



Web Services Quality Factors Version 1.0

Committee Draft 02 / Public Review Draft 01

01 July 2010

Specification URIs:

This Version:

<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors-v1.0-cd02.doc>
<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors-v1.0-cd02.html>
<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors-v1.0-cd02.pdf>

Previous Version:

N/A

Latest Version:

<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors.doc>
<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors.html>
<http://docs.oasis-open.org/wsqm/wsqf/v1.0/WS-Quality-Factors.pdf>

Technical Committee:

OASIS Web Services Quality Model TC

Chair(s):

Eunju Kim, National Information Society Agency

Editor(s):

Eunju Kim, National Information Society Agency
Yongkon Lee, Individual
Yeongho Kim, Daewoo Information Systems
Hyungkeun Park, IBM
Jongwoo Kim, Individual
Byoungsun Moon, Individual
Junghee Yun, National Information Society Agency
Guil Kang, National Information Society Agency

Related work:

N/A

Declared XML Namespace(s):

N/A

Abstract:

The purpose of this document is to provide the standard for quality factors of web services in their development, usage and management. Web services usually have distinguished characteristics. They are service-oriented, network-based, variously bind-able, loosely-coupled, platform independent, and standard-protocol based. As a result, a web service system requires its own quality factors unlike installation-based software. For instance, as the quality of web services can be altered in real-time according to the change of the service provider, considering real-time property of web services is very meaningful in describing the web services quality. This document presents the quality factors of web services with definition, classification, and sub-factors case by case. For each quality factor, related specifications annexed with a brief explanation. This

specification can be generally extended to the definition of quality of SOA and provide the foundation for quality in the SOA system.

Status:

This document was last revised or approved by the WSQM TC on the above date. The level of approval is also listed above. Check the “Latest Version” or “Latest Approved Version” location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee’s email list. Others should send comments to the Technical Committee by using the “Send A Comment” button on the Technical Committee’s web page at <http://www.oasis-open.org/committees/wsqm/>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (<http://www.oasis-open.org/committees/wsqm/ipr.php>).

The non-normative errata page for this specification is located at <http://www.oasis-open.org/committees/wsqm/>.

Notices

Copyright © OASIS® 2010. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is trademarks of [OASIS](http://www.oasis-open.org), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

Table of Contents

1	Introduction.....	6
1.1	Characteristics of WS Quality	6
1.2	WS Quality Factors.....	7
1.3	Audience.....	8
1.4	Terminology.....	8
1.5	Normative References	8
1.6	Non-Normative References	8
2	Business Value Quality.....	9
2.1	Definition.....	9
2.2	Sub Quality Factors	9
2.2.1	Price	9
2.2.2	Penalty and Incentive	9
2.2.3	Business Performance	10
2.2.4	Service Recognition.....	10
2.2.5	Service Reputation	10
2.2.6	Service Provider Reputation.....	10
3	Service Level Measurement Quality.....	11
3.1	Definition.....	11
3.2	Sub Quality Factors	11
3.2.1	Response Time	11
3.2.2	Maximum Throughput.....	12
3.2.3	Availability	12
3.2.4	Accessibility	12
3.2.5	Successability.....	13
4	Interoperability Quality.....	14
4.1	Definition.....	14
4.2	Sub Quality Factors	15
4.2.1	Standard Adoptability	15
4.2.2	Standard Conformability	15
4.2.3	Relative Proofness	15
4.3	Relationships to other Standards.....	16
5	Business Processing Quality	17
5.1	Definition.....	17
5.2	Sub Quality Factors	17
5.2.1	Messaging Reliability.....	17
5.2.2	Transaction Integrity	18
5.2.3	Collaborability	19
5.3	Relationship to other Standards.....	19
6	Manageability Quality	21
6.1	Definition.....	21
6.2	Sub Quality Factors	21
6.2.1	Informability	21
6.2.2	Observability.....	21

