A XML-BASED QUALITY MODEL FOR WEB SERVICES CERTIFICATION

J. Jorge Dias Jr., J. Adson O. G. da Cunha, Alexandre Álvaro
Roberto S. M. de Barros and Sílvio Meira
Informatics Center, Federal University of Pernambuco, Recife, Brazil

Keywords: Web Services Certification, Quality Model.

Abstract: Internet has made possible the development of software as services, consumed on demand and developed by third parties. In this sense, a quality model is necessary to enable evaluation and, consequently, reuse of the services by consumers. In this way, this paper proposes a quality model based on the ISO 9126 standard, defining a set of attributes and metrics for an effective evaluation of Web services. A XML-based representation model was created to support this quality model, and a security schema was proposed to guarantee integrity and authenticity of the model.

1 INTRODUCTION

In the last decade, Component-Based Development (CBD) has drawn great attention due to the development of plug-and-play, reusable software (Bertoa et al., 2002). Traditionally, software components have been adopted, developed and delivered as a product. However, the Internet has created a recent surge of interest in developing software as services, delivered and consumed on demand. The benefit of this approach is seen in the looser coupling between business problems and the associated software solutions (Elfatatry, 2004).

Several works point out that Service-Oriented Architectures (SOA) has influenced the development of new systems. The Gartner Institute and other authors claim that SOA proposes to be a CBD evolution (McCoy et al., 2003)(Szyperski, 1998).

Hence, Web Services, a SOA-based technology that has been used in industry in the last years, had emerged in order to construct applications that reuse services available in the network (Stal, 2002). Web Services allows that services developed in different languages and platforms communicate through open and universal standards. In this way, distributed applications development through the reused services of the net becomes faster, increasing the quality and decreasing product costs and delivery time (Elfatatry, 2004).

Just as CBD model aims the use of artifacts developed by third parties (Szyperski, 1998), the service-oriented development aims the use of services developed by third parties. In this sense, a quality model is necessary in system development through the services integration, in order to make possible the evaluation and, consequently, the reuse of services by consumers (developers and designers).

There are some difficulties that must be considered in the elaboration of such model, such as (i) what characteristics and quality attributes must be considered, (ii) how can we evaluate them and (iii) who must be responsible for such evaluation (Alvaro et al., 2005). Moreover, there is a lack of information on the quality attributes provided by the vendors that publish Web services, such as StrikeIron (StrikeIron, 2006), Xignite (Xignite, 2006) and XWebServices (XWebServices, 2006). The international standards of software product quality specification, such as ISO 9126 (ISO/IEC 9126, 2001) and ISO 14598 (ISO/IEC 14598, 1998) are very general, and, thus, it is difficult to apply them in specific domains, such as Web services.

We can define certification as a set of regulated and standardized procedures that result in the certification or declaration expedition of specific conformity for products or services of one determined area (ISO, 2006). The benefits to certificate products are numerous. Some of them are: demonstration of quality promise, consumer trust,

Jorge Dias Jr. J., Adson O. G. da Cunha J., Álvaro A., S. M. de Barros R. and Meira S. (2007).
A XML-BASED QUALITY MODEL FOR WEB SERVICES CERTIFICATION.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - DISI*, pages 288-294
DOI: 10.5220/0002389402880294

requirements conformance and to aggregate value to the product. In this context, trust certifier entities must analyze the Web services and emit a certification for them with the evaluated data.

This paper proposes a quality model based on the ISO 9126 standard (ISO/IEC 9126, 2001), defining a set of attributes and respective metrics for an effective evaluation of Web services. Moreover, a XML-based representation was elaborated to support the proposed quality model. A security schema is also proposed in order to avoid that malicious entities corrupt the certification data, or even that, the same certification document is attached to another services.

The remainder of this paper is organized as follows: section 2 presents related work referring to the existing quality models. Section 3 criticizes the ISO 9126 model, showing the suitable characteristics to the Web services context. Section 4 proposes the use of XML for attributes documentation, as well as a security schema to guarantee the authenticity and integrity of it. Then, section 5 presents some concluding remarks.

2 RELATED WORK

Specifications have been proposed to provide definitions and classifications of the quality characteristics of software products, like the ISO 9126 International Standard.

Because it has a high level of abstraction, it does not include in the specification a detailed set of attributes and metrics, so many studies have been realized towards a quality model for components, in the last few years (Bertoa et al., 2002)(Bertoa et al., 2003)(Alvaro et al., 2005), focusing the ISO 9126 model in the characteristics of such software. However, due to the particularities of Web services, it is necessary a new quality approach, that takes into account the relevant characteristics for such kind of software.

Many proposals have emerged towards Services Level Agreement (SLAs) to improve the Web services reuse by consumers (Tosic, 2004). Generally, these agreements have dynamic service information at QoS (Quality of Service) level, like performance, availability and reliability, facilitating their selection at runtime. On the other hand, as in the CBD approach, services are selected at the project life cycle. Therefore, other static quality characteristics, inherited from CBD, are important to facilitate the services selection, such as documentation, support, business model, etc.

Two works proposed to evaluate the quality of web services. One of them is the QMWS (Quality Model for Web Services) specification, from OASIS (OASIS, 2005), which provides a model to manage the Web services quality at development process and use of them. Nevertheless, there is only a conceptual level draft, being necessary the definition of the quality properties to be evaluated in accordance with the Web services characteristics. Other work was proposed by Ran (Ran, 2003) and it describes a discovery model of Web services based on functional and non-functional requirements of them. In this model, the author considers a set of non-functional attributes to evaluate the quality of Web Services. But, unfortunately, it does not describe the metrics used for such evaluation, nor its real adequacy to the industry.

In this sense, this work complements such proposals, elaborating a quality model with its characteristics, sub-characteristics, attributes and metrics for Web services, evaluating not only its dynamic characteristics, but also its static characteristics. A XML representation model is also elaborated for supporting the proposed model.

3 QUALITY MODEL FOR WEB SERVICES

In accordance with (ISO/IEC 9126, 2001), a quality characteristic is a set of software product properties through which its quality can be described and evaluated, being such characteristics refined in sub-characteristics.

An attribute is a property which a metric can be associated, being metric a procedure to check a product, giving a data that characterizes it.

A quality model, in turn, is a set of characteristics and sub-characteristics, as well as the relation among them, in order to provide the basis for the specification of the quality requirements, and its evaluation.

The idea proposed in this paper is to refine the ISO 9126 quality model in order to accommodate particular characteristics of the Web services, defining attributes to be used by the Web services vendors, making possible the services evaluation and selection by the consumers.

Although the proposed model is based on the ISO 9126 standard, some changes have been made in order to develop a consistent model that can be used to evaluate Web services. The main changes made on ISO 9126 model are explained below.

- Functionality: It describes if the functions and specific properties of the web service satisfy the consumer necessities. The Interoperability sub-characteristic was removed because was considered unnecessary to the model, since the Web services are accessed through known Internet standards. The Compatibility sub-characteristic was added in order to indicate if one given version of the Web service is compatible with its previous versions, as considered by (Bertoa et al., 2002). The meaning of the others sub-characteristics remains the same.
- Reliability: It indicates if the Web service is able to keep on performance level, throughout the time, in the established conditions. The sub-characteristics remain the same.
- Usability: It refers to the effort needed for using the Web service by the consumers. The meaning of the sub-characteristics remains the same.
- Efficiency: It describes if the involved resources and times are compatible with the performance level required by the Web service. Since the Web service is located in a place of the Internet, there is no quantitative concern of the system in relation to use resources by the Web services. For this reason, the Resource Utilization sub-characteristic is dismissible. On the other hand, the Time Behavior sub-characteristic has its name modified to Performance because it refers not only to the reply time of the service processing, but also to how fast the response is sent to the requester. The Stability sub-characteristic, in accordance with the ISO 9126, pertaining to the Manutenability sub-characteristic, by having its context modified, was relocated for the Efficiency characteristic, meaning how available is the service in a continuous and consistent form.
- Maintainability: This characteristic is not included in the proposed model. A natural characteristic of Web services is the fact that they are considered black-box components, that is, consumers are not worried about how the service is implemented, but what the service makes (McGovern, 2003). In this way, the developer can not make internal modifications, adaptations or reconfigurations in the Web service.
- Portability: Since the Web service does not have the same characteristic that components to be deployable, that is, the service is available in some place of the Internet to be accessed through the net, it is not necessary that services be installed in different platforms. Hence, the Portability characteristic is unnecessary to the model.
- Market: Express the market Web services characteristics, finishing the proposed quality model. Although this characteristic is not important for the evaluation of service quality, it was incorporated to the model for making possible the analysis of some factors related to the service organization provider in order to improve service credibility for the consumers (Alvaro et al., 2005b)(Carvalho et al., 2006).

Table 1 shows a summary of Web services quality model based on ISO 9126, with its modifications being divided into two classes: characteristics that are observed at runtime and characteristics that are observed during the service life cycle (development phase).

Table 1: Quality Model for Web Services.

Characteristics	Sub-characteristics (*Runtime*)	Sub-Characteristics (Life Cycle)
Functionality	Accuracy Security	Suitability Compliace Compatibility
Reliability	Recoverability Fault Tolerance	Maturity
Usability		Understandability Learnability Operability
Efficiency	Performance Stability	

3.1 Attributes and Respective Metrics

Since the quality model was defined, it is necessary to specify attributes in order to make possible the characteristics evaluation..

Three types of metrics will be used to evaluate the attributes: *Presence*, that defines if an attribute is present or not in the Web service, consisting of a boolean value, which indicates if an attribute is present, and a string, which describes how the attribute is described by the Web service; *Value* that indicates the accurate value of the Web service information, being described by an integer value and a string to indicate the unit, and; *Ratio* that describes percentages, being measured by an integer value in the interval of 0 to 100.

3.1.1 Attributes Measured at Runtime

Table 2 shows the quality attributes for Web services, which are evaluated at runtime, grouped by the sub-characteristics, and including the kind of metrics used.

A description of each attribute is presented below:

- Correctness: It evaluates the percentage of results gotten with precision. It is calculated dividing the number of correct results by the total number of results gotten in one determined series of calls to the service.
- Data Encryption: It indicates if the Web service makes use of encryption in order to protect the data that it manipulates.
- Controllability: It indicates how the access control to the Web service is made.
- Auditability: It indicates if the Web service implements some auditory mechanism.
- Error Handling: It indicates if the Web service manipulates error situations.
- Persistent: It indicates if the Web service can store its state in a persistent way for later consultation.
- Mechanism available: It indicates the tolerance mechanism implemented by the Web service.
- Response time: It indicates the time taken to receive a reply, from a request, including the processing time and net traffic.
- Throughput: It indicates the amount of output that can be successfully produced during a period of time.
- Processing Capacity: It indicates the amount of input that can be successfully produced during a period of time.
- Availability: It indicates the period of time which a service exists or is available for use.
- Successability: It is defined by the number of received messages divided by the number of required messages.
- Accessibility: It indicates if the Web service is accessible, through the verification if it can return an act for each required message.

3.1.2 Attributes Measured at Life Cycle

Table 3 presents quality attributes, as well as the kind of metrics used, referring to the sub-characteristics evaluated during the Web services life cycle.

Table 2: Attributes of the sub-characteristics evaluated at runtime.

Sub-Characteristics (*runtime*)	Attributes	Metric
Accuracy	Correctness	Ratio
Security	Data Encryption	Presence
	Controllability	Presence
	Auditability	Presence
Recoverability	Error Handling	Presence
	Persistent	Presence
Fault Tolerance	Mechanism available	Presence
Performance	Response time	Value
	Throughput	Value
	Processing Capacity	Value
Stability	Availability	Ratio
	Successability	Value
	Accessibility	Value

Table 3: Attributes of the sub-characteristics evaluated at life cycle.

Sub-Characteristics (*runtime*)	Attributes	Metric
Suitability	Coverage	Ratio
	Pre-condition and Post-condition	Presence
Compliance	Service agreement	Presence
	Certification	Presence
Compatibility	Compatibility	Presence
Maturity	Volatility	Value
	Evolution	Value
Understandability	Documentation available	Presence
Learnability	Time and effort to (use, configure, admin and expertise) the Web service	Value
Operability	Provided Interfaces	Value
	Dependencies	Value

- Coverage: It indicates how much of the required functionalities are implemented by the Web service.
- Pre-conditions and Post-conditions: It determines the input and output conditions of the service.
- Service agreement: It indicates if the Web service has a WS-* agreement.
- Certification: It indicates if the Web service is certified by some organization.
- Compatibility: It indicates if the service is compatible with its previous versions.
- Volatility: It indicates the time between the availability of a version and another one of the Web service.
- Evolution: It indicates the versions number of the Web service launched in the market.

- Documentation available: It indicates the availability of documentation, demos, user guide, APIs and tutorials.
- Time and effort to (use, configure, admin and expertise) the Web service: It measures the necessary time and effort for accomplishment of specific tasks, like use, configuration and administration of the Web service.
- Provided Interfaces: It indicates the amount of provided interfaces for the Web service.
- Dependencies: It indicates the amount of services of which the service is dependent.

3.1.3 Market Attributes

The market attributes complement the model with information about the market structure that involves the service, such as:

- Reputation: Description of the provider to perform similar projects based on past experiences;
- CMM Level: It indicates the CMM level of the provider.
- Support: It describes the support mechanism offered by the provider.
- Business model: It indicates how the service is acquired, as well as its cost.
- Target Market: It indicates the target market for which the service was implemented.

There is no specific metric to measure such attributes and, thus, they are described using a string with the information.

4 XML REPRESENTATION MODEL

Web Services architecture uses several standards used in different levels to obtain interoperability between three basic operations: publish, find and bind. Almost all these levels depend on XML language (Gottschalk, 2002). The main goal of XML use in Web Services architecture is to solve the service dynamic discovery problem (Tosic, 2004).

After the quality model is defined, a certifier entity must test and validate the service, certifying it. For this reason, a XML model was developed to represent the service certification with its characteristics, sub-characteristics and attributes mentioned in section 3.

The proposed XML model is basically divided in two parts (see figure 1), where the document root is *service-certification*. The first part refers to certification data, that is, the information collected in service tests and validation. The second part mentions the certifier identify, as well as its digital signature.

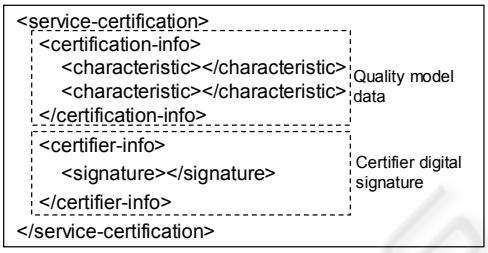

Figure 1: XML representation model correspond to service certification.

In its first part, the model contains certification information, such as the certification date (element *date*), and elements that represent the characteristics, sub-characteristics and collected metrics attributes. Attributes are represented by the same idea proposed in (Bertoa et al., 2002), where the ODP reference model is used (ISO 10746, 1997). In this approach, each attribute is written using a name-value pair. Each name has a type that determines the possible values that it can assume. The benefits in this approach include the fact that it is in accordance with an international standard, and also that it is easy to document using XML templates.

```
<certification-info date="2006-05-06">
    <characteristic name="Efficiency">
      <sub-characteristic name="Stability">
        <property name="Availability">
          <type>xsd:ratio</type>
          <value>95</value>
        </property>
      </sub-characteristic>
    </characteristic>
</certification-info>
```

Figure 2: Part of the model that contains certification information.

A security schema is necessary to guarantee that the service certification will not be corrupted by malicious entities. In this way, the proposed solution is based on a digital signature calculated on the XML certification model and the WSDL description URL, guaranteeing its integrity. Public key digital signatures are typically, in order to provide integrity (Stallings, 1995). For that, the XML-Signature (W3C, 2006) standard was used. The main characteristic of XML-Signature is the ability to sign only specific portions of the XML tree, instead of signing the complete original document.

This XML document serves as a service certification stamp, that is, the service provider will be able to attach this document to any other document that is related to the service, for example, a WSDL description or even to extend the tModel of the UDDI, such as considered in (Ran, 2003). In this way, the consumer locates the desired service, being able to analyze the certificate information and to verify if that certificate is signed by a trusted certifier or not. This verification is made by the consumer itself, since that XML document contains the certifier public key. If any person tries to modify any element of the certification document, then the signature becomes invalid.

5 CONCLUDING REMARKS

This work presented a quality model based on the ISO 9126 international standard in order to certify Web services. However, in contrast of existent service contracts and SLAs, the goal of the proposed model is not limited to services dynamic selection, as it also aims to facilitate, in a simple way, a static choice of the most appropriate service to be reused by the consumer.

