_HOP object carrying optical transport unit identifiers as well as Call-ID object carrying call identifier),

– creating PSB block carrying outgoing optical resource identifiers,

– creating PHopSB block carrying incoming optical resource identifiers,

– handling of RESV message with Call-ID object extension,

– creating RSB block,

– Diameter protocol communication between RSVP daemon and resource terminal (RT) emulating optical resources,

– handling of PATH TEAR message with Call-ID object extension,

– updating local resource database,

– communication with API client in order to confirm resource allocation/release.

All performed test scenarios confirmed correctness of ASON/GMPLS architecture implementation. Results of the selected basic scenarios are described in the next part of the paper.

Handling of single connection creation request. Single connection creation request was generated using WWW/PHP graphical interface of service control server (SCS) (Fig. 8). Bandwidth of optical transport unit (TRU)

Single call request

New call request

IP_ADDRESS_S:
192.168.200.201
IP_ADDRESS_D:
192.168.200.203
BANDWIDTH:
500
PRIORITY (*):
0
Generate

* - optional

Return to "Request generation" menu
Return to main page

Fig. 8. Generation of connection creation request with 500 Mbit/s bandwidth.

was set to 1000 Mbit/s. The request generated in the test scenario concerned creating connection in the transport layer between resource terminals RT1 (emulating OXC of IP address 192.168.200.201) and RT3 (emulating OXC of IP address 192.168.200.203). According to Fig. 7, this connection involved one relay connection control server (CCS2) as well as one relay resource terminal (RT2). Requested bandwidth was 500 Mbit/s.

After generating the request and receiving the response from connection control server, state of the request stored in the service control server database was updated (Fig. 9). In implemented system allocation of optical TRUs is emu-

CALL_ID	IP_ADDRESS_S	IP_ADDRESS_D	BANDWIDTH	STATE	T_REQ_1	T_CONF_23	T_REQ_4	T_CONF_56
42	192.168.200.201	192.168.200.203	500	2	2010-10-26 15:45:42.649267	2010-10-26 15:45:42.967983		

Fig. 9. Content of the CALL_STATE table in the service control server database after successful connection creation.

ID	ID_OXC	ID_PORT	ID_TRU	STATE
1	1	0	1	0
2	1	2	2	1
3	1	2	3	0
4	1	2	4	0
5	1	2	5	0
6	1	3	10	1
7	1	3	11	0
8	1	3	12	0
9	1	3	13	0
10	1	5	29	0
11	1	5	30	0
12	1	5	31	0
13	1	5	32	0
14	1	7	14	0
15	1	7	15	0
16	1	7	16	0
17	1	7	17	0

Fig. 10. Content of the TRU_OXC table for resource terminal RT2 after successful connection creation between CCS1 and CCS3. TRUs with STATE = 1 are allocated.

lated in resource terminals as changes in TRU_OXC tables of local databases. Contents of TRU_OXC table for resource terminal RT2 after successful test scenario execution is presented in Fig. 10. As RT2 is a relay resource terminal, two optical TRUs are allocated (one incoming and one outgoing).

Handling of single connection termination request. Similarly to connection creation, single connection termination request was generated using WWW/PHP graphical interface of service control server (SCS). Connection termination request generated in the test scenario concerned releasing previously allocated (in subsection: Handling of single connection creation request) optical resources for the connection of unique Call-ID value (Fig. 11).
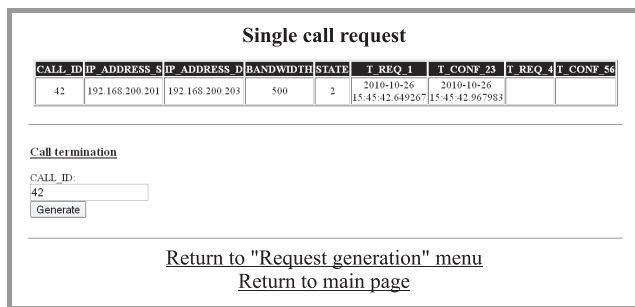
Single call request

CALL_ID	IP_ADDRESS_S	IP_ADDRESS_D	BANDWIDTH	STATE	T_REQ_1	T_CONF_23	T_REQ_4	T_CONF_56
42	192.168.200.201	192.168.200.203	500	2	2010-10-26 15:45:42.649267	2010-10-26 15:45:42.967983		

Call termination

CALL_ID:
42
Generate

Return to "Request generation" menu
Return to main page

Fig. 11. Generation of connection termination request for the connection with Call-ID = 42.