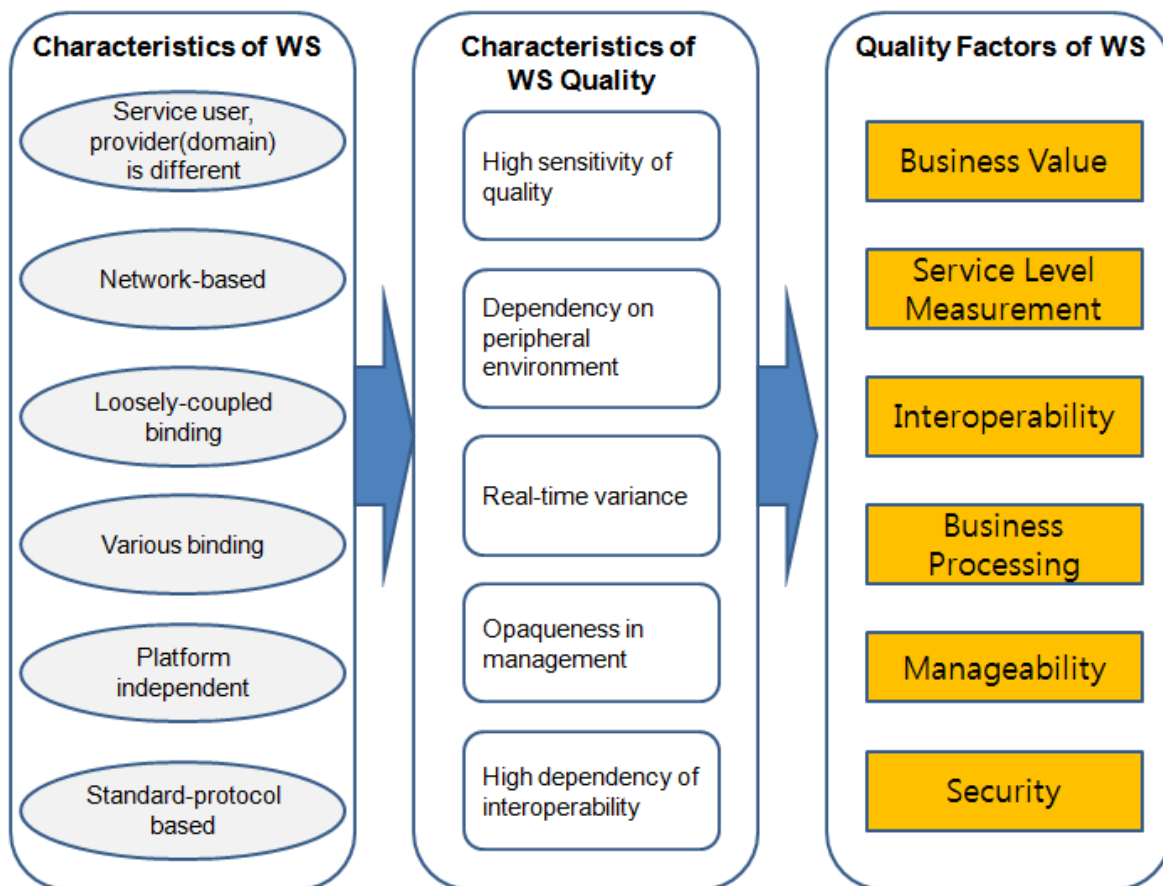
6.2.3 Controllability	22
6.3 Relationship to other Standards.....	22
7 Security Quality	24
7.1 Definition.....	24
7.2 Sub Quality Factor	24
7.2.1 Encryption	24
7.2.2 Authentication.....	25
7.2.3 Authorization	25
7.2.4 Integrity.....	25
7.2.5 Non-Repudiation.....	26
7.2.6 Availability	26
7.2.7 Audit	26
7.2.8 Privacy.....	27
7.3 Relationship to other Standards.....	27
8 Conformance.....	28

1 Introduction

The importance of web services has been raised as an enabler of Service Oriented Architecture (SOA). As a result, most software communities who are related in the planning, development and management of SOA have significantly interested in Web Services quality. This document specifies web services quality factors conceptually along with definition and explanation of sub-factors. This chapter presents the basic characteristics of web services and quality factors induced from them.

