Sixth LACCEI International Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2008)
"Partnering to Success: Engineering, Education, Research and Development"
June 4 – June 6 2008, Tegucigalpa, Honduras.

Quality view of Web Services domain

Francisca Losavio
Universidad Central de Venezuela, Caracas, Venezuela, flosav@cantv.net

Alfredo Matteo
Universidad Central de Venezuela, Caracas, Venezuela, almatteo@cantv.net

Roberto Rahamut
Universidad Central de Venezuela, Caracas, Venezuela, rrahamut@cantv.net

ABSTRACT

Web services are an alternative to the lack of interoperability of Web applications, to facilitate integration via Service Oriented Architecture (SOA). An important aspect in developing WS-based applications is to consider the client requirements, where elicitation, analysis and specification are main concerns to ensure the WS quality as a software component, as well as the overall quality of the application using the WS. Requirements specification and in particular, quality requirements specification, are still open research issues. The main goal of this paper is to characterize early the WS applications domain using the architectural knowledge, focusing on a standard specification of the quality properties related to requirements. This characterization considers a WS categorization based on functionality and the following frameworks: WSA (Web Services Architecture) base-line of the W3C (World Wide Web Consortium), the standards ISO/IEC 9126-1 to specify the domain quality properties and ISO/IEC 13236 to specify the QoS metrics. The importance of this quality-based domain characterization is the automatic generation of standard contractual bases between WS clients and providers. Moreover, guidelines for traceability among the standards are provided, towards the establishment of a common language for the WS community. Finally, the applicability of our approach is illustrated for transactional WS.

Keywords: web services requirements, domain characterization, quality model, ISO/IEC 9126-1, ISO/IEC 13236.

1. INTRODUCTION

Web Services (WS) are reusable software components accessible over the standard Internet Protocol (IP), allowing interoperability and portability in Web applications, according to a Service Oriented Architecture (SOA) style. A WS is considered a software component offering a service, i.e. providing a kind of functionality, characterized by its functional and nonfunctional requirements. Different types of WS can be grouped on the basis of the functionality or service they provide. WS-based software development considers component-based programming platforms, providing modular and reusable structure, independent from the technological details, such as operating system and data format (Mani et al., 2002; Caudwell et al., 2001); however, within this scheme contractual issues have to be established and respected.

The architecture underlying WS applications is a layered SOA style instance for components, and a message-passing style, following a Peer-to-Peer pattern (W3C. Web Services Description Requirements, 2002), for connectors. A reference architecture has been defined by the Web Services Architecture Working Group (WSA) to guarantee the interoperability of WS-based applications. It will be called the WSA base-line architecture in this context. According to the W3C (WWW Consortium), a WS is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format, specifically Web Service Description Language (WSDL). Other systems interact with the WS in a manner prescribed by its interface description using SOAP (Simple Object Access Protocol) messages,

typically conveyed by HTTP with an XML serialization in conjunction with other Web-related standards. The main principles to construct WS are based on three aspects or mechanisms, location, description and call, and follow the XML universal data format (Manes, 2003; Caudwell et al., 2001). The WSA requirements document of the W3C describes seven critical top-level quality goals that are the minimal set of requirements for a common architecture that a WS-based application should comply (W3C. Web Services Architecture Requirements, 2004), those are: *Interoperability, Reliability, WWW Integration, Security, Scalability and Extensibility, Team Goals and Management and Provisioning.*

A broad initial characterization of the domain of WS-based applications is given by identifying the quality goals for the base-line architecture, shared by a family of applications within the domain, independently from the application functionality. The critical goals are in general provided by the market standards. From (Berard, 1992), Berard defines a domain as the minimal set of properties that accurately describe a family of problems for which computer applications are required. In this sense, the base-line or reference architecture is the solution to a family of problems.

