Specifying QoS Structural Constraints in RM-ODP

Jalal Laassiri, Member, IAENG, Salah-ddine Krit, Said El Hajji and Mohamed Bouhdadi

***Abstract*—QoS already forms an important part of the studies being carried out by the international standards organizations, ITU and ISO, In order to support interoperability in open distributed systems, an information service is needed that can provide dynamic knowledge about available service providers. So network infrastructure specification, identified by Basic Reference Model of Open Distributed Processing (RM ODP). In RM-ODP a user provider concept is developed as a component standard. Understanding the new network infrastructure specification and how the parameters affect network performance is vital to ensuring excellent performance.**

In this paper, we present an extensive survey of existing QoS specification languages, and new verification for QoS parameters to assure control of the priority, reliability, speed and amount of traffic sending over a network.

We propose a language of specification defined and compliant with the object paradigm, intended to specify QoS in distributed system design and in networked environments.

***Index Terms*— RM-ODP, user- provider, QoS specification, Open distributed processing, UML.**

I. Introduction Background on RM-ODP

One property of a distributed system is that a user of the system is unaware of the differences in computers and operating systems in which their applications run. Such systems are inherently complex. Despite this, distributed processing is growing rapidly, primarily due to the computer industry's ability to produce cheaper, more powerful computers. As a result of this growth, the need for the coordinated production of standards for distributed processing has been identified.

ODP is already a major effort between the ISO and ITU-T which will lead to significant product development in the coming years. The ODP work identifies and attempts to provide a framework for distributed systems. This has been set out in a Reference Model of ODP (RMODP) [1].

It defines an architecture through which distribution, interworking and portability can be achieved. The RMODP recognizes that it cannot provide an infrastructure to meet all of the needs of distribution. Different systems will almost certainly have different demands on the infrastructure.

[1] Laassiri jalal is with the Faculty of Sciences, Department of Informatics, Ibn Tofail University, BP - 33, Kenitra, Morocco (e-mail: Laassiri.jalal @gmail.com)
Krit salah.fr is with the Faculty Polydisciplinary of Ouarzazate Ibn Zohr University- Agadir BP/638 Morocco (e-mail: Krit_salah@yahoo.fr).
El hajji said and Bouhdadi Mohamed are with theFaculty of Sciences Department of Mathematic and Informatics, Laboratory of Mathematic informatics and Applications Mohamed V University-Agdal Rabat BP/1014 Morocco, (e-mail: elhajji@fsr.ac.ma , bouhdadi@fsr.ac.ma).

The RM-ODP is divided into four main parts [1-4].

Part 1 - Overview and Guide to Use: contains an overview and guide to use of the RM-ODP.

Part 2 - Descriptive Model: contains the definition of concepts and gives the framework for descriptions of distributed systems.

Part 3 - Prescriptive Model: contains the specification of the required characteristics that qualify distributed system as open. It defines a framework comprising five viewpoints, five viewpoint languages, ODP functions and ODP transparencies. The five viewpoints are enterprise, information, computational, engineering and technology.

Part 4 - Architectural Semantics: contains a formalization of a subset of the ODP concepts.

The ODP systems offer an infrastructure of connections which will be used by the stream binding object [3]. Which is an ODP common object? ODP aims to provide distribution-transparent utilization of services over heterogeneous environments. In order to use services, users need to be aware of potential service providers and to be capable of accessing them. Since sites and applications in distributed systems are likely to change frequently, it is advantageous to allow late binding between service users and providers. If this is to be supported, a component must be able to find appropriate service providers dynamically. The ODP network infrastructure [3] provides this dynamic selection of service providers at run time.

The languages Z, SDL, LOTOS, and Esterel are used in RM-ODP architectural semantics part [4] for the specification of ODP concepts. However, no formal method is likely to be suitable for specifying every aspect of an ODP system.