1.1 Characteristics of WS Quality

Web services have distinguished characteristics different from installation-based software because of their service-oriented nature. The provider and consumer of services could belong to different ownership domains so that there are many cases that a service cannot meet the consumer's service requirements in respect of service quality and content. Web services are usually invoked through networks, so the network performance critically affects the overall web service quality. As a web service client binds to a web service server with loosely-coupled manner and various binding mechanisms, both client and server could be bound easily and flexibly. On the contrary, the client and the server cannot guarantee for proper operating performance. They may be operated platform-independently, so it requires more efforts for guaranteeing interoperability between them. Even though web services are based on standard protocols of communication, misconception of the protocols can produce critical results in non-interoperable services.



<Figure 1-1> Extracting Quality Factors of WS

21 Due to the characteristics described above, web services show distinct quality characteristics from those
22 of general software. Firstly, the usage of web services is highly sensitive to their quality, especially in
23 regard to performance and business. A web service consumer is willing to change a service while using it
24 if it cannot satisfy his requirements on performance or business. Most of web service consumers have an
25 interest in the quality of web services and thus would correspond to the quality problem immediately.
26 Secondly, as web services are operated in the close relation with the other systems, the web service
27 quality depends on the peripheral technical environment: network, security system, and software resource
28 system, and business effectiveness. For example, the quality of transport media influences deeply on the
29 web service quality. Consequently, even though a web service shows very rapid response time on the
30 server side, we cannot expect rapid response time if the bandwidth of transport network is narrow. In the
31 same way, although a web service has been implemented efficiently, it is difficult to expect good
32 performance of the web service when a service provider has low processing capability. Thirdly, a web
33 service client and a server are bound loosely and variously. The web service client can change the web
34 service server dynamically, so the client can experience considerable variation of web service quality.
35 The client can change web services in real-time when the quality is not satisfied. Fourthly, a web service
36 consumer is usually not able or restricted to manage and control a web service, because in many cases a
37 consumer's domain is different from a service provider's. Accordingly, the web service consumer requires
38 guaranteeing the higher level of web service quality. Finally, more effort to assure interoperability of web
39 services is required, because a web service client and a server system could be deployed on
40 heterogeneous platforms and web service developers could misunderstand related standards.

41 To summarize above, the characteristics of web services lead to the distinguished characteristics of web
42 service qualities unlike those of installation-based software. Therefore, it is required to induce quality
43 items in alignment with consideration of these characteristics of web service quality during overall web
44 service lifecycle.