The main goal of this work is to establish a standard characterization of the WS-based applications domain in the sense of Berard (Berard, 1992). A process is defined which on one hand establishes a correspondence between the WSA top-level critical goals of the W3C and the ISO/IEC 9126-1 standard (ISO/IEC 9126-1,2, 1999) for software product quality, to facilitate a common understanding of the architectural properties. This standard can be easily adapted to specify architectural properties, considering the architecture as an intermediate product of the software development process (Losavio et al., 2004; Sommerville, 2000; Thayer et al., 2000). Moreover, the quality requirements associated with the functionality of each WS type, defining sub-families of applications, are also specified by the ISO/IEC 9126-1 quality model. On the other hand, the ISO/IEC 13236 standard (ISO/IEC 13236, 1999) which specifies the quality of the services is used to specify the measurable attributes, according to general quality of service metrics. In consequence, a standard quality model is defined for the WS application domain. In what follows, the main concepts on the standards ISO/IEC 9126-1 (ISO/IEC 9126-1,2, 1999) and ISO/IEC 13236 (ISO/IEC 13236, 1999) are presented.

1.1. THE ISO/IEC 9126-1 STANDARD

The achievement of quality software products is the goal of software engineering. The quality properties related to functional and non functional requirements are called quality requirements and are specified in the standard as quality characteristics and sub-characteristics. They are classified into three views of quality for a software artifact or product: external quality (considered within a testing environment), internal quality (considered within the development process) or in use quality (considered by the final user). ISO/IEC 9126-1 defines a hierarchical quality model based on six main high level quality characteristics. They are refined into sub-characteristics, continuing with the refinement until attaining the attributes or measurable elements. A sub-characteristic can then be associated to several attributes and metrics or counting rules can be associated to each attribute; this process is part of the so called measurement model. We use here the external/internal quality views, which are given below: *Functionality (*Suitability, Accuracy, Interoperability, Security*), Reliability(*Maturity, Fault tolerance, Recoverability, Compliance*), Usability (*Understandability, Learnability, Operability, Attractiveness*), Efficiency (*Time behavior, Resource utilization*), Maintainability (*Analyzability, Changeability, Stability, Testability*), Portability (*Adaptability, Installability*).*

It is clear that this framework must be adapted for each specific domain, in particular for WS. Adaptation is obtained eliminating non applicable characteristics and adding new sub-sub-characteristics if required.

1.2. THE ISO/IEC 13236 STANDARD

The goal of the ISO/IEC 13236 standard is to assist in the specification and design of technology-based software systems; it describes how to characterize, specify and manage requirements related with the quality of the service (OoS). It provides a common language to services, clients and providers. According to ITU (ITU, 2002), QoS is defined as a set of quality requirements present in the collective behavior of one or more objects. Note also that a requirement originates from a client entity that uses a service and it is translated to different QoS (ISO/IEC 13236, 1999) requirements, expressed as parameters. A mechanism is realized for the entity to satisfy one or more QoS

parameters. Mechanisms are part of the management functions and parameters are part of the context of a QoS. The range of values for the attributes (value of QoS parameters) is established by the metrics for the WS quality requirements external view. This paper is structured as follows, besides this introduction: a second section presents the process to characterize the WS application domain. According to the process guidelines, a general classification of WS based on their functionality is given. The standard quality model is then constructed establishing a correspondence with the WSA requirements and completing it with the quality requirements related with the WS functionality. The model is refined until attributes and metrics for a Transactional WS are obtained. The application of the process to transactional WS is the case study presented in the third section, where the contractual agreements are automatically generated from the quality model specification. The fourth section reviews some related works. Finally, the last section presents the conclusion and future work.

2. CHARACTERIZATION OF THE WS DOMAIN

For a standard quality-based characterization of the domain of WS applications, the following process is proposed:

(i) Define functionality. Establish a classification of WSs, according to functionality.

(ii) Define Quality Model. Specify the quality requirements for the domain (family of WS-based applications) considering the ISO/IEC 9126-1 standard, as follows:
   (a) Specify architectural quality. Use the WSA critical goals (W3C, Services Architecture Requirements, 2004), establishing a correspondence with the ISO/IEC 9126-1 sub-characteristics and sub-sub-characteristics, to obtain a standard quality model for WSA.
   (b) Specify functional quality. For each type of WS, quality properties are assigned to the functional requirements to express the goals that must be fulfilled. Standard quality requirements are obtained for each type of WS. New sub-characteristics or sub-sub-characteristics can be added accordingly, if needed.

