On Multi-Domain Connection Admission Control in the EuQoS System

W. Burakowski, M. Diaz, O. Dugeon, A. Pietrabissa, F. Racaru, G. Santoro and H. Tarasiuk

***Abstract*—This paper describes an approach for performing a Connection Admission Control (CAC) function that is investigated in the EuQoS system designed for providing QoS over a heterogeneous, multi-domain network. The CAC plays a key role in QoS provision since it limits the volume of submitted traffic and, in this way, guarantees the assumed QoS level for transferred packet flows. The CAC algorithms are associated with specific end-to-end Classes of Service (CoS). In the EuQoS system we define a set of CoSs that are supported by the system. Each CoS is designed for transferring data corresponding to given type of applications (as voice, video-conference, high data transfer, video on demand etc.) with assuring appropriate QoS level, expressed in the form of packet loss ratio, mean packet transfer delay and packet delay variation. To cope with multi-domain network, the CAC is performed in many places in the EuQoS system. More specifically, the CAC is distributed horizontally, among the different QoS domains, as well as vertically, among Network Technology Independent (NTI) and Network Technology Dependent (NTD) infrastructures developed by EuQoS. NTD infrastructure allows CAC to interact with the underlying networks, such as WiFi, UMTS, LAN/Etherent, xDSL, Satellite, and IP.**

***Index Terms*—Connection Admission Control (CAC), Heterogeneous Networks, Quality of Service (QoS), End-to-end QoS.**

I. Introduction and EuQoS Architecture

THIS paper presents the Connection Admission Control (CAC) function for the EuQoS system designed for providing end-to-end QoS over heterogeneous, multi-domain network. The EuQoS system is developed and implemented by the IST EuQoS project. The key objective of EuQoS ([1], [3]) is to develop and demonstrate end-to-end QoS technologies to support advanced QoS-aware applications (voice, video-conferencing, video streaming, tele-engineering, tele-medicine) over multiple, heterogeneous networks, as WiFi, LAN/Ethernet, UMTS, xDSL, Satellite, and IP core ([2]).

In EuQoS system, each of considered QoS-aware applications is mapped into an appropriate globally well-known end-to-end class of service (CoS), with its predefined QoS parameters as end-to-end packet delay, packet losses, and packet jitter. An application that wants to use the EuQoS system generates, so called, QoS request associated with end-to-end CoS and the role of the CAC is to check whether available resources in the network.

The EuQoS architecture is depicted on Figure 1. It consists of three layers that are: the Application layer, the Network Technology Independent (NTI) layer and the Network Technology Dependent (NTD) layer ([3], [4]).

At the Application layer, EuQoS relies on well-defined user requirements, which are guaranteed by controlling an adequate set of user-defined parameters. The EuQoS signalling sub-system is based on QoS enhancements of the IETF Session Description Protocol (SDP) and the IETF Session Initiation Protocol (SIP). In the Application layer, EuQoS tasks are: (i) to check the coherence between the users and the codec types used by the applications; (ii) to request and check the availability of QoS requirements.

At the NTI level, EuQoS defines the Resource Managers (RM), which are the NTI devices. In current stage of the EuQoS system specification, we assume that each QoS domain is managed by single RM device. The innovative aspect in EuQoS approach is that RMs hold a NTI representation of the network. In this way, the procedures and algorithms developed within the RMs are always the same regardless of the underlying network technologies. The NTI functionalities implemented within the RMs are: provisioning of classes of service, network resource management, admission control algorithms and management of the signed service contracts (i.e. SLSs).

At the NTD layer, EuQoS defines the Resource Allocators (RA), which are the NTD devices. NTD functions have to be implemented to interface the underlying networks with the RM. On the one hand, NTD measures of the underlying network (e.g., topology, link capacity, delays, …) have to be collected to allow the RMs to build a NTI representation of the underlying networks. On the other hand, RM control commands (e.g., decisions on new traffic admission, …) have to be enforced by the RAs to properly set the underlying

This work was supported by project EuQoS (End-to-end Quality of Service support over heterogeneous networks), financed by the European Commission under the IST 6th Framework Programme, IST contract: 004503.

Wojciech Burakowski and Halina Tarasiuk are with Institute of Telecommunications, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland (e-mails: {wojtek, halina}@tele.pw.edu.pl).