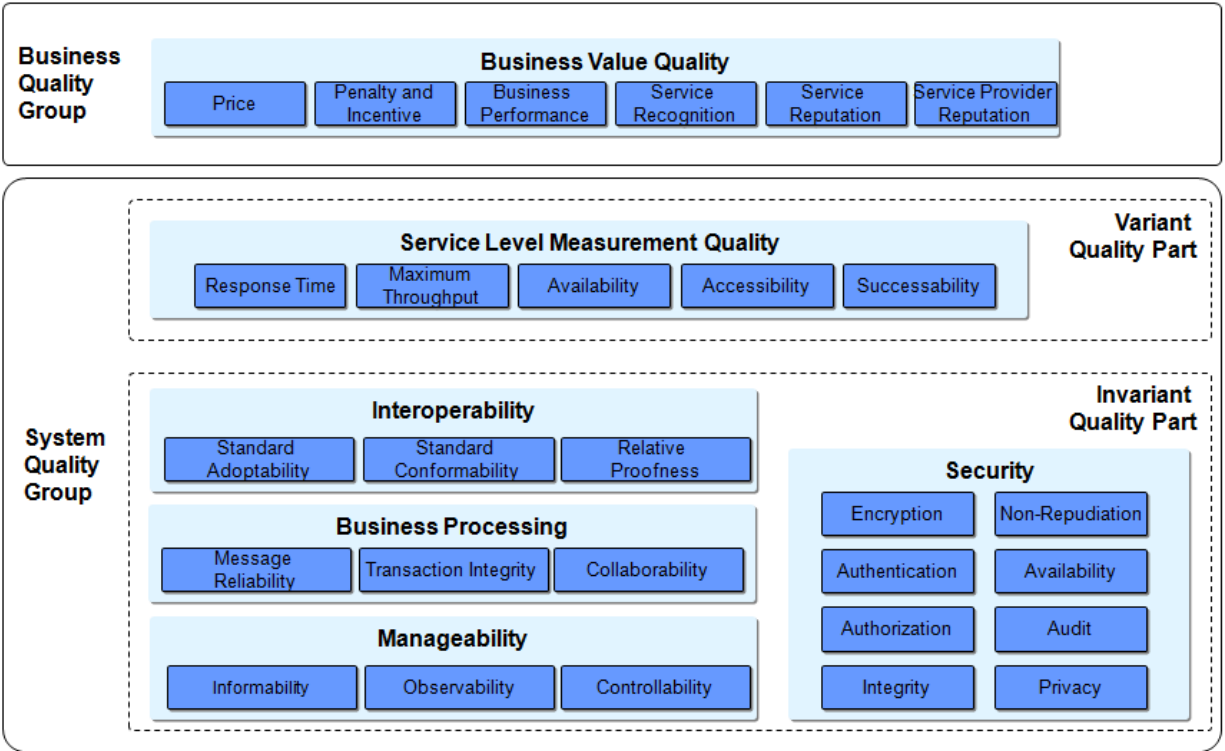
45 **1.2 WS Quality Factors**

46 A web service quality factor refers to a group of items which represent web service's functional and non-
47 functional properties (or values) to share the concept of web services quality among web service
48 stakeholders. Based on the characteristics of web service quality described previously, the web service
49 quality factors are composed of business value quality, service level measurement quality, interoperability
50 quality, business processing quality, manageability quality and security quality.

51 Based on whether quality factors are related with business perspective or system perspective, they can
52 be categorized into two groups: the business quality group and the system quality group (Refer to <Figure
53 1-2>). Business quality group includes only the business value quality factor. System quality group is
54 comprised of the variant quality part and the invariant quality part. The variant quality part includes quality
55 factors whose values can be dynamically varied in run-time while a service is being used. On the while,
56 the invariant quality part refers to quality factors whose values are determined as soon as the service
57 development is completed. The invariant quality part includes interoperability quality, business processing
58 quality, manageability quality and security quality.

59 Business value quality refers to a business perspective to help to make the right selection of a service by
60 evaluating the business value of web services. For evaluating business value, it includes the sub-factors:
61 price, penalty and incentive, business performance, service recognition, service reputation and service
62 provider reputation. Service level measurement quality measures the performance of web services in
63 numeric value: response time, maximum throughput, availability, accessibility and successability.

64 Interoperability quality is a quality factor to evaluate whether a web service system conforms to standard
65 adoptability, standard conformability and relative proofness. Web services may be used in mission-critical
66 work between business partners, and in that case reliability and stability of web services are very
67 important quality items. The business processing quality factor evaluates these items, including
68 messaging reliability, transaction integrity and collaborability. Manageability quality is about to whether
69 web services are manageable or managed items, including informability, observability and controllability,
70 web services are also vulnerable to security attack and fraud of their frequent exposure to open networks.
71 Security quality guarantees the safety of web services for use. That is, it is a collection of quality items to
72 evaluate the functionality and the metric performance of a security system. It includes the sub-factors:
73 encryption, authentication, authorization, integrity, non-repudiation, availability, audit and privacy.



74
75

<Figure 1-2> Structure of Web Services quality factor

76 **1.3 Audience**

77 The intended audiences of this document include non-exhaustively:

- 78 • Quality associates of web services and SOA: quality managers, quality assurers, quality authenticators, quality information providers, etc.
- 79
- 80 • Architects and developers designing, identifying or developing a system based on web services or SOA concept.
- 81
- 82 • Standard architects and analysts developing specifications of web services or SOA.
- 83 • Decision makers seeking a "consistent and common" understanding of web services or SOA.

84 **1.4 Terminology**

85 The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

88 **1.5 Normative References**

- 89 [RFC2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*,
90 <http://www.ietf.org/rfc/rfc2119.txt>, IETF RFC 2119, March 1997.

91 **1.6 Non-Normative References**

92 None

93 2 Business Value Quality

94 2.1 Definition

95 When a service party decides to use a web service for business, it should consider surely the value of the
96 service on the business. In some cases, the web service may give positive value such as profit,
97 convenience, collaboration to the service party. But in the other cases, it may impose more burdens on
98 the party than the value it delivers. For example, it may require an extra cost for keeping the service in
99 stable condition or cause economic loss due to service provider's failure to meet promised quality. As a
100 result, web service consumers tend to be very sensitive generally to the value of web services on a
101 business.

102 The business value of web services means the economic worth delivered by applying web services on a
103 business. The business value depends on the price of a service, a penalty/compensation policy, service
104 recognition, service reputation and service provider reputation. In addition to those sub-factors, business
105 benefit, profit, and ROI (return on investment) caused by web services could be added in the business
106 value quality. But, it's very difficult to evaluate these values by the effect caused by web services alone
107 because the values depend heavily on each individual business context. Thus, we exclude the business
108 benefit, profit, and ROI from the business value quality.

109 The price and the penalty/compensation sub-factors represent the monetary value, which could be
110 determined by a service provider or the contract between a service provider and a consumer. The service
111 recognition, the service reputation and the service provider reputation are related with the trust of web
112 services, so they could be evaluated as a part of business value.

113 Business value quality provides a business perspective to help to make a right selection of service by
114 evaluating the business value of web services. Consumers refer to the value of this factor to reach a
115 decision to select the most appropriate web service for a given business.

116 2.2 Sub Quality Factors

117 2.2.1 Price

118 The price sub-factor is a monetary value of service that a consumer pays for services to a provider while
119 or after using web services. The price of a web service can be determined by a service provider. A
120 service consumer considers the price of a web service in respect of the functions, contents, and the
121 quality of the web service in order to make the decision whether he uses the web service. For general
122 software, users pay overall price for a software package. On the contrary, a consumer has to pay the fee
123 of a web service continuously based on the amount of time usage or use data, so he has continuously
124 interest in the service price. Therefore, the price of a web service affects in the usage of the web service.
125 A service provider should decide the appropriate price by considering the service quality and value. In
126 relation to the price factor, a billing method is also important quality factor for measuring business value of
127 a web service. For example, a reasonable and systematic billing system can improve the trust of a web
128 service. A convenient billing system, a discount policy, and mileage points enhance consumer's loyalty.

129 2.2.2 Penalty and Incentive

130 Penalty or compensation is the financial compensation for business losses due to nonfulfillment of a
131 contract or failure to meet promised quality. Penalty can be charged to a service provider or a service
132 consumer based on a contract. When a service provider fails to keep service quality levels specified in
133 the contract, the service provider needs to compensate for the loss of a service consumer. The
134 compensation rules need to be specified in the contract. Penalty can be calculated based on service
135 downtime, maximum or average response times, or security requirements of a service, and so on. The
136 performance monitoring of the web service is necessary to determine whether compensation is required
137 or not, and how much compensation is required. Penalty can be charged to a service consumer when the
138 consumer breaches the contract unilaterally, which brings financial loss to a service provider.

139 On the contrary, incentives as positive rewards can be specified in a contract. For example, an incentive
140 can be paid when a service provider has provided higher quality than the quality level specified in the
141 contract. In addition, an incentive can be paid to a service consumer when the service consumer uses a
142 service more than a certain usage level during a given time period.

143 **2.2.3 Business Performance**

144 In the case that a service provider provides commodities and services in the real world as well as
145 information by web services, the performance of business activity provided for them affects the business
146 value of the web services directly. For example, consider a delivery service with which web services are
147 provided for an order and a payment process. In this case, all the processes including the order,
148 confirmation, delivery, notification and payment which are all connected within the information flow and
149 business activity flow. Overall service quality is related with the capability of the business body as well as
150 the quality of web services.

151 Business performance is defined as the capability of a business party performing business activities for
152 services. The business performance can be measured by the time it takes to complete a business service
153 or the throughput. The time to complete is composed of the duration for performing business activities
154 and a latency for ready or a condition. The throughput is the amount of outcomes for a service per a unit
155 time

156 **2.2.4 Service Recognition**

157 Service recognition quality is defined by how many potential consumers perceive the existence of a
158 service. That is, it is related to the popularity of the service. A highly recognized service means that it has
159 more potential for many people to use the web service. Service recognition can be measured by various
160 methods. For example, it can be estimated by the number of clicks on a service description in a service
161 registry or the number of page views on a service web page.

162 Service recognition is not derived from the service consumer's experience of service usage. Also, the
163 service recognition level can be improved through promotion or advertising of a web service. However, it
164 does not guarantee the superiority of the other quality factors of the web service such as response time,
165 availability, and reliability.

166 **2.2.5 Service Reputation**

167 Service reputation is a social evaluation of service consumers toward a web service. It refers to
168 consumers' opinions on the quality of web services. Service reputation can be evaluated by performing a
169 survey or vote on service quality and consumer satisfaction. In addition, service reputation can be
170 estimated from the replies, comments or reviews of service consumers. Service reputation is very
171 influential for potential service consumers to select web services.

172 While service recognition reflects expectation of the service value before use, service reputation mainly
173 refers to the experienced service quality after use.

174 **2.2.6 Service Provider Reputation**

175 Service provider reputation is the opinion of the group of service parties toward a service provider on
176 certain criteria. Service provider reputation is an asset that gives the service provider a competitive
177 advantage because a service provider with a good reputation will be regarded as a reliable, credible,
178 trustworthy and responsible one for service consumers. It can be sustained through consistent quality
179 management activities on services as a whole. Service provider reputation can be influenced by
180 customer's previous experience on other services of the provider as well as advertisement or public
181 relations.

182 Service consumers pay attention to service provider reputation as well as a service itself. Service provider
183 reputation can be estimated by brand value, financial soundness, the quality of customer service,
184 technical support and sustainability of a service provider. .

185 3 Service Level Measurement Quality

186 3.1 Definition

187 As a service could be provided by third parties and invoked dynamically via network, service performance
188 might be varied by the network speed or the number of connected users at a given time. Service Level
189 Measurement quality is a set of quantitative attributes which describe the runtime service responsiveness
190 in a view of consumers. This quality factor represents how quickly and soundly web services can respond
191 which can be measured numerically on system.

192 Service Level Measurement Quality consists of five sub-quality factors; response time, maximum
193 throughput, availability, accessibility, and successability.

194 3.2 Sub Quality Factors

195 3.2.1 Response Time

196 Response time refers to duration from the time of sending a request to the time of receiving a response.
197 The response time can be varied by the point of measurement and affected by three types of latency:
198 client latency, network latency and server latency as depicted in <Figure 3-1>.

199 Client latency refers to the delay time caused by a client system in the whole processing time for a service
200 request. It is a sum of the time taken between 'a client application requests a service' event and 'the
201 request is sent by a client' event (t1~t2), and the time taken between the 'response arrives to the client'
202 event and 'the application system receives the response' event (t7~t8).

203 Network latency refers to the time taken on a network for transmitting request message and response
204 message. It is a sum of the time taken between 'a client sends a request' event and 'the web services
205 server receives the request' event (t2~t3), and the time taken between 'the server sends a response'
206 event and 'the client receives the response' event (t6~t7).

207 Server latency is a delay time caused by a server system in the whole processing time for a service
208 request. It is a sum of the time taken between 'the server sends the request' event and 'web services
209 receives the request' event (t3~t4), 'the time taken for processing the service' event (t4~t5), and the time
210 taken between 'the response is sent by the web services' event and 'the server receives the response'
211 event (t5~t6).

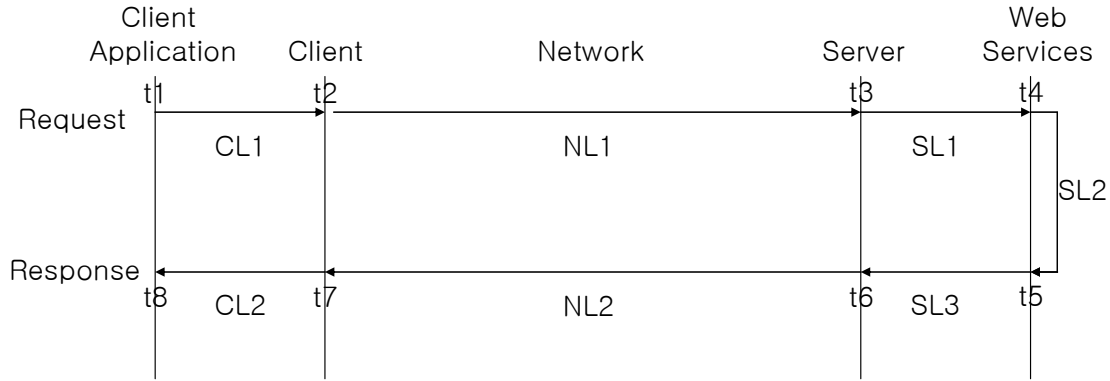
212 Three types of latency and response time can be calculated by the following formulas.

$$213 \quad \textit{ClientLatency} = \textit{CL1} + \textit{CL2}$$

$$214 \quad \textit{NetworkLatency} = \textit{NL1} + \textit{NL2}$$

$$215 \quad \textit{ServerLatency} = \textit{SL1} + \textit{SL2} + \textit{SL3}$$

$$216 \quad \textit{ResponseTime} = \textit{ClientLatency} + \textit{NetworkLatency} + \textit{ServerLatency}$$



217

* CL : Client Latency, NL : Network Latency, SL : Server Latency, ti : Measurement Time

218

<Figure 3-1> Response Time and Latency

219

3.2.2 Maximum Throughput

220

Maximum throughput refers to the maximum amount of services that the service provider can process in a given time period. It is the maximum number of responses which can be processed in a unit time. The following formula expresses the maximum throughput.

221

222

223

$$\text{MaximumThroughput} = \max\left(\frac{\text{NumberofRequestsProcessedbyServiceProviderInMeasuredTime}}{\text{MeasuredTime}}\right)$$

224

225

3.2.3 Availability

226

Availability is a measurement which represents the degree of which web services are available in operational status. This refers to a ratio of time in which the web services server is up and running. As the DownTime represents the time when a web services server is not available to use and UpTime represents the time when the server is available, Availability refers to ratio of UpTime to measured time. In order to calculate Availability, it is conveniently rather using DownTime than UpTime and it can be expressed as the following formula.

227

228

229

230

231

232

$$\text{Availability} = 1 - \frac{\text{DownTime}}{\text{MeasuredTime}}$$

233

3.2.4 Accessibility

234

Accessibility represents the probability of which web services platform is accessible while the system is available. This is a ratio of receiving Ack message from the platform when requesting services. That is, it is expressed as the ratio of the number of returned Ack message to the number of request messages in a given time. To increase accessibility, a system needs to be built in expansible architecture.

235

236

237

238

$$\text{Accessibility} = \frac{\text{NumberofAckMessage}}{\text{NumberofRequestedMessage}}$$

239 **3.2.5 Successability**

240 Successability is a probability of returning responses after web services are successfully processed. In
241 other words, it refers to a ratio of the number of response messages to the number of request messages
242 after successfully processing services in a given time. 'Being successful' means the case that a response
243 message defined in WSDL is returned. In this time, it is assumed that a request message is an error free
244 message.

245
$$\text{Successability} = \frac{\text{NumberofResponseMessage}}{\text{NumberofRequestedMessage}}$$

246

4 Interoperability Quality

247

4.1 Definition

248

249 For executing web services, there should be no semantic and technical problems in processing a
 250 message transmitted between a service provider and a service consumer. No semantic problem refers to
 251 the process when a receiver understands a message in the exact meaning as the sender intended. A
 252 prerequisite condition for the mutual understanding of semantics in a message, the name of service, the
 253 name and type of parameters, the type of return values are consistent between a service provider and a
 254 service consumer. This prerequisite condition may be satisfied if the service consumer implemented its
 255 system exactly according to the information of a service description (i.e. WSDL). But, this requires an
 256 agreement between all service parties for using service contents such as the name of items in e-
 257 documents and codes without any semantic problems. No technical problem refers to the conditions
 258 when all components for messaging including transport, security, reliability, encoding, and message
 259 structure coincide during implementation. Two systems of communication parties are said to be
 260 interoperable when they exchange and use information as if both could operate appropriately on the
 261 same platform. Implementing the messaging technology adopts related standards for assuring
 262 interoperability. If there is no standard available, one of service associates could adjust its technical
 263 implementation to the other's or both can agree to match their implementation specifications bilaterally.
 264 However, if a standard exists, the service associates can achieve interoperability by adjusting their
 265 implementation to the standard specification. Even though a service associate follows a standard,
 266 interoperability problems could arise in the case that an implementer misunderstands the standard or
 267 implements a module with a different intention. In some cases, the implementer may add new functions
 268 not described in the standard arbitrary. However this difference of a platform or network device could
 damage the interoperability between the two parties.