(iii) Define Measurement Model: For each type of WS, refine the quality model obtained in step 2, by specifying the attributes and metrics for each type of WS, obtaining a characterization of a sub-family of the domain, as follows:
   (a) Specify quality attributes and metrics for each sub-characteristic or sub-sub-characteristic of the quality model.
   (b) Assign the attributes (QoS characteristics) and the corresponding metrics, according to the ISO/IEC 13236 standard.

The process described provides guidelines that can be practically applied to facilitate traceability among the standards, which in the literature appear separate and to settle the basis for the automatic generation of a standard contractual specification between WS clients and providers. The guidelines can be used as a starting point for a common language for the WS community. Each step is detailed in what follows.

2.1. STEP 1. DEFINE FUNCTIONALITY. CLASSIFICATION OF WS.

The classification presented in Table 1 shows the functional requirements for WS (Thomas, 2003; Ginige et al., 2001). Metrics must be provided as part of the contractual issue to establish the extent to which the WS fulfills its functionality (Menascé et al., 2000; Caudwell et al., 2001) and "See Section Step 3". Note that in this taxonomy, the types of WS are not necessarily disjoint. For example, a security WS can be used by a transactional WS. This aspect is known as orchestration (W3C, Web Service Modeling Ontology, 2002), where a service can use another to accomplish the required functionality.

Table 1. Functionality based WS classification.

WS type	Functional requirements	Example of applications
Information and collaborative environments	Data Base operations	Product catalogues, discussion groups, books, white papers.
Transactional	E-commerce operations, encrypting	E-commerce . On-line banking
Workflow	Process monitoring operations	On-line planning. Scheduling/Management systems.
Web Portal	E-search and e-communication	Search engines. E-shopping centers.
Security	Access control, encrypting	Authentication and authorization. Encrypting, System access.

2.2. STEP 2. DEFINE QUALITY MODEL

The quality model specifies a minimal set of properties characterizing applications; all the applications within this domain will share these properties. The quality requirements for the family of WS-based applications are specified using the ISO/IEC 9126-1 standard. A correspondence with the WSA critical goals (W3C, Web Services Architecture Requirements, 2004) is established to specify architectural quality in terms of sub-characteristics and sub-sub-characteristics. Hence, the standard quality model is customized to WSA goals.

Table 2. Quality Model for WS-based application domain, showing traceability among ISO/IEC 9126-1 and WSA. The codes of the WSA goals are taken from (W3C, Web Services Architecture Requirements, 2004)

ISO/IEC9126-1 characteristics	Correspondence between ISO/IEC9126-1 sub-characteristics and WSA critical goals		
Functionality It refers to the capability of fulfilling the proposed tasks under specific operation conditions.	**Interoperability**		The capability of software to interact with other software, without adding new functionality (ISO 9126-1). According to WSA, WS must be integrated to the Web and consistent with its architectural principles, allowing interoperability within this environment. Semantics is similar. ***Goal: high***
	ISO/ IEC	Interoperability	
	WSA	Interoperability (AG001)	
	Security		The capacity of software to protect information and data from non-authorized access, as well as to facilitate authorized access (ISO/IEC 9126-1). According to WSA, is to provide a secure environment for on-line processes, habilitate client protection and privacy, facilitating an easy access. Semantics is similar. ***Goal: high***
	ISO/ IEC	Security	
	WSA	Security (AG004)	
	Suitability		According to ISO/IEC 9126-1, is to provide the appropriate set of functions for the user specific tasks. According to WSA, the architecture must satisfy the user and the community needs. Semantics is similar. ***Goal: medium***
	ISO/ IEC	Suitability	
	WSA	Team Goals (AG006)	
Reliability It refers to the capability of software of maintaining a certain performance level under established operation conditions.	**Availability**		According to ISO/IEC 9126-1, *availability* is the combination of *maturity* (to avoid failures), maintaining a level of performance without failures (*fault tolerance*) and the capacity to recover from failures (*recoverability*). According to WSA, is the availability and stability of the service. Semantics is similar. ***Goal: high***
	ISO/ IEC	Availability	
	WSA	Reliability (AG006)	
Maintainability It refers to the capability software of being modified: corrections, improvements, changes with respect to requirements.	**Extensibility**		According to ISO/IEC 9126-1, the capacity to support changes (scalability refers to changes in volume of transactions). According to WSA, it allows extensible and scalable applications. The WSA extensibility goal is considered as sub-sub-characteristic of changeability. **Goal: high.**
	ISO/ IEC	Changeability	
	WSA	Scalability and Extensibility (AG006)	
	Management and provisioning		ISO/IEC 9126-1 refers to changeability as the capacity to support changes in general. According to WSA, it is the capacity to add or delete components without affecting the performance of the application. The WSA management and provisioning goal is considered a sub-sub-characteristic of changeability **Goal: high.**
	ISO/ IEC	Changeability	
	WSA	Management and Provisioning (AG007)	
	Integration		According to ISO/IEC 9126-1, changeability is the capability to support changes in general. According to WSA, it is the capacity to be coupled with the Web evolution. The WSA integration goal is considered a sub-sub-characteristic of changeability. **Goal: medium**
	ISO/ IEC	Changeability	
	WSA	Integration (AG003)	
Portability It refers to the capacity of software to be transferred from an environment to another: organizational, software or hardware.	**Scalability**		According to ISO/IEC 9126-1, adaptability implies platform independence (operating system or particular technology). WSA refers to scalability and extensibility. Scalability in ISO/IEC 9126-1 is included into adaptability and it refers to the software capability to adapt to changes in the display size or volume of transactions. The WSA scalability goal will be considered a sub-sub-characteristic of adaptability. **Goal: medium**
	ISO/ IEC	Adaptability	
	WSA	Scalability and extensibility (AG006)	