Elsewhere, we used the meta-modeling approach [9] [10] to define syntax of a sub-language for the ODP QoS-aware enterprise viewpoint specifications. We defined a meta-model semantics for structural constraints on ODP enterprise language [11] using UML and OCL. We also used the same met-modeling and denotation approaches for behavioral concepts in the foundations part and in the enterprise language [12] [13].

Furthermore, for modeling ODP systems correctly by construction, the current Specifying and Verifying techniques [17] are not widely accepted.

The paper is organized as follows. Section 1 introduces RM-ODP and QoS specification. Section 2 reflects on the experience from this work and describes our view of the implications of distributed systems and mobile computing for RM-ODP standards. Some strategies are also given to aid the development of specific RM-ODP-based platforms for specification of a measurement process. Section 3 describes the subset of concepts considered in this work named the QoS object and reactions patterns. Finally, section 4 presents some concluding remarks.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

II. RM-ODP AND QOS SPECIFICATION

A. RM-ODP

Standardization in the area of Open Distributed Processing is a joint activity by both IS0 and ITU-T. The standard provides a methodology based on the division of a specification into a set of five viewpoints: enterprise, information, computational, engineering and technology by representing the full range of perspectives on a distributed system development from business objectives to detailed implementation choices.

B. Provider: Supporting network infrastructure.

The stream binding object uses the supporting network infrastructure to exchange data flows between users who are geographically separated. The provider offers an infrastructure of connections which will be used by the stream binding object. A provider is an agent that manages the stream binding object in accordance with the status of the network & infrastructure support (Figure 1).

The provider is responsible for the flow topology, billing aspects, security aspects, fault management, and QoS provided by the underlying network and resources [14]. The provider is concerned with the management of the network so that the binding contract is guaranteed. For instance, it selects an appropriate routing for the data channel and reserves resources on each node on that route. If the provider can no longer guarantee the binding contract, it performs actions [16] that affect the stream binding object with respect to the (re)negotiation of the binding contract.

III. QOS SPECIFICATION

Understanding the new network infrastructure specification and how the parameters affect network performance is vital to ensuring excellent performance.

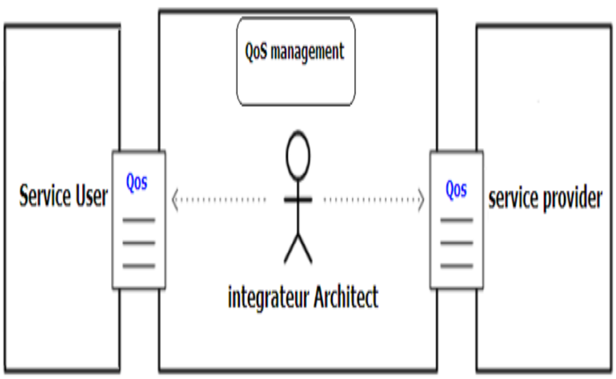

Fig.1. Network infrastructure schema

An application's QoS requirements are conveyed in terms of high-level parameters that specify what the user requires. QoS specification is different at each system layer and is used to configure QoS mechanisms at each layer. Possible system layers are:

Protocols - transport, network ,middleware , operating system - scheduling, resource management, real-time support, distributed platforms - CPU, memory/buffers, devices, application.

QoS specification encompasses requirements for:

Performance - expected performance characteristics are needed to establish resource commitments,

Synchronization - characterizes the degree of synchronization required between related services, events, or information flows,

Level of service - specifies the degree of resource commitment required to maintain performance guarantees, cost of service - the price a user is willing to incur to obtain a level of service,

QoS management - the degree of QoS adaptation that can be tolerated and scaling actions to be taken in the event the contracted QoS cannot be met.

QoS requirements are assessed to determine if they can possibly be met. If, for example, the level of service requested cannot be provided, the user can be asked if a certain level of degradation is acceptable before proceeding further.

A QoS specification is the part of a user-provider related to the QoS issues. In this paper, the ODP components of the QoS specification structure proposed by the QoS management are identified (Figure 2). Each of the components is further elaborated in the following sections.

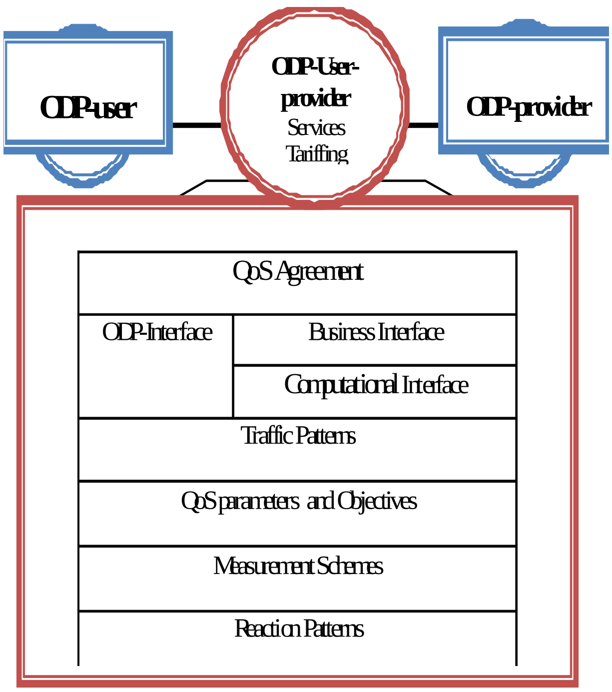

Fig.2 ODP structure of a QoS agreement

A. Interface description

An interface is defined as a logical boundary between two entities and consists of a set of interaction points. Interaction points are always adjacent to the user, but may not be adjacent to the provider. However, all interaction points are controlled by the provider.

The interface description is further categorised into descriptions of a business interface (BI) and a technical interface (TI).

The BI includes all interaction points that enable the unambiguous specification of the QoS agreement

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

components. For example, the BI includes (re)negotiating interaction points where both entities (re)negotiate the user-provider agreement that includes terms and obligations concerning their relationship. It also contains the interaction points used for assessing whether the QoS agreement is fulfilled Interaction points in the BI are always between the user and the provider [20][21][22].

The TI includes all interaction points that enable exchange of information relative to the service. It may include for example: points of interactions for service delivery, QoS measurement points and points where reaction patterns are applied. There may be several such interaction points, which are located between the user and other entities involved in the service provision (the provider or sub-providers). In some cases, the user may even have no technical interaction point with the primary provider.

Exchange of information, related to each phase in the service life cycle, could be associated with a TI. The TI may differ for different phases of the service life cycle.

B. Traffic patterns

The notion of traffic patterns is introduced as a part of QoS agreement in order to cover:

- Describing the characteristics of the expected incoming traffic flows. This information allows an entity to manage resources in its domain in order to deliver the agreed QoS.
- Assuring the understanding of threshold conditions under which reaction patterns may be activated.
- Traffic patterns consider both application and management information flows.

C. Identification of the traffic flows to be characterised

Several traffic flows may cross any given interaction point of an interface between two entities. In particular, flows may either enter an entity's domain (incoming traffic) or exit an entity's domain (outgoing traffic). The outgoing traffic from one entity's domain is the incoming traffic into another entity's domain, which implies that it is necessary to provide traffic patterns for both incoming and outgoing traffic flows (Figure 3).

- Traffic patterns for the incoming traffic characterise the type of traffic flows an entity is expected to support in accordance with a QoS agreement. Given a certain traffic pattern for incoming traffic, the receiving entity has an opportunity to react if the incoming traffic is not conformant with the agreed traffic pattern.
- Traffic patterns for outgoing traffic characterise the type of traffic flows an entity has to deliver at a given interface.

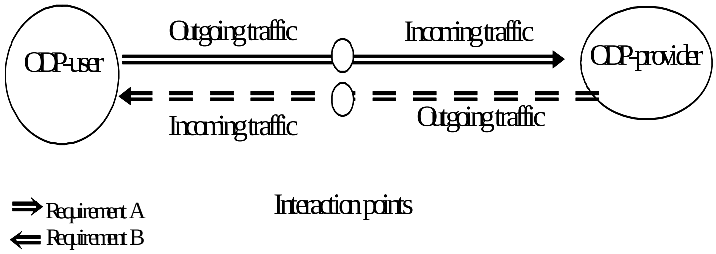

Fig. 3 Traffic flows to be characterised at an interaction point

D. QoS parameters and objectives

In defining the QoS parameters, the interests of both entities should be taken into account. The relevance of individual parameters is assumed to differ during the various phases of a service life cycle. Different viewpoints and instances may then be referred to e.g. the requested QoS, the offered QoS, the contracted QoS, the delivered QoS, ect.

E. Specifying QoS parameters

The performance of a service is expressed by assigning values to a number of QoS parameters [9]. Since each QoS parameter corresponds to the behaviour of a service component between two interaction points, it is necessary, when specifying a QoS parameter, to indicate the interaction points between which it applies and associated objective values.

QoS parameters include primary and derived parameters.

- A primary QoS parameter is determined on the basis of direct observations of "events" at interaction points;
- A derived QoS parameter can be determined as a function of other, previously defined QoS parameters.

If QoS parameters are purely technical/objective, the "events" to be observed may be precisely defined in standards.

Once reference events have been defined, it is possible to specify a correspondence between them, which relates reference events occurring at two interaction points. This process of observing the occurrence of events and of associating corresponding events allows the definition of outcomes [14] which are the data that are considered for defining parameters.

F. Classification of primary QoS parameters:

For a telecommunication service and distributed system in general, defined as a set of functions, primary QoS parameters can be classified.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

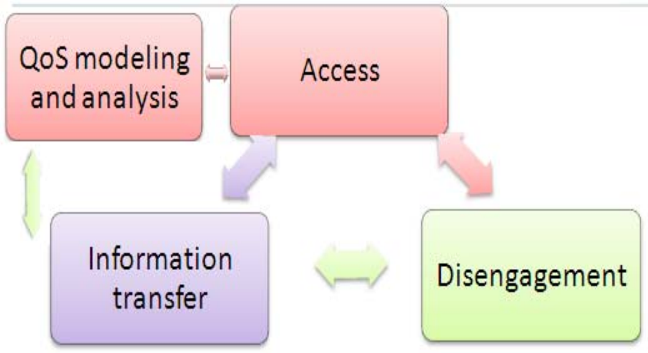

Fig 4: functions and primary QoS parameters

We identify and describe generic function for a given service.

Speed characterises the temporal aspects of QoS associated with a function, showing time related efficiency characteristics. Speed parameters are defined on the basis of statistics made on sets of "duration times" for the reason that is defined as the time elapsed between a pair of corresponding events.

Accuracy characterises the degree of correctness with which a given function is realised. Dependability characterises the degree of certainty that a function is performed.

We illustrate the functional operator of generic primary QoS parameters by (Statistics on service, mean, maximum, quantile on access time or duration of transfer time or disengagement time)

G. QoS objectives

For some of the QoS parameters, QoS objectives can be specified in a QoS. A QoS objective may be a target value on a defined measurement scale that the provider intends to deliver to the user for a given QoS parameter. A QoS objective may also be an upper (or a lower) bound set to a QoS parameter.

Two types of objectives may be specified in a QoS:

- Firm QoS commitments for the provider, usually associated with tight traffic patterns and rigid reaction patterns.
- QoS indications, which are associated with loose traffic patterns and slow reaction patterns. The extreme case is when QoS parameters are "unspecified" and when the provider commits to"best effort" behaviour. In this case, QoS measurements may not be needed.

QoS objectives apply to either static or dynamic scenarios. In the latter scenario, the delivered QoS is allowed to vary within the range defined by the bounds, and the end-user may also adapt to various levels of QoS.

H. Specification of a measurement process

a provider who has offered a QoS guarantee, or a QoS indication to a user, should be able to provide associated measurements of the conformance of the delivered QoS to the agreed QoS. The measures can be obtained by some of the sub-providers. Note that in a multi-provider environment, where measurements are taken at different measurement points, it can be difficult to correlate and merge information collected within different entities.

When defining measurements, the description of a measurement process should cover all the relevant information necessary for setting such a process:

A measurement point is an interaction point at which measurements are performed. It is located at a point, where reference events and/or outcomes can be observed, or approximated. Measurement points may be part of the technical interface if they are located at the interface specified in the QoS agreement.

A measurement is a process by which numbers/symbols are assigned to variables that characterise either traffic or QoS parameters specified for a given service.

A measure is the result of a measurement. It is a value specified on a definite scale, which is assigned to a traffic or QoS parameter.

- Identification of all relevant measurement points,
- Specification of the measurement environment,
- Description of the techniques for obtaining the measured values,
- Specification of the methodology to be used for taking decisions concerning the conformance of the measurement to commitments.

IV. QoS Object and Reaction Patterns

A set of reaction patterns, related to failure to meet either traffic patterns or one/more of the agreed QoS parameter values should be described in the QoS agreement. Some examples of reaction patterns are:

If the agreed traffic patterns are not delivered, the provider may be unable, or unwilling, to provide the agreed QoS levels. Reaction patterns describe the potential provider's reaction in this case (e.g. traffic policing mechanisms).

Users (or objects type) are active entities operating on enterprise objects (passive entities) of the system.

Summing up, a QoS specification is composed of specifications of the elements previously mentioned, i.e. the system's communities (sets of QoS objects), roles (identifiers of behavior), processes (sets of actions leading to an objective), policies (rules that govern the behavior and membership of communities to achieve an objective), and their relationships [15].

A contract specifies obligations, permissions and prohibitions for objects comprising in a communities.

Just as for the objects, the actions are also gathered in processes, this implies that there are two levels of abstraction in ODP enterprise viewpoint:

Abstract level: roles, processes and enterprise viewpoint of the system on which various permissions, prohibitions and obligations are expressed [18][19].

Concrete level: object type (User, Provider, policy maker, policy administrator), actions (create, delete) and QoS objects of the system.

If the agreed QoS levels are not delivered, while the agreed traffic patterns were delivered, the user may react, or expect the provider to react e.g. by applying a compensation

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

scheme. This is also described in the set of reaction patterns associated with the user-provider agreement.

If the provider is unable to deliver all agreed QoS objectives, a reaction pattern may specify what the most important QoS aspects for the ODP-user.

A reaction pattern can be described as a process, including its inputs (QoS or traffic data Obtained from measurements) and outputs (the response, terminating the reaction).

V. Conclusion

This paper has reported on experiences with applying the ISO RM-ODP standards for a Reference Model for Open Distributed Processing in a mobile environment.

RM-ODP provides a framework to enable services to be accessed in a heterogeneous environment spanning multiple organizational domains. Our experiences in the QoS Studies indicate that this QoS specification is sufficiently general to accommodate the particular requirements of distributed applications. In addition, the particular distributed system in RM-ODP, the approach of selective distribution transparency and the concept of QoS-managed bindings all provide strong support for distributed applications.

Our experiences have also shown that the design decisions and address of the syntax and semantics for a fragment of ODP object concepts defined in the RM-ODP foundations -based on The UML 2.0 Infrastructure and the OCL specification and limit its effectiveness in distributed systems environments. In particular, changes were required to the Network infrastructure and to the implementation of trading. The paper concluded with a set of guidelines for future developers of RM-ODP compliant platforms in these and other areas.

References

[1] ISO/IEC, ‘’Basic Reference Model of Open Distributed Processing-Part1: Overview and Guide to Use, ‘’ISO/IEC CD 10746-1, 1994

[2] ISO/IEC, ‘’RM-ODP-Part2: Descriptive Model, ‘’ ISO/IEC DIS 10746-2, 1994.

[3] ISO/IEC, ‘’RM-ODP-Part3: Prescriptive Model, ‘’ ISO/IEC DIS 10746-3, 1994.

[4] ISO/IEC, ‘’RM-ODP-Part4: Architectural Semantics, ‘’ ISO/IEC DIS 10746-4, July 1994.

[5] M. Bouhdadi and al., ‘’A UML-Based Meta-language for the QoS-aware Enterprise Specification of Open Distributed Systems’’ IFIP Series, Vol 85, Springer, (2002) 255-264.

[6] Mohamed Bouhdadi and al. ‘A Semantics of Behavioural Concepts for Open Virtual Enterprises’. Series: Lecture Notes in Electrical Engineering, Vol. 27 .Springer, 2009. P.275-286.

[7] Jalal laassiri and al. A Denotational Semantics of Concepts in ODP Information Language, Proceedings of the World Congress on Engineering 2009 Vol I, WCE '09, July 1 - 3, 2009, London, U.K., Lecture Notes in Engineering and Computer Science, pp. 784-788, 2009

[8] Mohamed Bouhdadi and al., ‘Using BPEL for Behavioural Concepts in ODP Enterprise Language’, Virtual Enterprises and Collaborative Networks, IFIP, Vol. 283, pp. 22 1-232, Springer, 2008

[9] Mohamed Bouhdadi and al., ‘Meta-modelling Syntax and Semantics of Structural Concepts for Open Networked Enterprises’, Lecture Notes in Computer Science, Vol. 4707, pp. 45-54, Springer, 2007.

[10] Myers, G. The art of Software Testing, John Wiley &Sons, New York, 1979

[11] R. Binder ‘Testing Object Oriented Systems. Models’. Patterns, and Tools, Addison-Wesley, 1999.Klara Nahrstedt and Jonathan M. Smith. The QoS Broker. IEEE Multimedia Magazine, 2(1):53–67, 1995.

[12] Binder, R. Testing Object Oriented Systems. Models. Patterns, and Tools, Addison-Wesley, 1999.

[13] Said El hajji and al: ‘Using Event B to specify QoS in ODP Enterprise language’. PRO-VE'10 11th IFIP Working Conference on VIRTUAL ENTERPRISES, Saint-Etienne, France, 11-13 October 2010.

[14] ISO/IEC. RM-ODP Enterprise Language. Draft International Standard ISO/IEC 15414, ITU-T X.911, ISO, 2001.

[15] J.Laassiri and al: ‘Specifying Behavioral Concepts by engineering language of RM-ODP’, Journal of Theoretical and Applied, Information Technology (JATIT), pp. 1 to 10 Volume (15). 2010.

[16] A.Naumenko, A.Wegmann, ‘Proposal for a formal foundation of RM-ODP concepts’ conference woodpecker 2001.

[17] Myers, G. The art of Software Testing, John Wiley &Sons, New York, 1979.

[18] J. Laassiri and al ,’Using BPEL for Behavioral Concepts in ODP Computational Language’ , Proceedings of the World Congress on Engineering 2009 Vol I, WCE '09, July 1 - 3, 2009, London, U.K., Lecture Notes in Engineering and Computer Science, pp 162-168, 2009.

[19] J. Laassiri and al, ‘ Event B for ODP Enterprise Behavioral Concepts Specification’, Proceedings of the World Congress on Engineering 2009 Vol I, WCE '09, July 1 - 3, 2009, London, U.K., Lecture Notes in Engineering and Computer Science, pp 784-788, 2009.

[20] Eurescom Project P1008, "Inter-operator interfaces for ensuring end-to-end IP QoS", Deliverable 2, Selected Scenarios and requirements for end-to-end IP QoS management,2001.

[21] JP Vasseur, Arthi Ayyangar, "Inter-area and Inter-AS MPLS Traffic Engineering", IETF Internet Draft, 2004.

[22] JL Le Roux, JP Vasseur, Jim Boyle, "Requirements for Inter-area MPLS Traffic Engineering", Internet Draft, TEWG Working Group, 2004.

[23] Raymond Zhang, JP Vasseur, "MPLS Inter-AS Traffic Engineering requirements", Internet Draft, TEWG Working Group, 2004.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)