The service control server software identified parameters required to release resources using the Call-ID value carried by the request generated using user interface. Once the request was processed by service control server it was sent using Diameter protocol to the connection control server which had initiated connection creation. After generat-

CALL_ID	IP_ADDRESS_S	IP_ADDRESS_D	BANDWIDTH	STATE	T_REQ_1	T_CONF_23	T_REQ_4	T_CONF_56
42	192.168.200.201	192.168.200.203	500	2	2010-10-26 15:45:42.649267	2010-10-26 15:45:42.967983	2010-10-26 15:49:07.638234	2010-10-26 15:49:07.735563

Fig. 12. Content of the CALL_STATE table in the service control server database after successful connection termination.

ID	ID_OXC	ID_PORT	ID_TRU	STATE
1	1	0	1	0
2	1	2	2	0
3	1	2	3	0
4	1	2	4	0
5	1	2	5	0
6	1	3	10	0
7	1	3	11	0
8	1	3	12	0
9	1	3	13	0
10	1	5	29	0
11	1	5	30	0
12	1	5	31	0
13	1	5	32	0
14	1	7	14	0
15	1	7	15	0
16	1	7	16	0
17	1	7	17	0

Fig. 13. Content of the TRU_OXC table for resource terminal RT2 after successful connection termination. TRUs with STATE = 0 are free.

ing the request and receiving the response from connection control server, state of the request stored in the service control server database is updated (Fig. 12). In implemented system release of optical TRU is emulated in resource terminals as changes in TRU_OXC tables of local database. Contents of TRU_OXC table for resource terminal RT2 after successful test scenario execution is presented in Fig. 13.

6. Conclusions

The aim of the work described in the paper was to implement software and testbed of ASON/GMPLS network, which elements can be mapped into ITU-T NGN functional architecture. Development of initially implemented connection control layer for ASON/GMPLS resulted in creation of the fully operational architecture with basic functionality consisting of service control layer, connection control layer and resource layer – the latter emulates control functions of real devices, optical cross-connects.

In the realization of this goal we used open-source implementations of communication protocols (RSVP and Diameter) and achieved full success. Selecting the most appropriate implementation for both protocols involved comprehensive review and tests of the available code. Finally, we chose KOM RSVP Engine and CDiameter Peer solutions as the most proper and introduced indispensable extensions. In order to use RSVP to control optical transport network consisting of optical cross-connects we had to extend chosen protocol implementation by adding new objects. Moreover, we added new Diameter application as well as messages for the purpose of communication between the particular layers of the architecture.

The implemented functionality of ASON/GMPLS was thoroughly tested in laboratory conditions. The set of tested scenarios included setting-up and releasing connections in various conditions. Performed tests validated correctness of all network elements operation including communication procedures and request processing. Furthermore, for implemented ASON/GMPLS architecture basic performance tests regarding call setup and termination operations durations were conducted. Performance results of request handling in the implemented architecture highly depend on database system used in connection control server. We applied and performed tests with PostgreSQL, Oracle 10g Express Edition databases and our own implementation based on data structures written in C language. Due to space limitation, the results of system performance tests with different types of database system are not included.

Summing up, we managed to implement ASON/GMPLS architecture, which is suitable for transport layer of ITU-T NGN proposition. As developed software is flexible, it could be readily adopted to any network architecture of optical network. Application of our solution in existing and future telecommunication networks requires further work and research concerning performance and reliability issues.

Acknowledgement

References

[1] "Functional Requirements and Architecture for Next Generation Networks". ITU-T Recommendation Y.2012, April 2010.