The quality model shows the minimal characteristics that providers must comply to guarantee user satisfaction. These properties are part of the contractual agreement for the control and measure procedures. In consequence, a WS must satisfy some of the quality properties indicated in Table 4. According also to this table, a WSA compliant service is now also compliant with the ISO/IEC 9126-1 standard characteristics (sub-characteristics and sub-sub-characteristics) for internal/external product quality, to which a high, medium or low goal ranking has been assigned by consensus by an expert group. Usability and efficiency are ranked low and are not shown in the table, because they are not present as WSA critical goals (W3C, Web Services Architecture Requirements, 2004). In consequence, the quality model for WS domain considering relevant architectural properties is conformed only by characteristics ranked high or medium, which are the following: functionality (interoperability, security, suitability), reliability (availability), maintainability (changeability (extensibility, management and provision)), portability (adaptability (scalability)). Notice that the WSA critical goals extensibility and management and provision are shown as sub-sub-characteristics of maintainability.

Table 3. Quality requirements for each WS type.

WS type	**Functional requirements**	**Quality requirements for sub-families of WS-based applications - characteristics and sub-characteristics according to ISO/IEC 9126-1**				
		Functionality	Reliability	Maintainability	Portability	**Efficiency**
Information and collaborative environments	Data Base operations: query, access, modification, exchange	**- accuracy**	- availability	- changeability		
Transactional	e-commerce operations: data exchange, access control, encrypting	- security (integrity) **- accuracy**	- availability			**- time behavior - resource utilization**
Workflow	Process monitoring operations: Control, planning	- suitability				
Web Portals	E-search and e-communication operations: consult, access				- adaptability: scalability	**- time behavior - resource utilization**
Security	Integrity operations: access control, encrypting	- security				

Now, for each type of WS, quality properties are assigned to the functional requirements to express the goals they should fulfill. New sub-characteristics or sub-sub-characteristics can be added accordingly, if needed. Besides the WSA requirements concerning the architecture of the services, the quality related to the functionality of the WS must be also considered for the quality model of the domain. Table 3 shows that efficiency for Transactional and Web portals and accuracy for Transactional and Collaborative environments were not considered as WSA requirements (W3C, Web Services Architecture Requirements, 2004), since they are concerned on how the functionality must be accomplished. Observe that the fulfillment of some of the quality requirements imply a commitment or trade-off with other requirements. For example, using a message queue can improve reliability for remote WS access, compromising efficiency (response time).

Notice also that Table 3 is useful to facilitate the early identification of crosscutting concerns (Moreira, 2002). In particular, the security quality property appears related to the access control and encrypting functionality, in applications using Transactional and Security WS. An additional encrypting or control access WS could be suggested in such situations to handle properly code scattering.