Michel Diaz and Florin Racaru are with LAAS-CNRS, 7 avenue du Colonel Roche - 31077 Toulouse, France, Cedex 4 (Tél. : (33) 05 61 33 62 00 - Fax : (33) 05 61 55 35 77, e-mails: diaz@laas.fr, fracaru@laas.fr).

Olivier Dugeon is with France Telecom R&D/CORE/M2I, 2, Avenue Pierre Marzin, F-22307 Lannion, France, (e-mail: Olivier.Dugeon@francetelecom.com).

Antonio Pietrabissa and Gianfranco Santoro are with the Computer and System Science Dept. (DIS), University of Rome “La Sapienza”, via Eudossiana 18 - 00184 Rome, Italy, (e-mails: pietrabissa@dis.uniroma1.it, gsantoro@dis.uniroma1.it).

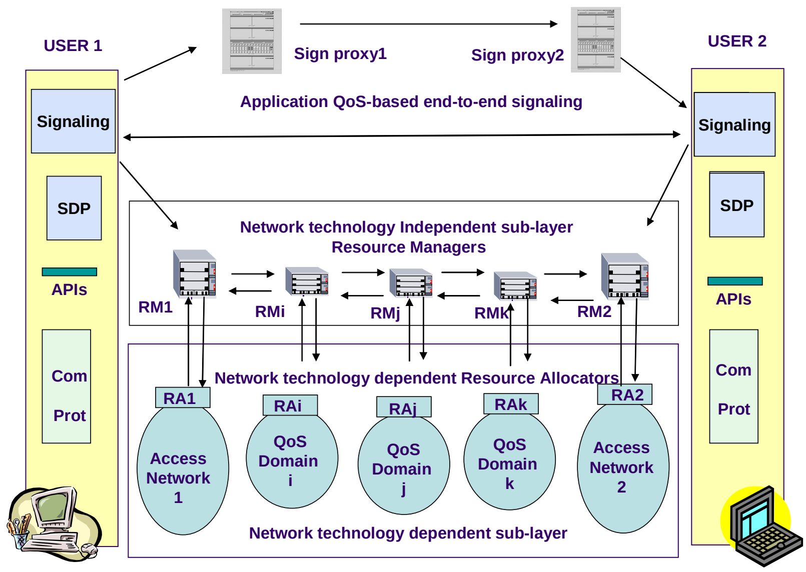

Figure 1 – EuQoS Architecture

network devices (e.g., in provisioning process: DiffServ flow classification, scheduling priorities; in invocation process: packet marking, policing, …).

The EuQoS project could be regarded as a continuation of the AQUILA[1], TEQUILA[2] and CADENUS[3] projects, grouped in IP Cluster inside 5. IST Framework. The main achievements of these projects one can find in [13], [14], [15], respectively.

The paper is organized as follows. Section II describes the overall CAC architecture. Section III presents preliminary implementation results. Finally, Section IV concludes the paper and presents the on-going and future work.

II. CAC Architecture

The CAC function plays a key role in providing end-to-end QoS at the network level in EuQoS system. The CAC limits the traffic submitted to the network and, thanks to that, we can assure the QoS as assumed for the packet level (it means packet delay, packet jitter, packet losses). The CAC function is invoked by each new QoS request and connection release. The CAC decision about admitting or rejecting new connection depends on: the requested resources, the available resources and the resources needed for serving running connections.

The CAC function is strictly related to other functions defined in EuQoS system. In particular, the Traffic Engineering and Resource Optimisation (TERO) function is responsible for the routing process at the NTI layer, where traffic routes to be identified are among network domains. TERO function is also responsible for provisioning network resources, which are allocated to particular class of service in specified network devices. Finally, the Monitoring, Measurement and Fault Management (MMFM) function is aimed at managing network measurements in order to monitor the network resources and to discover the network topology for supporting CAC and TERO functions.

CAC architecture is distributed horizontally, among the different QoS domains, as well as vertically, among the Network Technology Independent and Network Technology Dependent infrastructures developed by EuQoS (see Figure 2). In particular, within a given QoS domain, different CAC modules are present in the NTD and NTI layers:

- The Call Controller is a part of the RM and is NTI. It manages the Service Level Specifications (SLS), representing the QoS requirements of the connections.
- The End-to-End and the Domain CAC are parts of the RM and are NTI. The resources availability checking is performed within the QoS-domain managed by the RM, based on the domain representation contained by the RM data-base (RM-DB).
- The Underlying Network CAC (UN CAC) is a part of the RA and is NTD. Following the QoS request coming from Domain CAC (RM), UN CAC checks whether new connection can be admitted or rejected, and this decision is made by performing appropriate CAC algorithm. The CAC algorithm can be different with respect to both network technology (LAN, xDSL, UMTS, Satellite, WiFi, IP) and type of class of service (e.g. real time, non-real time, etc.) [12].

The SLS are communicated to the RM of a certain domain by using the Signalling and Service Negotiation (SSN) path defined in EuQoS. As shown in Figure 2, the Application layer signalling (ASIG) makes use of Enhanced QoS SIP (EQ-SIP) messages. At the Application layer, the A-SSN

[1] AQUILA - Adaptive Resource Control for QoS using an IP-based Layered Architecture, 5 European Framework, Project No. IST-1999-10077

[2] TEQUILA – Traffic Engineering for Quality of Service in the Internet at Large Scale – 5 European Framework, Project No. IST-1999-11253

[3] CADENUS – Creation And Deployment of End-User Services IN Premium IP Networks – 5 European Framework, Project No. IST-1999-11017

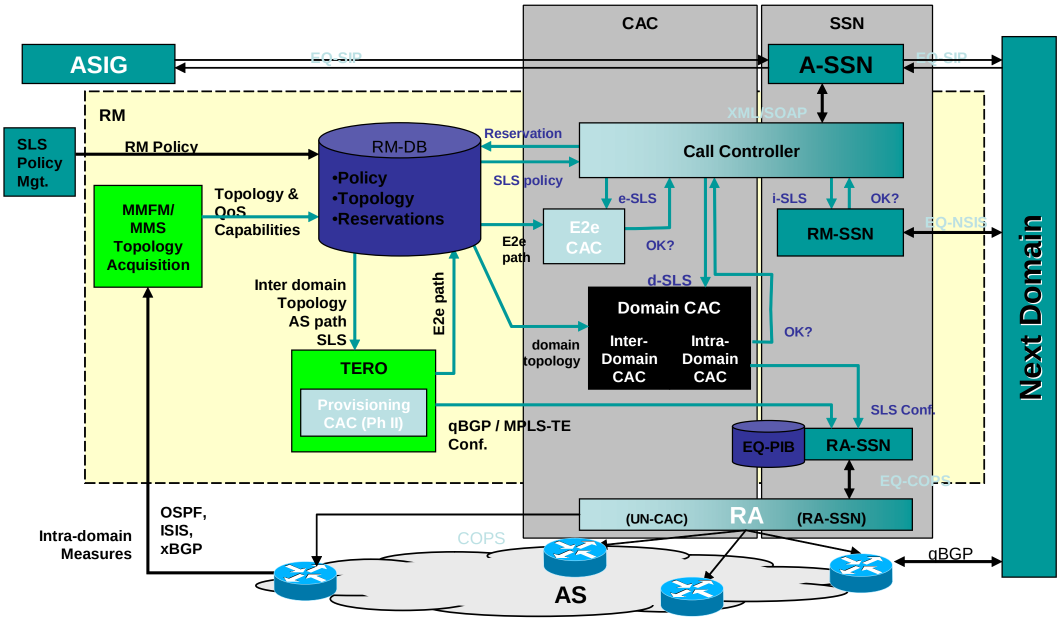

Figure 2 – CAC modules and interaction with other EuQoS functions

(Application SSN) communicates to the Call Controller module the SLS requirements. SLS management within the domain is performed following the policy rules stored in the RM-DB and managed by a SLS management module.

Given that the CAC manipulates QoS parameters at different levels, a certain number of SLS has been defined:

- *e-SLS*: the QoS parameters corresponding to the end-to-end (multi-domain) part;
- *i-SLS*: the QoS parameters of the inter-domain part, defined as the link between two domains;
- *d-SLS*: the QoS parameters belongs to the domain managed by the RM;
- *r-SLS*: the QoS parameters given to the next RM.

Different CAC modules use the above-defined SLSs:

- The Call Controller module, described in Section II.A, receives the SLS from the application level and controls all CAC sub-modules by computing and communicating the *e-*, *i-*, *d-* and *r-SLSs*;
- The End-to-End CAC, described in Section II.B, is in charge of checking whether we have end-to-end path between source and destination domains that satisfies the QoS requirements of new connection.
- The Domain CAC, described in Section II.C, has two sub-modules, devoted to the control of the intra-domain part and the inter-domain part by checking resources availability on the peer link.
- Finally, the Underlying Network CAC (UN-CAC), described in Section II.D, receives the appropriate SLS information via the signalling module devoted to the communication between RM and RA (RA-SSN).

In the next paragraph we describe briefly each CAC sub-module shown in Figure 2.

A. *Call Controller*

The Call Controller, located in the RM, is in charge of the coordination of all CAC sub-modules and interacts with the RM-SSN module, which is responsible for the signalling between peers RMs. Its principal functions are:

- Management of all received requests with the purpose to reserve/release resources that will fulfil application requirements.
- Interaction with other CAC modules to perform global admission control.
- Transmitting signalling to peer Call Controllers if the communication passes through several domains; the communication is performed by means of the EQ-NSIS protocol ([6][5]), a NSIS (Next Steps In Signalling) modified protocol designed to carry EuQoS system signalling messages.
- Communicating with the MMFM module in order to indicate the beginning, the end of connections, or the detection of SLS violation.
- Accessing the RM Database in order to insert, update or retrieve information.

The Call Controller can be triggered to check the resource availability in two ways: (i) by the A-SSN module via a request/response protocol, in the RM of the first domain in the data path; (ii) by a peer Call Controller, via the EQ-NSIS protocol, in all other cases.

A new QoS request received by the Call Controller module contains all needed information to reserve resources for user connections belonging to a certain session. Particularly, the Call Controller requires two types of parameters: (1) QoS parameters specified by an end-to-end SLS (e.g., bandwidth, delay, loss ration, jitter, …); (2) stream parameters that identify each data flow (e.g., IP addresses and ports for sender and receiver, transport protocol to be used, end-to-end CoS, ...).

The first operation performed by the Call Controller is to parse and split those parameters into end-to-end SLS (e-SLS), domain SLS (d-SLS) and inter-domain SLS (i-SLS) – see Figure 2. Afterwards, only in the first domain, it activates the End-to-End CAC module in order to check if a path with the requested Class of Service is available from sender to receiver. If the path is available, End-to-End CAC triggers the Domain CAC module to check if domain resources are available and if yes to make the related reservation (we deal with a per flow reservation at this level). If the response is positive, the Call Controller activates the RM-SSN module (unless its RM is the last on the path) to signal next peer Call Controller. Once having a positive answer, it returns the acknowledgement to the entity that sent the request (A-SSN or peer RM) and stores the reservation information in the database. The Call Controller communicates also with the MMFM module in order to inform it about the start/termination of a connection.

In the case of a release request, the mechanism is identical, with the difference that the End-to-End CAC module is not activated. Presently, only requests for reserve and release are supported; in the final implementation, update requests for active sessions will be considered.

B. End-to-end CAC

End-to-end CAC is defined by checking if the network has capabilities for admitting new connection, as demanded by QoS request. More precisely, we involve this module for checking if from source to destination AS (Autonomous System) exists the EQ-path (Enhanced QoS path), which guarantees the requested QoS at the packet level. This checking is performed only in source domain and only positive decision about EQ-path availability allows continuing invocation process. Otherwise, the process returns an answer about rejection of new connection. Let us pay your attention that according to the assumed framework for QoS provision in EuQoS [4], a EQ-path supports a given end-to-end class of service. Information about available EQ-paths with respect to particular end-to-end class of service come from TERO function and, for this purpose, TERO engages EQ-BGP (Enhanced QoS Border Gateway Protocol) protocol [7]. At this part of the invocation process there is no information about resources allocated on the EQ-path. Therefore, only if the decision about EQ-path availability is positive, the next steps of invocation process can be proceeding.

C. Domain CAC

As it was above mentioned, the Domain CAC consists of two sub-modules, which are Inter-domain CAC and Intra-domain CAC.

Inter-domain CAC. Even if the EQ-path exists, it may happen that there are no available resources on this path because of running connections. As a consequence, in the general case, when QoS paths exist, their capabilities along the paths can be quite different: the existence of the requested resources along the path, in the consecutive inter-domain links has to be dynamically checked for the requests. The inter-domain CAC triggers RA.

Intra-domain CAC could be different for each domain since different underlying networks are considered. Intra-domain CAC can apply to QoS domains or AS constituted either by only an access network or by both an access network and a related core network. It is quite difficult to define a generic efficient solution for all considered by EuQoS technologies. The choice of the solution and of its implementation has to be left as a technology dependent matter. The decision has to be taken by the different technology providers, i.e. by the designers in charge of providing the QoS inside a given domain. Thus, the Intra-domain CAC has to trigger the RA, performs the network independent CAC, and integrates the results.

D. UN CAC

During the session set-up phase, the Domain CAC (Intra-domain and Inter-domain CAC) invokes the RA (see Figure 3) for performing the required actions at the technology dependent layer that are necessary for admitting the new connection in the considered QoS domain. Since the RA has no information about the domain topology (intra and inter-domain), the Domain CAC module provides to the RA the list of devices that are considered as bottlenecks. Those devices for which the CAC is performed are the bottlenecks in the network, where we need to make the resource reservation for assuring the QoS for given connection. When the new request arrives, the UN CAC process is the following:

- RA Controller asks the CAC algorithm associated with a given class of service. For each underlying network technology (e.g. WiFi, LAN/Ethernet, inter-domain link, etc.), a specific set of classes of service should be defined, which are mapped into end-to-end classes of service.
- CAC algorithm checks whether the amount of available resources (in RA-DB) is sufficient for serving new connection.

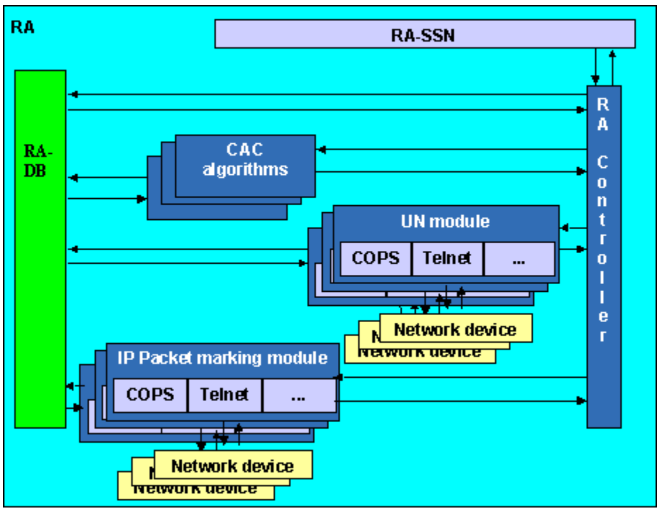

Figure 2 – RA architecture

- If the decision is positive then RA Controller asks the appropriate UN module to configure the involved network devices for handling the new connection. Obviously, different mechanisms must be configured depending on the type of network device (e.g. IP router, LAN/Ethernet switch, WiFi Access Point, etc…). Note that this step is optional. For example, per flow QoS mechanisms in inter-domain border routers will be not configured.
- In the case of negative decision, RA Controller terminates the invocation process and sends this information to the Domain CAC.
- RA Controller asks the appropriate IP Packet Marking module to configure packet marking in the involved network devices.
- After receiving a positive answer from the UN module, RA Controller inserts information about the new active session in the RA-DB and sends confirmation to the Domain CAC.

UN CAC is also responsible of releasing the reserved resources, when the session is terminated.

A proposal of CAC algorithms and method for calculating the admissible traffic load of real time class of service for inter-domain peers is presented in [11].

III. Preliminary implementation results

This paragraph presents several aspects of CAC modules implementation [8]. A prototype of EuQoS architecture was demonstrated at CER 2005 in Brussels [9]. The Resource Manager was implemented in Java [10], and the whole system was tested using three kinds of applications: Visio Conference, Video-on-Demand and a Medical Application.

The demonstration scenario encompasses two EuQoS domains (WiFi and LAN), supporting three Classes of Service (real time, non real time and standard) with guaranteed QoS parameters.

The main functionalities of the CAC are already implemented in the RM and RA prototypes. The Call Controller is provided with full interface with A-SSN, end-to-end CAC, domain CAC and RM-SSN. Moreover, it loads in the database information related to active/terminated sessions for administration, log or billing purposes. The Domain CAC was capable of triggering the proper RA CAC, of checking the Inter-domain link based on the data stored in the RM DB (only bandwidth availability was considered in the prototype), and of returning the domain CAC decision to the Call Controller.

The demonstration succeeded in providing session establishment and termination. In particular, the CAC function was demonstrated by testing Video-on-Demand application. The video degradation due to background traffic was effectively overcome by upgrading the CoS of the application from standard to real-time, thanks to resource reservation provided by the coordinated actions of CAC modules in the RMs and in the RAs.

IV. Conclusions and Future Work

This paper presents the overall CAC architecture implemented in EuQoS system that is designed for providing QoS in a multi-domain heterogeneous network environment. The CAC approach follows the EuQoS architecture that assumes three–layer structure: Application, Network Technology Independent and Network Technology Dependent layers. It appears that to perform overall CAC we need to implement CAC modules at each of these layers, and, in addition, over each domain where the network layer is considered. As a consequence, a variety of CAC modules (and algorithms) are engaged. This leads to a complex process required for establishing new connection. Despite of this, a working prototype of EuQoS system has been implemented and partially tested.

Next step is to simplify the CAC procedure keeping in mind the scalability of the solution. This can be achieved by forcing e.g. the MPLS tunnels in the network.

Acknowledgment

The authors wish to thank all the EuQoS partners for their help in the development of the architecture.

References

[1] FP6-IST/EuQoS project, (September 1, 2004 – August 31, 2007); http://www.euqos.org
[2] EuQoS – Annex I "*Description of Work*", vers 5.0.1, May 2004.
[3] O. Dugeon, D. Morris, E. Monteiro, W. Burakowski, M. Diaz, "*End to End Quality of Service over Heterogeneous Networks*", Network Control and Engineering for QoS, Security and Mobility (NetCon'05), Lannion (France), 14-18 November 2005.
[4] EuQoS Consortium, Deliverable D1.1.3, *"Business models and system design specification"*, (not yet published), August 2005.
[5] S. Salsano, *"SIP Extensions for QoS support"*, draft-veltri-sip-qsip-01.txt.
[6] R. Hancock et al., "*Next Steps in Signalling (NSIS): Framework*", RFC4080, IETF, June 2005.
[7] A.Beben, "*EQ-BGP: an efficient inter-domain QoS routing protocol*", In proc. of the SOCNE'06 Workshop, Vienna, Austria, April 18-20, 2006 (accepted for publication).
[8] EuQoS Consortium, Deliverable D3.1.1, *"Extended QoS API and Middleware layer for phase 1 application use-cases"*, (not yet published), August 2005.
[9] Communicating European Research (ERA) 2005 International Conference, Brussels, 14-15 November 2005, http://europa.eu.int/comm/research/conferences/2005/cer2005/index_en.html
[10] EuQoS Consortium, Deliverable D3.1.3 - Annex A, *"EuQoS Software Installation Manual"*, (not yet published), February 2006.
[11] H. Tarasiuk, R. Janowski, W. Burakowski, "*Admissible traffic load of real time class of service for inter-domain peers*", In Proc. of ICAS/ICNS 2005, published by IEEE Computer Society, October 2005, Papeete, Tahiti, French Polynesia.
[12] EuQoS Consortium Deliverable D1.1.3, "*Business Models and System Design Specification*", Chapter 2: Framework for QoS provision, September 2005.
[13] A. Bąk, W. Burakowski, F. Ricciato, S. Salsano, H. Tarasiuk, „*A framework for providing differentiated QoS guarantees in IP-based network*", Computer Communications 26 (2003), Elsevier, pp. 327-337.
[14] E. Mykoniati, et al., „*Admission Control for Providing QoS in DiffServ IP Networks: The TEQUILA Approach*", IEEE Communications Magazine, Vol. 41 Issue 1, January 2003, pp. 38-44.
[15] D. Goderis et al, "*A Management and Control Architecture for Providing IP Differentiated Services in MPLS-Based Networks*", IEEE Communication Magazine, May 2001.