A XML representation model was developed to support the proposed quality model, such as a security schema using the XML-Signature standard in order to guarantee the integrity and authenticity of the certificate generated by the certifier entity.

In this way, this model could be adopted by Web services vendors that publish Web services, guaranteeing information standardization in these sites, making possible the services selection by consumers in the system development adopting the SOA approach.

A preliminary case study was elaborated with Web sites that sell commercial and financial Web services were visited, namely StrikeIron (StrikeIron, 2006), Xignite (Xignite, 2006), and XWebServices (XWebServices, 2006), that have their products published in Xmethods, one of the most popular sites that lists publicly available Web services. From this survey, it is possible to have the first impression about the sort of information available at these sites, how it is advertised, and how difficult it is to extract the quality information we are looking for. It was not carried out a rigorous statistical analysis of the Web services population, since the objective is only to have coverage that is just enough to make a first impression of the available information possible. However, due to the limited pages, we could not expose the results of this preliminary case study.

As future work, a formal case study about the services information available in the vendor sites will be done, in order to verify how much information of the model is actually available in the Web services providers.

REFERENCES

Alvaro, A., et al, 2005. *Quality Attributes for a Component Quality Model*, In The 10Th International Workshop on Component-Oriented Programming (WCOP). Glasgow, Scotland.

Bertoa, M., Troya, J., Vallecillo, A., 2003. *A Survey on the Quality Information Provided by Software Component Vendors*. In Proceedings of QAOOSE 2003 at 7th ECOOP.

Bertoa, M., Vallecillo, A, 2002. *Quality Attributes for COTS Components*. Proc. of the 6th International ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE), Spain.

Carvalho, J. P., Franch, X., 2006. *Extending the ISO/IEC 9126-1 Quality Model with Non-Technical Factors for COTS Components Selection*. ICSE. China.

Elfatatry, A., Layzell, P., 2004. *Negotiating in Service-Oriented Environments*. Communications of the ACM. Vol 47, Nº 8.

Gottschalk, K., Graham, S., Kreger, H., Snell, J., 2002. *Introduction to Web Services Architecture*. IBM System Journal 41, nº 2.

ISO - *International Organization for Standardization*, 2006. http://www.iso.org.

ISO 14598, 1998. *Software Engineering – Product evaluation*. Internacional Standard ISO/IEC 14598, Internacional Standard Organization.

ISO 9126, 2001. *Information Technology – Product Quality – Part1: Quality Model*, International Standard ISO/IEC 9126, International Standard Organization.

ISO/IEC. RM-ODP, 1997. *Reference Model for Open Distributed Processing*. Rec. ISO/IEC 10746-1 to 10746-4, ITU-T X.901 to X.904, ISO/ITU-T.

McCoy, D. W., Natis, Y. V., 2006. *Service-Oriented Architecture: Mainstream Straight Ahead*. http://www4.gartner.com/DisplayDocument?doc_cd=114361

McGovern, J., Tyagi, S., Stevens, M., Matthew, S., 2003. *Java Web Services Architecture*. Morgan Kaufmann Publishers.

OASIS, 2006. *Web Services Quality Model (WSQM)*. Available at http://www.oasis-open.org/committees/download.php/15910/WSQM-ver-2.0.doc.

Ran, S., 2003. *A Model for Web Services Discovery With QoS*. ACM SIGecom Exchanges, Vol. 4.

Stal, M., 2002. *Web Services: Beyond Component-Based Computing*. Communications of the ACM, Vol. 45, No 10.

Stallings, W., 1995. *Network and Internetwork Security: Principles and Practice*. Prentice Hall.
StrikeIron. (n.d.). Retrieved November 4, 2006, from http://www.strikeiron.com.
Szyperski, C., 1998. *Component Software: Beyond Object-Oriented Programming*. Addison-Wesley.
Tosic, V., Pagurek, B., 2004. *On Comprehensive Contractual Descriptions of Web Services*. Technical Report, Department of Computer Science, The University of Western Ontario.
W3C World Wide Web Consortium, 2006. *XML Signature WG*. http://www.w3.org/Signature/.
Xignite. (n.d.). Retrieved November 4, 2006, from http://www.xignite.com.
XMethods. (n.d.) Retrieved November 03, 2006, from http://www.xmethods.com.
XWebServices. (n.d.). Retrieved November 4, 2006, from http://www.xwebservices.com.