The WSA quality model, enriched with qualities related to WS functionality, constitutes the standard quality model for the WS-based application domain. The enrichment is obtained considering all the quality characteristics

shown before for WSA and adding the quality characteristics (shown in boldface), derived from the functionality of each WS, shown in Table 3: efficiency (time behavior, resource utilization) is required for some of the WS and the sub-characteristic compliance to standards and regulations is required to achieve interoperability, in order that the service conforms to standards like SOAP, UDI, WSDL in their respective versions. We assume that this characteristic is present for all WS types and so it is not specified in Table 1. Sub-characteristic accuracy has been included for data transactions indicating the precision of an event, set of events, condition or data (ISO/IEC 13236, Quality of Service, 1999). Notice that often the term integrity is used in WS transactions to denote the fact of maintaining the correction of the transaction with respect to the source, which we are considering in the model as suitability.

2.3. STEP 3. DEFINE MEASUREMENT MODEL.

In this work, the quality requirements of the service (QoS), which are quantifiable aspects or parameters, are considered attributes of the sub-characteristics of the WS domain quality model.

Observe that traceability between the standards ISO/IEC 13236 (ISO/IEC 13236, Quality of Service, 1999) and ISO/IEC 9126-1 (ISO/IEC 9126-1,2, 1999) is not explicitly provided by the standards making difficult their practical usage. This work is a contribution towards the fulfillment of this gap. It is clear that the metrics presented are quite general and should be customized to establish the contractual part when using the service in a particular application.

In what follows, the quality model is further refined for each WS type of Table 3. The refinement consists in finding the attributes or measurable elements and their metrics, for each WS quality property. These attributes and metrics correspond to the QoS characteristics considered in the ISO/IEC 13236 standard (ISO/IEC 13236, Quality of Service, 1999).

Table 4. Measurement Model: QoS metrics for Transactional WS.

WS	Quality characteristics and sub-characteristics according to ISO/IEC 9126-1		Attributes (QoS characteristics, according to ISO/IEC 13236)	Definition, according to ISO/IEC 13236	Metrics, according to ISO/IEC 13236
Transactional WS	Functionality	Compliance with standards and regulations for Interoperability	Depending on the regulations	Depending on the regulations	Depending on the regulations
		Security	protection	Security related with a resource or information.	Probability of failure of the protection. Degree to which a service provider attempts to counter security threats using security services.
			access control	Protection against unauthorized access to a resource	Value or level derived from an access control policy.
			data protection	Protection against unauthorized access to data	Value or level derived from the data integrity policy.
			confidentiality	Protection against unauthorized viewing to data	Value or level derived from the data confidentiality policy.
			authenticity	Protection for mutual authentication and data origin authentication	Value or level derived from the data authentication policy.
		Accuracy	accuracy: {addressing, delivery, transfer, transfer integrity, allowable, release establishment} error	The correctness of an event, set of events, a condition or data. It refers to the integrity of the user information only	Probability

Table 4 (Cont.). Measurement Model: QoS metrics for Transactional WS.

WS	Quality characteristics and sub-characteristics according to ISO/IEC 9126-1		Attributes (QoS characteristics, according to ISO/IEC 13236)	Definition, according to ISO/IEC 13236	Metrics, according to ISO/IEC 13236
Transactional WS	Reliability	Availability	fault-tolerance	Mean Time Between Failures	Average time it takes for the system to fail plus the average it takes to recover. MTBF[a] =MTTF[b] + MTTR[c]
			fault-containment	The ability to operate in presence of one or more errors/faults	Probability
			resilience, recovery error	The ability to recover from errors (Recoverability).	Probability
			Agreed service time (channel, connection, processing)	Proportion of agreed service time for which satisfactory service is available	A=MTBF/(MTBF+MTTR) when maintainability is involved, $0 \leq A \leq 1$
	Efficiency	Time behavior	date/time	The absolute time in which an event occurs	Any unit of time
			time delay: transit, request/reply, request/confirm	The elapsed time between two general events E1, E2, occurring in times T1, T2	TD=T2- T1
			lifetime	The period of time in which a data is valid.	Any unit of time
			remaining lifetime	The time remaining before the data ceases to be valid.	Any unit of time
			freshness (or age of data)	The time since the data was produced	Any unit of time
			capacity	The amount of service that can be provided in a specific period of time	Any unit of time
		Resource utilization	throughput (communication capacity)	The rate of user data output from a channel, averaged over a time interval	Units depend on the resource type
			processing capacity	Amount of processing that can be performed in a period of time	Rate (bits/seconds, bytes/seconds)
			operation loading	Proportion of capacity being used in a period of time	Instructions/seconds - Relation between used and available capacity

Note: Table notes.
[a] MTBF: mean time between failures.
[b] MTTR: mean time to replace.
[c] MTTF: Mean Time to Failure.

Table 4 shows the refinement for the transactional WS category, since this is a complex WS type. The other refinements can be obtained in a similar way and they will not be shown here to ease the presentation. It must be noticed that Table 4 only shows some of the attributes, to facilitate legibility. They must be used depending on the application requiring the WS and on what is to be measured. For detailed information on attributes and metrics the ISO/IEC13236 (Losavio et al., 2005; ISO/IEC 13236, Quality of Service, 1999; Chirinos et al., 2006) standard document should be consulted.

3. CASE STUDY: VOICE PORTAL WS FOR AIRLINE COMPANIES.

A Voice Portal (VP) outsourcer, AIRPORTAL, offers a suite of Web Services (WS) based on a self service platform which combined with IP Telephony and Open Standards, like VoiceXML , providing a powerful speech and touch-tone solutions for airline companies. AIRPORTAL enables integration through interoperability with its standardized approach to voice-based applications and common airline systems, enriching and refining a caller's experience using simultaneous voice and data interactions. Functionalities like phone-based booking for travel (make reservations, cancellation, blocking and confirmation transactions), user registration, online ticketing (purchasing, payment, status), query flights information, 24/7 speech automation of airline-related call center routines, millage account balance, among others, can be offered by AIRPORTAL, through Voice Portal Web Services (VP-WS), to give their customers the highest levels of self-service whether they are using a telephone, a computer or a mobile phone. In order to characterize the domain for VP-WS, the process proposed on Section 2 is applied. Based on the WS functionality of VP-WS, AIRPORTAL is a provider of Transactional Web services (see Table 1).

AIRLINE is a regular AIRPORTAL's customer. AIRPORTAL, rather than acquiring and owning the infrastructure for hosting these WS, seeks out a provider of computational services called AIRSERVICE. It supplies to AIRPORTAL the computing resources needed to host the VP-WS, which include VP applications hosting and auditing (monitoring, calculation, notification), call center agents on-line reporting, storage systems, data base hosting, server-rental, networking components, and Internet connectivity (Dan et al., 2004; Menascé, 2000). Moreover, AIRSERVICE is involved in the AIRPORTAL reservation transaction offering auditing services for this request. Then, AIRPORTAL gather the flight information. This process contains auditing services, performed by AIRSERVICE, like the SLA metrics calculation. Finally is checked the SLA and the information requested is send to the AIRLINE. A contract between AIRPORTAL and AIRLINE is then established, this SLA (Service Level Agreement) specifies a minimal set of properties and characteristics of the provided voice portal services, such as: average conversation time, volume of abandoned calls, call center agents occupation, transactions successfully executed, among others. In this case the SLA is expressed via the Web Service Level Agreement (WSLA) language (Dan et al., 2004; Menascé et al., 2000). One of the functional and non functional requirements is:

- *Reservation Transactions*: At least 95% of the reservation requests shall always provide real time information about flight availability information, 80% of the confirmation volume is accepted, a lower value is considered a fault and the call must be delivered directly to an agent. Those requests should have a server-side response time of 10 seconds.

The general information for SLA is contained in the Measurement Model for WS domain (see Table 4). Using the Quality Model of the domain for WS (See Table 1), the characteristics that providers must comply to guarantee *user satisfaction* can be identified. Then, instantiating the Measurement Model for the AIRLINE SLA, the attributes or measurable elements and their metrics are defined (Chirinos et al., 2005; ISO/IEC 13236, 1999; Losavio et al., 2006) and how they are going to be measured and presented in the contract.

4. RELATED WORKS

In general, standards are used in manufacturing, but not much has been done at the early stages of software development. Moreover, the use of standards is not easy, since they often lack of guidelines or rationale. We have successfully used the ISO 9126-1 standard to specify architectural properties and found it a useful tool, even if lack of flexibility is claimed. We favor the fact that is important to speak a common language (Losavio et al., 2003; Berard, 1992; De Champeaux et al., 1993), particularly the unification of different standards is a goal of this work. The goal-oriented approaches to build architectures from requirements could easily use such standards, however they do not (Gross, 2001; Weiss, 2003). To characterize early the crosscutting concerns, some attempts have been made that use a similar approach (Navarro et al., 2004; Moreira et al., 2002). The characterization of the application domain (De Champeaux et al., 1993) is crucial for architectural design and in particular for product line architectures, where the identification of family of applications is crucial. Moreover, the handling of domain knowledge is recently gaining importance to break the gap between the modeling of business and system

requirements; it is the starting point to define a first rough architecture for the system, where the specification of business requirements is crucial. The ISO/IEC 9126-1 quality model has demonstrated to be a useful tool at this stage.

A toolkit to generate the contract file in XML format is publicly available on the IBM alphaWorks™ Web site (Dan et al., 2004). This utility is part of the IBM Emerging Technologies Tool Kit and serves as a guideline to learn how to specify the SLA using Web Service Level Agreement (WSLA) language. Our toolkit automatically generates the WSLA based contractual agreement, providing also the standard default QoS value or range (see table 3) for the specific WS type.

5. CONCLUSION AND FUTURE WORK

The domain of WS-based applications has been characterized in this work. The standard specification of the quality properties related with a WS has been emphasized. To establish this characterization, a process has been proposed: first, take into account the WSA (Web Services Architecture) critical goals (W3C, Web Services Architecture Requirements, 2004; W3C, Web Services Description Requirements, 2002). The WSA marketing reference establishes the minimal set of requirements that a family of WS applications must hold. A quality model is established for this domain, according to ISO/IEC 9126-1 (W3C, Web Services Description Requirements, 2002) Secondly, the functionality of WS is identified; for each functionality, quality properties and goals are established. The initial quality model is enriched, to characterize the family of WS, focusing also on the quality properties inherent to the functionality. Finally, this quality model is instantiated for families of WS, considering the attributes or QoS metrics, according to ISO/IEC 13236 (ISO/IEC 13236, 1999). In this way, three separate standards have been related and put into a practical use. Common understanding among different stakeholders has been set by this correspondence. As a case study, a family of Transactional WS has been considered, to show the final quality model with metrics; the process is similar for any type of WS. The quality model established is a reusable artifact and can be customized to any WS-based application and family of WS. Ongoing research works are:

- On one hand the definition of a specification pattern based on the contractual Web service level specification language (Web Service Level Agreement Language Specification, WSLA); this pattern will facilitate the automatic identification of requirements that must be considered, as well as the metrics that must be calculated to guarantee the commitment. The contractual specification based on quality issues allows guaranteeing the functionality of the services, since it can be monitored by notifying mechanisms that take actions accordingly, if required.
- On the other hand, we are working further on the domain characterization using a standard quality model, often product of the combination of different standards, as a specification tool for quality concerns. This model is used as a common language between different stakeholders and can be used to monitor quality through the development process, considering the different quality views. In an architectural design approach (Losavio et al., 2005) it is used to define the properties that the initial system architecture must comply.

REFERENCES

Berard, E. (1992). "*Essays in Object-Oriented Software Engineering*", Prentice Hall, 1992.

Caudwell, P., Chaawla, R., Chopra, V., Damschen, G., Dix, C., Hong, T., Norton, F., Ogbuji, U., Olander, G., Richman, M., Saunders, K., Zaev, Z.. "*Servicios Web XML*". (2001). Ediciones ANAYA.

Chirinos, L., Losavio, F., Boegh, J. (2005). "*Characterizing a Data Model for Software Measurement*". (2004). Journal of Software Systems, Elesevier Science, vol. 74, pp. 207-226.

Dan, A., Davis, D., Kearney, R., Keller, A., King, R., Kuebler, D., Ludwig, H., Polan, M., Spreitzer, M., Youssef, "*A. Web services on demand: WSLA-driven automated management"*. 0018-8670/04/$5.00 © 2004 IBM. IBM SYSTEMS JOURNAL, VOL 43, NO 1, 2004. http://www.research.ibm.com/journal/

De Champeaux, D., Lea, D., Faure, P. (1993). "*Object-Oriented System Development*", Addison-Wesley

Ginige, A., Murugesan, S. (2001). "*Web Engineering: An Introduction*". University of Western Sydney, Australia. IEEE Multimedia, vol. 8, no. 1, Jan-March, 2001, pp. 14-47.

Gross, D., Yu, E. (2001). "*From Non-Functional Requirements to Design Through Patterns*", Requirements Engineering Journal, vol. 6, pp. 18-36

ISO/IEC 9126-1, 2. "*Quality characteristics and guidelines for their use* (1, 2), (1999). International Organisation for Standardisation / International Electrotechnical Commission.

ISO/IEC 13236. "*Quality of Service: Framework*. Version 1". (1999). International Organisation for Standardisation / International Electrotechnical Commission.

ITU, International Telecommunication Union. "Framework *for QoS routing and related traffic engineering methods for IP-, ATM-, and TDM-based multiservice networks*". E.360.1 (2002). http://www.itu.int/home/

Losavio, F., Chirinos, L., Lévy, N., Ramdane-Cherif, A., (2003). "*Quality Characteristics for Software Architecture*", Journal of Object Technology (JOT), March/April 2003. vol 2, no. 2, pp.133-150. http://www.jot.fm/issues/issue_2003_03/article2

Losavio, F., Chirinos, L., Matteo, A., Lévy, N., Ramdane-Cherif, A. (2004) "*ISO quality standards to measure architectures*", Journal of Systems and Software, Elsevier Science, vol. 72, pp. 209-223

Losavio, F., Lévy, N., Davari, P., Colonna, F. (2005), "*Pattern-Based Architectural Design Driven by Quality Properties: A Platform for Scientific Calculatiom*, 2nd. European Workshop, EWSA, Pisa, Italy, June 2005, LNCS 3527, pp. 94-112.

Losavio, F., Matteo, A., Rahamut, R. "*Caracterizacion De Aplicaciones Con Servicios Web Basada En Calidad"*. CIAWI 2006 – IADIS. Conferencia Ibero Americana WWW/Internet 2006. Murcia, España., October 2006, pp. 90-95.

Manes, Thomas A. "*Web Services. A Manager's Guide*". (2003). Addison Wesley.

Mani, A., Nagarajan, A.. "*Understanding quality of service for Web services*". (2002) IBM Software Labs, India.

Menascé, D., Almeida, V.. (2000). "*Cappacity Planning and Performance Modeling*", Prentice Hall.

Moreira, A., Araujo, J., Brito, I. (2002) "*Crosscutting Quality Attributes for Requirements Engineering*", SEKE, pp. 167-174

Navarro, E., Letelier, P., Ramos, I. (2004). "*Goals and Quality Characteristics: Separating Concerns*", Spanish CICT, DYNAMICA project, www.um.es/giisw/i-oficial/proyectos.htm

Sommerville, I., Sawyer, P. (2000). "*Requirements Engineering*, A Good Practice Guide".

Thayer, R., Dorfman, M. (2000). "*Software Requirements Engineering*", IEEE Computer Society.

Weiss, M. (2003). "Pattern-driven Design of Agent Systems: approach and case stud", proceedings caise03.

W3C, World Wide Web Consortium. *Voice Extensible Markup Language (VoiceXML)*. Version 2.0. W3C Recommendation 16 March 2004. Copyright © 2004 W3C® (MIT, ERCIM, Keio), All Rights Reserved W3C. http://www.w3.org/TR/voicexml20/

W3C, World Wide Web Consortium. *Web Services Architecture Requirements*. W3C Working Group Note 11 February (2004). Copyright © 2004 W3C® (MIT, ERCIM, Keio). http://www.w3.org/TR/wsa-reqs

W3C, World Wide Web Consortium. *Web Services Description Requirements*. W3C Working Draft 28 October (2002). Copyright © 2002 W3C® (MIT, ERCIM, Keio). http://www.w3.org/TR/ws-desc-reqs

W3C, World Wide Web Consortium. *Web Service Modeling Ontology*. W3C Member Submission 3 June 2005. W3C Working Draft 28 October (2002). Copyright © 2005 W3C®. http://www.w3.org/Submission/WSMO/

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.