[2] "Architecture for the Automatically Switched Optical Network (ASON)". ITU-T Recommendation G.8080/Y.1304, June 2006.

[3] E. Mannie, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", IETF RFC 3945, October 2004.

[4] OIF Guideline Document: Signaling Protocol Interworking of ASON/GMPLS Network Domains, June 2008 [Online]. Available: http://www.oiforum.com/public/documents/DIF-G-Sig-IW-01.0

[5] S. Kaczmarek, M. Narloch, M. Młynarczuk, M. Sac, "Next generation services and teleinformation networks – technical, application and market aspects; Network architectures and protocols", PBZ-MNiSW-02-II/2007/GUT/2.6, Gdańsk, December 2010 (in Polish).

[6] R. Braden et al., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification", IETF RFC 2205, September 1997.

[7] A. Fredette et al., "Link Management Protocol (LMP) for Dense Wavelength Division Multiplexing (DWDM) Optical Line Systems", IETF RFC 4209, October 2005.

[8] P. Calhoun et al., "Diameter Base Protocol", 3588 IETF RFC, September 2003.

[9] "Distributed Call and Connection Management: Signalling Mechanism Using GMPLS RSVP-TE". ITU-T Recommendation G.7713.2/ Y.1704.2, March 2003.

[10] M. Karsten, "Design and implementation of RSVP based on object relationships", in Proc. Networkings 2000, vol. 1815, Springer, LNCS, 2000, pp. 325–336.

[11] S. Kaczmarek, M. Młynarczuk, and M. Waldman, "The realization of ASON/GMPLS Control Plane", in Information Systems Architecture and Technology, System Analysis Approach to the Design, Control and Decision Support, J. Świątek et al., Eds. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2010, pp. 313–324.

[12] S. Kaczmarek, M. Młynarczuk, and M. Waldman, "RSVP-TE as a reservation protocol for optical networks", in Proc. XIV Poznań Telecommun. Worksh. PWT 2010, Poznań University of Technology, 2010, pp. 28–31.

[13] D. Awduche et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels", IETF RFC 3209, December 2001.
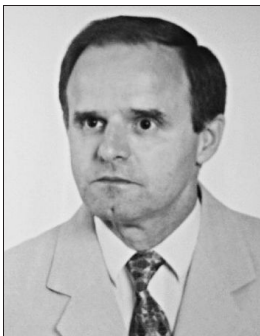
Sylwester Kaczmarek received his M.Sc. in Electronics Engineering, Ph.D. and D.Sc. in Switching and Teletraffic Science from the Gdańsk University of Technology, Poland, in 1972, 1981 and 1994, respectively. His research interests include: IP QoS and GMPLS networks, switching, QoS routing, teletraffic, multimedia services and quality of services. Currently, his research is focused on developing and applicability of VoIP technology. So far he has published more than 200 papers. Now he is Professor and the Head of Teleinformation Networks Department at GUT.
E-mail: kasyl@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Magdalena Młynarczuk was born in Poland in 1980. She received her M.Sc. degree in Telecommunication Systems and Networks from Gdańsk University of Technology in 2004. Since 2008 till March 2011 she has been an assistant at Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics. Now she works as lecturer at GUT. Her research interests include control of optical networks, transmission and switching technology and network design. Her doctoral thesis is entitled: "QoS Routing in multi-domain optical network with hierarchical structure of control plane".
E-mail: magdam@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Marcin Narloch was born in Poland in 1974. He received M.Sc. and Ph.D. in Telecommunications from Gdańsk University of Technology in 1998 and 2006, respectively. Since 2006 he has kept assistant professor position at Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics. His research activities focus on traffic control and service design in IP QoS and GMPLS Next Generation Networks.
E-mail: narloch@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Maciej Sac was born in Poland in 1985. He received his M.Sc. degree in Telecommunications from Gdańsk University of Technology in 2009. Since 2009 he has been a Ph.D. student at Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics. His research interests are focusing on ensuring reliability and Quality of Service in IMS/NGN networks by the means of traffic engineering.

E-mail: Maciej.Sac@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland