A Generic Service Model for QoS Management

Service Management in Next Generation Network

Tatiana Aubonnet
Département Informatique
Conservatoire National des Arts et Métiers, CEDRIC
Paris, France
tatiana.aubonnet@cnam.fr

Noemie Simoni
Department INFRES
TELECOM ParisTech, LTCI, UMR 5141 CNRS
Paris, France
simoni@telecom-paristech.fr

***Abstract*—We introduce a generic service model aiming at ensuring Quality of Service management. This modelling approach has been proposed by the Next Generation Network and Service Management project. A fairly high integration level of the tools has been reached using object-oriented paradigm and Model Driven Interoperability approach. We provide an example of Quality of Service management through a Service Level Specification of the Virtual Private Network service. We consider important points to reflect the complexity introduced by the Service Level Agreement and Quality of Service management: reaction model, cooperation model and co-ordination model.**

Keywords-QoS management; SLS; proactive SLA; VPN

I. INTRODUCTION

Today, new services have to be rapidly deployed upon various types of networks. Moreover, the provisioning and the assurance of a wide range of services depends on the orchestration of heterogeneous, widely distributed software components, which can be owned by different service providers and operate over diverse networks. In such a scenario, designing and providing complex, value-added services, ensuring their nominal quality levels with traditional, service deployment, provisioning, monitoring and management becomes increasingly difficult and costly.

To answer this problem, one possible key of the service management is to build new services based on SLS (*Service Level Specification*) templates. According to the object-oriented concept (OMG standards) and MDI (*Model Driven Interoperability*) approach [2], we propose the object model description of the SLS template for QoS (*Quality of Service)* management. The main advantages of the object-oriented approach are the modelling, the overall behaviour of the system and the flexibility which permit modularity, portability, re-usability and easily extensible object classes.

Our contribution to this problem is to take into account three management responsibilities: user, application and network. Additionally, we use the same modelling to which we add the following models: the *co-ordination model* which addresses the dynamic management process by identifying the different steps which should be taken in a running (changing) context; the *interaction behaviour (reaction) model* further which specifies the autonomous degree of the distributed components (delegated agents: passive, active, interactive, proactive). By assigning the proactive behaviour type to the delegated agent according to different cases, more proactive SLA (*Service Level Agreement*) can be obtained for the QoS management.

We present in this paper the feedback of our experience on the QoS dynamic management. This paper is organized as follows. The SLS context for QoS management is described in Section 2. Section 3 presents characteristics of existing SLS templates. Section 4 is devoted to our propositions for generic service model, the specification of a VPN service is developed as an example. We give an example of VPN architectural model in Section 5. Our propositions for a proactive SLA are presented in Section 6. Finally, in Section 7, we exhibit the advantages of our approach in Next Generation Network.

II. SLS TEMPLATES USAGE

A Service Level Agreement (SLA) is a formal negotiated agreement between two parties. It is designed to create a common understanding about services, priorities, responsibilities, etc. [25]. A Service Offer or a Commercial Offer may by a set of elementary services.

A Service Level Specification (SLS) is the technical part of a SLA. More formally it has been defined in [3] as a protocol independent representation of a set of technical parameters and their associated semantics that describe the transport service that a (packet) flow is to receive over the transport domain, between ingress and egress interfaces.

The TMF (*TeleManagement Forum*) SLA Description corresponds to the SLS [14], the SLA template corresponds to the SLS template (Figure 1). A SLS template is associated to an elementary service. The SLS is a totally instantiated SLS template that can be used to provision, activate and monitor the corresponding elementary service. The question will remain in the latter case on how to manage the services consistency at the provisioning stage (synchronisation, rollback in case one of the services is unavailable, subscription sequence, etc.) and for assurance (correlation of elementary services alarms on the service offer, execution of proactive and reactive maintenance activities).

 ISBN: 978-1-61208-183-0

A structure of information blocks that can be seen as concrete classes in the oriented-object paradigm.

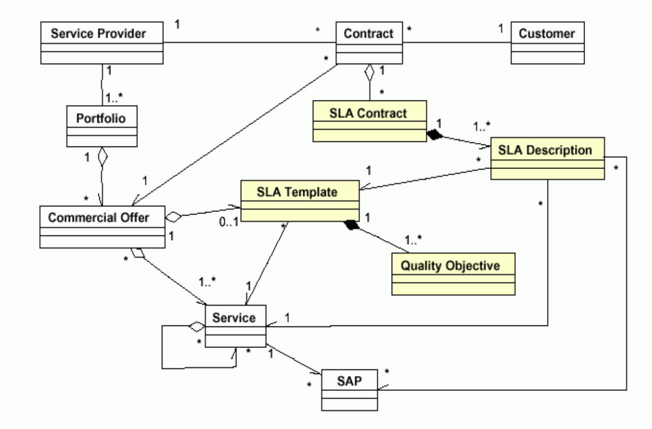

Figure 1. TMF SLA Model.

Moreover, the set of QoS parameters that the service should follow and that should be monitored is instantiated.

III. CHARACTERISTICS OF SLS TEMPLATES

In this section, we present a panorama concerning the SLS templates in different projects: TEQUILA consortium (Traffic Engineering for Quality of Service in the Internet, at Large Scale), Eurescom P1008 project "Inter-operator interface for ensuring end-to-end IP QoS" and Eurescom P1103 project "Inter-Operator IP QoS Framework - ToIP and UMTS Case Studies" [8, 9, 10, 11]. In these projects the template is given as an example and SLS negotiation is described. The basic information to be included in SLSs, and lists a set of basic parameters, which will actually compose the elementary contents of an SLS. The common characteristics of existing SLS templates are:

- Scope – topological region (ingress, egress) interfaces.
- Flow description– SLS is per flow (diffserv info, source info, destination info, app info).
- Traffic description– test if in- or out-of-profile [peak rate, MTU, bkt depth].
- Excess treatment – how to handle out-of-profile traffic [dropped, shaped, remarked].
- Performance Parameters – service guarantees the network offers customer [delay, jitter, pkt loss and throughput].
- Service Schedule – start time and end time – i.e., when the service is available.
- Reliability – downtime and time to repair.
- Others Parameters – route, reporting guarantees, security etc.

We analyzed these characteristics and the documents of these projects and in the following section we propose a *generic service model*, i.e., a template generic compatible with the different next generation services.

IV. A GENERIC SERVICE MODEL

In this section, we consign our propositions for a generic service model. We will present our propositions for:

- Information model (Section A).
- The different levels of visibility for the SLS description (Section B).
- QoS model (Section C, D).

The information model proposed in Section 4.1 is used to analyze the NGNSM SLS template, in order to obtain a generic NGNSM SLS.

A. Information Model

The Information Model (IM) is an approach to the management of systems and networks that applies the basic structuring and conceptualisation techniques of the object-oriented paradigm. The approach uses a uniform modelling formalism that-together with the basic repertoire of object-oriented constructs, supports the cooperative development of an object-oriented schema across multiple organizations. Ideally, information used to perform tasks is organized or structured to allow disparate groups of people to use it. This can be accomplished by developing a model or representation of the details required by people working within a particular domain. Such an approach can be referred to as an information model. An information model requires a set of legal statement types or syntax to capture the representation, and a collection of actual expressions necessary to manage common aspects of the domain of QoS management.

This section describes a generic QoS information model object-oriented. This model includes expressions for common elements that must be clearly presented to management applications. The purpose of the Information Model is to give the structure to the management information and to model management aspects of the related resources [16]. The information model deals with managed objects which provide abstract views of the physical and logical resources for the purposes of management. It provides guidelines for describing the logical structure of the managed objects and other pertinent management information about such objects.

A generic information model is essential to the generation of uniform fault, configuration, performance, security, and accounting management which can be applied to the heterogeneous and distributed environment. On the basis of analysing and comparing with the existing work which has been done in [15, 16] the ENST has proposed an information model.

The information model is presented as a set of structured classes of objects in *different levels of visibility*. The class Network Element (NE), which represents each network objects and is the root of this logical structure, is described as an element consisting of:

- *Network Elements* (v) which are in the same level (v) with the considered NE.
- *Network Elements (v-1),* which are in the lower level(v-1) and provide *a service to the level* (v).

- *Architecture Element*, whose behaviour is expressed with the help of static and dynamic properties.
- A *service class*, which is used to express the service it offers (role *server*) and the operations called by the element (role *client*).

Precisely, with the managed *object model* [23] it is possible to set a managed object as a delegated autonomous agent, as each managed object is provided, in addition to its basic *Operational Service Interface*, with a *Management Interface*. This interface can be very well used for today's management where simply there are notification transits and sometimes tunings of QoS parameters. But it can also, and especially, be used for our distributed perception of QoS management. It makes it possible, indeed, to set management rules and to endow managed objects with some delegated intelligence in order to have some distributed LDPs (*Local Management Decision Points*), thus avoiding depending solely on a unique MDP (*Management Decision Point*). These different behaviours correspond to different object status that appears in the management class description in the information model. QoS information model is dependent on QoS instrumentation, which could by done during the design phase.

B. Different Levels of the SLS Description

The information model described in the preceding section is used to analyze the NGNSM SLS template, in order to obtain a generic NGNSM SLS. Initially we analyzed the report of Alcatel “SLS template and principles” and in the present section, we will present our propositions for modelling. Our first proposition contains one of our important rules: the different levels of visibility. Considering all this we would recommend, it would be necessary to have the QoS constraints for each level and the QoS metrology associated to handle the QoS contracted.

We proposed the following levels for the description of information:

- Generic Service Level (for the SLA).
- Traffic flow level.
- Network connectivity level.

The QoS per flow is proposed for the traffic flow level (Figure 2).

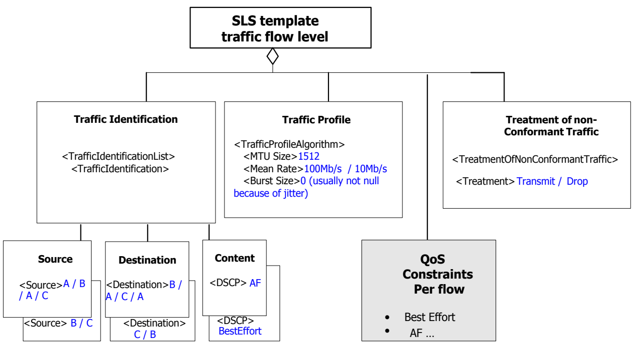

Figure 2. IP/VPN information model : traffic flow level

The information model in these different levels, which is object-oriented, provides an abstraction of the resources and flexibility of the development. The link between the levels of visibility is made thanks to the architectural model (see Section 5) and to the aggregation of levels V-i.

C. QoS Model : Principles

Always on the basis of the document [20] we analyze the SLS template for Generic Service Level. Our “grid of reading” is always done through our conceptual tools [1, 12, 13, 16, 17, 21].

QoS model provides the basic support for organizing management activities. In the Figure 3 we introduce some of the propositions for QoS model:

- As much as it is necessary to validate its generic information model by checking that all the applications will find information which they need, as much as it is necessary that the information model is independent of the applications. This is why, it is necessary to have a generic terminology and to choice a good vocabulary (e.g., Provisioning, Monitoring, etc.). Nevertheless, if we want to mention some application, we propose the “application QoS scope” (Figure 3).
- We think that the “Provisioning QoS” which contains information describing the QoS of service used for the provisioning process, represents *contract QoS information* it must be monitored. So “provisioning QoS” and “Monitoring QoS” contain the parameters which we indicate by QoS “Design_value”.
- Commitment: Parameter that should be monitored with all the needed configuration in term of assurance is the parameter which we indicate by QoS “thresholds_value” [19].

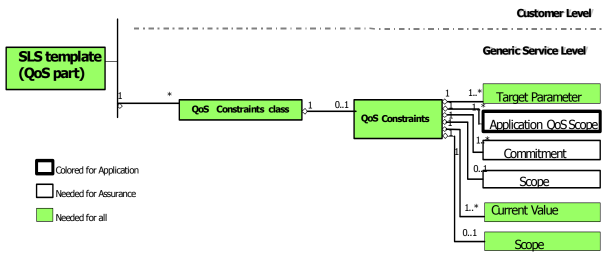

Figure 3. Propositions for SLS Template (QoS part) in Generic Service Level

D. QoS Model : Parameters

According to the description given in [5], the following set of eight QoS criteria are needed for a comprehensive QoS appraisal: Availability, Fidelity, Capability, Speed, Reliability, Flexibility, Usability and Security. Among the eight listed above, only four are essential to describe the behaviour of the service: speed, fidelity/accuracy, capacity and availability [7]; all of which will be taken into consideration.

 ISBN: 978-1-61208-183-0

Service Components / Medium	QoS criteria and parameters					QoS class Y.1541	CoS UMTS	QoS class G1010	PHB	QoS criteria depending classes INTRADIFF
	Delay	*Delay variation*	*Fidelity (Information loss)*	*Capacity*	*Availability*					
Interactive games /data	< 200 ms	U	Zero	DBW	UAT	Class 2	Interactive	EI Interactive	AF1.1	CoS 6.4
Telecontrol / Data	< 250 ms	U	Zero	DBW	UAI	Class 2	Interactive	EI Interactive	AF1.1	CoS 6.4
Telnet / Data	< 200 ms	U	Zero	DBW	UAI	Class 2	Interactive	EI Interactive	AF2	CoS 6.4
Video TeleConfer. service (VTC) / Video	<150 ms	U	Error tolerant PLR < 1%	DBW	UAT	Class 0	Interactive	ET Interactive	AF1.2	CoS 6.2
	400 ms with echo control	U	Error tolerant PLR < 1%	DBW 16-384 kbit/s	UAT	Class 1	Interactive	ET Interactive	AF2	CoS 6.3
Audio-conference /Audio	<150 ms	< 1 ms	Error tolerant PLR < 3%	DBW 4-64kbit/s	UAT	Class 0	Convers.	ET Interactive	EF	CoS 6.0
	400 ms with echo control	< 1 ms	Error tolerant PLR < 3%	DBW 4-64kbit/s	UAT	Class 1	Convers.	ET Interactive	EF	CoS 6.1
Telephone service / Audio	<150 ms	< 1 ms	Error tolerant PLR < 3%	DBW	UAT	Class 0	Convers.	ET Interactive	EF	CoS 6.0
	400 ms with echo control	< 1 ms	Error tolerant PLR < 3%	DBW	UAT	Class 1	Convers.	ET Interactive	EF	CoS 6.1
Voice messaging record / Audio and playback / Audio	< 2 s for record	< 1 ms	Error tolerant PLR < 3%	DBW 4-32kbit/s	UAT	Class 1	Interactive	ET Responsive	AF3.2	CoS 5.2
	< 1 s for playback								AF3.2	CoS 5.0
Electronic mail SMTP/POP server access / Data	< 2 s (< 4 s/page acceptable)	U	Zero	VBW	UAT	Class 4		EI Responsive	AF3.1	CoS 5.2
Web Browsing / Data	<2 s/page (< 4 s/page acceptable)	U	Zero	VBW	UAT	Class 3		EI Responsive	AF3.1	CoS 5.2

TABLE I. QOS PARAMETERS FOR EACH SERVICE COMPONENT

Table 1 gives an example of QoS requirements for the above services and a quantitative comparison between the proposed model and other SLS templates models. The difference consists in using only one model. The QoS agent is included in each component and it manages QoS according to the four criteria defined for the current value: Delay / Delay variation, Fidelity (Information loss) Capacity, Availability. Each one these of these criteria should be expressed in quantifiable and measurable parameters (see five columns of the QoS criteria and parameters).

A state-of-the-art effort has been performed in order to situate this model with respect to other generic models of the international community (ITU-T M3100 [15], ETSI GOM [6], TINA-C NRIM [24]) and to propose our SLS template model which is in this context instantiated to the VPN service.

V. ARCHITECTURAL MODEL: VPN APPLICATION

The following section is the connection between the levels of visibilities. Indeed, it is necessary to be able to have the traceability between the levels. Our answer is given by the architectural model which translates the aggregation and the co-operation of the whole of the network components.

In this section, we well examine our proposition of architectural model for end to end QoS, by using DiffServ/IP/VPN case study [4, 22]. After having introduced our QoS model in the previous section, it would be interesting to explore its capabilities and contributions through the case study on DiffServ/ IP / VPN. We consider for this purpose a distributed system consisting of a carrier's network built from multiple DiffServ [12] domains. The network is intended to provide customers with differentiated services.

In IP, VPN service relies on the VR (Virtual Routing) functionality that may include tunneling (encapsulation) and securing (IPSec). In addition to this functionality, the DiffServ VPN region handles, through the IDC (Inter Domain Connection) function, differentiation interoperation among the DiffServ domains regarding traffic aggregation (TA), traffic conditioning (TC) and aggregate forwarding (PHB). A domain PHB shares to the bearer IP network elements IPF (IP Forwarding) and IPR (IP Routing). Thus, the manageable distributed components of the system are VRs, IDCs, TAs, TCs, PHBs, IPFs and IPRs. This leads to the network abstract model depicted in Figure 4.

In accordance with the <Node, Link, Network> abstract model [23], we see that the DiffServ VPN is a network resource composed of nodes (VR, IDC) and links (VPN link) of the same visibility level. It relies on DiffServ domains, which are networks of lower visibility level. Each DiffServ domain is, at its turn, composed of nodes and links of the same visibility level and relies on a network of lower visibility level. We have assigned DiffServ domain components TA, TC and PHB to different visibility levels to be able to make accurate decisions according the services performed by each one of them.

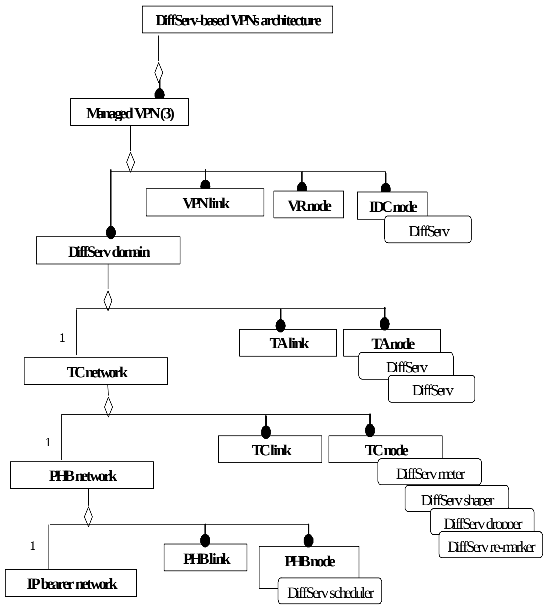

Figure 4. VPN architectural model

Each DiffServ domain (and its sub-networks) relies on the IP bearer network. Visibility level IP and visibility levels below than IP are not developed here. Application of the managed object model leads to the following managed objects: VR nodes, IDC nodes, TA nodes, TC nodes, PHB nodes, IPF nodes and IPR nodes. Note that since manageable objects belong to different visibility levels, some of the managed nodes may share the same physical equipment.

VI. A PROACTIVE SLA

In this section, the request is: how to define a proactive SLA between a service provider and a user. To answer this request, we present:

- Modelling supports with our view point for SLA/QoS modelling (Section A).
- Different models to confer the proactively capacity (Section B).

The first answer is the reaction capacity during the operating phase, i.e., to meet the dynamics requirements. Then the possibility *to anticipate* and to decide in an autonomous way, the nodes will confer to them the proactively capacity.

A. A View Point for SLA/QoS Modelling

Until now, we considered the SLA as the user QoS request with respect to his applications. However the nature of these applications induces their own constraints. This is why, we think that to take into account the user "desired", it is necessary to pass at the higher visibility level, above service visibility level.

We can say that the service visibility level is constrained by application dependent or ISP dependent where as the user visibility level (SLA) could be:

- Agenda dependent.
- Localization dependent,
- Terminal dependent, etc.

In other words, on the service level we have the decisions which rise of the responsibility for the applications. On the network level we have the tactical decisions which concern the responsibility for the network and thus for the operator. Whereas, for the user level we have the strategic and organizational type decisions, concerned with the enterprise and its users responsibilities.

Therefore, we propose to consider the SLA/user level as a "service" and to keep the same model and the same modelling support for other levels services.

B. Our propositions for proactive SLA

To fulfil the requirements indicated in the precedent sections, a powerful and flexible approach should be adopted. In accordance with these objectives, in the present section, we propose the different sub-models for dynamic SLA/QoS :

- *Reaction model* which applies in every node and which classifies four types of object behaviours reflecting different autonomous levels.
- *Co-operation model* which identifies the roles engaging in the distributed management activities, as well as the relations among them.
- *Co-ordination model*, which is proposed to provide a means to support dynamic management to guarantee an end-to-end QoS.

Reaction model (behaviour model)

Management tasks are performed via the interactions among the objects. The interactions between the objects are performed by sending messages from one to another. The different behaviours exhibited by the objects during their interactions. Specifically, four types of interaction behaviours are identified: passive, active, interactive, pro-active.

A passive object encapsulates some resource and a set of routines and operations that can be performed on the resource. It provides services which are used by one or more active objects. A passive object can only be involved in the manager-agent relation, and plays the role of agent. All the manageable objects should be at least passive.

An active object performs some function and may also encapsulate some resource and the operations for accessing it, but it may invoke operations on other objects. It can be

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-183-0

used to describe the object behaviour exhibited in the manager-agent relation. An example of this type of behaviour occurs when an agent is requested to perform an m-action by a manager using the protocol CMIP (Common Management Information Protocol) defined in [18]. An interactive object describes the interaction behaviour of an active object in which the object has needed to obtain interactively complementary information in order to continue the on-going process. It can be used in the negotiation between the manager of managers and a manager, or in the peer-to-peer relation such as the relation between two managers who are not in the same domain. It also can be applied in the manager-agent relation.

A proactive object describes the interaction behaviour of an active object in which the object, who is highly-autonomous, does not simply act in response to their environment stimulus (changes), they are able to exhibit goal-directed behaviour by taking the initiative thereby reacting to indicators rather than reacting to severe problems as perceived by the user. This is the case where a managed object can automatically detect problems and find the pre-determined solutions when an event occurs without the manager's intervention, allowing for network self-healing. The proactive agent can be used to maintain the QoS dynamically.

A fundamental approach to achieve the proactive management is to characterize carefully different problem conditions in the network and to address appropriately their resolution for recovering from complicated situations or situations that require higher levels of reasoning and the correlation of multiple, seemingly, disparate problem conditions.

Under different conditions (e.g., in different contexts or when receiving different stimuli), the object behaviour can change among status of passive, active, interactive, and pro-active. These four types of interaction behaviours outline how the network components support the management policies in order to maintain the contracted QoS, especially to contribute to the dynamic QoS management.

Cooperation model (relation model)

Managing distributed systems introduce more complexity. Management responsibilities are structured and partitioned to the sub-systems. Each sub-system is responsible for only a local portion of the overall area. In order to reflect these above characteristics, the roles and their relation model are needed to identify the roles of an entity involved in the management activities and the relations between these entities, which is shown in Figure 5.

According to their different responsibilities taken in the management, four types of roles. These objects can be:

- Manager is used to refer to any entity, human or automated, that can perform management activities such as control, co-ordination and monitoring.
- Manager of Managers (MoM), similar to manager, but in a higher-level in comparison with other managers;
- Agent refers to any entity that provides the access and performs the operations requested by the manager on the managed objects. It reflects to the manager a view of these objects and sends notifications reflecting the behaviour of these objects
- Managed objects (Mos) provide abstract representations of the managed resources. Managed objects may be organized into sets called management domains as a result of organizational requirements. These domains achieve a partitioning of the management environment based on functional areas or according to geographical, technical or organizational criteria.

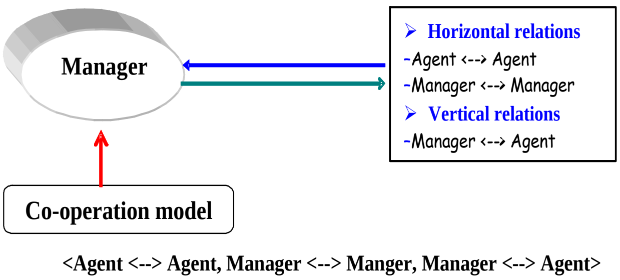

Figure 5. A cooperation model

This supports the distribution and delegation of management functionality and also supports co-operation between different components of the management infrastructure. It is, therefore, not only possible to delegate functionality from managing systems (managers) to managed systems (agents) but also between managed systems. Another point is that the roles participating in management activities are subject to dynamic changes: roles like manager and agent are temporary and bound to the tasks to be done. The change between these two roles results from the relations with other entities during the management activities.

Co-ordination model (organization model)

The co-operative management process can be represented by the co-ordination model which contains seven phases:

- Request
- Translate
- Hop-by-hop negotiate
- Accord
- Reject
- Supervise
- Re-negotiate

The objective of the management is to guarantee the end-to-end QoS required by the user. This is achieved by managing the co-operative management process among the individual object QoS. The QoS is requested and then translated into comprehensible parameters (QoS parameters). All the negotiation results should be reported to the corresponding responsibility level according to management policies.

 ISBN: 978-1-61208-183-0

VII. Conclusion

In today's deregulated and competitive market, telecom operators and ISPs need to be able to define their individual management goals and to adjust their individual decisions to meet their specific needs while respecting the general management agreement that governs their global cooperation.

Some ideas from them, especially the idea of the integration of the information model in different levels of abstraction to re-use the information and to provide a template generic compatible with the different NGN services.

To enable implementation of consistent end-to-end QoS for such environments, the QoS model applied in this article provides capabilities to structure and to partition management in a large distributed system as well as to adapt flexibly to changing requirements:

- Through the managed object model and the network abstract model, it makes it possible to organize the management system into distributed domains and to support dynamic QoS management by distributing management tasks and decisions among the system domains.
- Through the architectural model it provides the traceability.

In this article, we considered two important points to reflect the complexity introduced by the SLA/QoS management:

- We proposed to take into account the SLS template based object-oriented paradigm.
- We used the same modelling to which we added the following models: *interaction behaviour (reaction) model, cooperation model and co-ordination model.*

By assigning the proactive behaviour type to the delegated agent according to different cases, the proactive SLA can be obtained for the QoS dynamic management.

Acknowledgment

We would like to thank partners Alcatel: Gilles Désoblin, Olivier Martinot and Véronique Daurensan for their fruitful participation in NGNSM (*Next Generation Network and Service Management*) project, providing a SLS reference model and support of the tool suite.

We would also like to thank Michel Cotten and Kirill Polishchouk for their help and advice in finalizing this article.

References

[1] T. Aubonnet and N. Simoni, “PILOTE: A Service Creation Environment in Next Generation Network,” Proceedings IEEE Intelligent Network Workshop, IN’2001, pp. 36-40, Mai 6-9, 2001, Boston, USA.

[2] J. Bézivin, R.M. Soley, and A. Vallecillo, “Model-Driven Interoperability: MDI 2010,” Proceedings of The First International Workshop on Model-Driven Interoperability, MDI 2010, pp. 147-149, October 5, 2010, Oslo.

[3] E. Bouillet, D. Mitra, and K. Ramakrishnan, “The Structure and Management of Service Level Agreement in Networks,” IEEE Communications Magazine 41(7), pp. 102–109, 2003.

[4] M. Conti, A. Hasani, and B. Crispo “Virtual private social networks,” Proceedings of The First ACM conference on Data and application security and privacy, CODASPY 2011, February 21-23, 2011, San Antonio, USA.

[5] ETSI EG 202 009-1, “User Group; Quality of Telecom Services; Part 1 : Methodology for identification of parameters relevant to the Users,” 2009.

[6] ETSI NA 43316, “Telecommunications Management Network (TMN) Generic Managed Object Class Library for the Network Level View,” Mars 1995.

[7] ETSI TR 000 029-1 V0.1.8, “User Group; End-to-end QoS management at the Network Interfaces; Part 1: User’s E2E QoS – Analysis of the NGN,” 2009.

[8] EURESCOM IST project TEQUILA “Traffic Engineering for Quality of Service in the Internet, at Large Scale,” April 2000.

[9] EURESCOM, Project P1008 “Specification of Inter-domain Quality of Service Management Interfaces,” May 2001.

[10] EURESCOM, Project P1103 “Inter-Operator IP QoS Framework - ToIP and UMTS Case Studies,” Jan. 2002.

[11] EURESCOM, Project P806-d1 “A Common Framework for QoS/Network Performance in a multi-Provider Environment, Deliverable 1: The EQoS Framework,” Janvary 2002.

[12] IETF, Network Working Group, “Quality of Service (QoS) Mechanism Selection in the Session Description Protocol,” RFC5432, March 2009.

[13] ITU-T Recommendation E.800, “Terms and definitions related to quality of service and network performance including dependability,” August 1994.

[14] ITU-T Recommendation GB921 “Enhanced Telecom Operations Map (eTOM): The business Process Framework. TeleManagement FORUM,” Jun 2002.

[15] ITU-T Recommendation M.3100, “Generic Network Model,” July 1995.

[16] ITU-T Recommendation X. 723, “Structure of management: generic management information,” 1993.

[17] ITU-T, Recommendation X.641, “Information technology - Quality of service: Framework,” December 1997.

[18] ITU-T Recommendation X.711, “Information technology - Open Systems Interconnection - Common Management Information Protocol: Specification,” 1997.

[19] S. Kessal, N. Simoni, "A Deployment of Service Elements Based on QoS," services, Proceedings IEEE World Congress on Services, SERVICES 2011, pp. 93-94, July 4-9, 2011, Washington.

[20] E. Marilly, O. Martinot O, S. Betge-Brezetz, and G. Delegue, “Requirements for service level agreement management,” Proceedings IEEE Workshop on IP operation and Management, IPOM 2002, pp. 57-62, 2002, Dallas.

[21] N. Simoni, “Dés réseaux intelligents à la nouvelle génération de services,” ISBN: 978-2-7462-1218-3, Edition Hermes, 2007.

[22] N. Simoni, X. Xiong, and C. Yin, "Virtual Community for the Dynamic Management of NGN Mobility," Proceedings of The Fifth International Conference on Autonomic and Autonomous Systems, ICAS 2009, pp. 82-87, April 20-25, 2009, Valencia.

[23] N. Simoni, “Gestion de Réseau et de service: Similitude des concepts, spécificité des solutions,” ISBN: 2225829802, Edition Masson, 1998.

[24] Telecommunications Information Networking Architecture TINA-C, “Network Resource Information Model,” Version 3.0, 1997.

[25] TeleManagement Forum, GB917, “Service Level Agreement Management Handbook," Volume 2 Concepts and Principles, Release 3.0, May 2010.