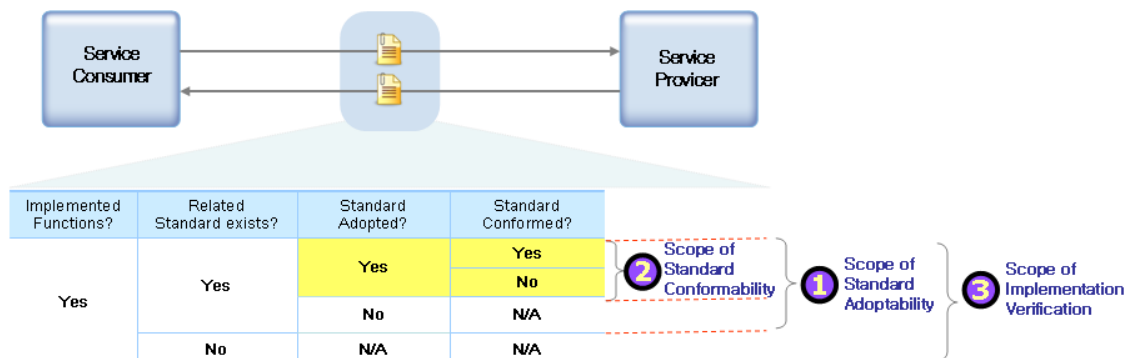
269

270 Considering the above cases, we can categorize the problems of interoperability into 3 groups: ①a
 271 system which has been implemented by not adopting a standard, ②a system which has been
 272 implemented according to a standard, with some functions implemented without regard to the standard, ③
 273 a system which has been implemented according to standards properly, but which has a problem of
 274 interoperability due to difference of platform of network. All of these issues must be considered in
 evaluating interoperability of a service system

275

276 Interoperability quality includes standard adoptability, standard conformability and relative proofness as
 277 shown in <Figure 4-1>. Standard Adoptability of web services evaluates how many functions of a web
 278 service are implemented by adopting related standards. The function of a web service means necessary
 279 requirements such as user authentication, data encryption, service delivery, transaction processing, etc.
 280 They could also include the original features of the business such as codes, document formats, business
 281 terms etc. Standard conformability of a web service evaluates whether the standards adopted to
 282 implement the functions of the web service conforms completely and correctively to the specification of
 283 the standards. Relative proofness evaluates whether a client and a service can communicate successfully
 on specific platforms.

284



285

<Figure 4-1> Interoperability of web services

286 4.2 Sub Quality Factors

287 4.2.1 Standard Adoptability

288 In order to guarantee interoperability of web services, functions of a web service should be implemented
289 by adopting related standards. Standard Adoptability is measured by the ratio of functions which are
290 implemented by adopting related standards. Standard adoption function f is defined on the set of
291 functions $X=\{x_1, \dots, x_n\}$ of a web service S and returns one of binary values, 0 and 1. f is formulated as
292 follows:

$$293 \quad f(x_i) = \begin{cases} 1 & \text{if function } x_i \text{ is implemented by adopting of related standards} \\ 0 & \text{otherwise} \end{cases}$$

294 Based on the standard adoption function f , *standard adoptability* of web service S is defined as follows:

$$295 \quad \text{StandardAdoptability}(S) = \frac{\sum_{i=1}^n f(x_i)}{n}$$

296 4.2.2 Standard Conformability

297 Assuming that a web service S has n functions, and only m functions of them are implemented by
298 adopting related standards. *Standard conformability function* g is defined by the set of standard adopted
299 functions $X_a=\{x_1, \dots, x_m\}$ of web service S and returns one of binary values, 0 and 1. g is defined as
300 follows:

$$301 \quad g(x_i) = \begin{cases} 1 & \text{if function } x_i \text{ conforms all adopted standards} \\ 0 & \text{otherwise} \end{cases}$$

302 Based on the standard conformability function, *standard conformability* of web service S is defined as
303 follows:

$$304 \quad \text{StandardConformability}(S) = \frac{\sum_{i=1}^m g(x_i)}{m}$$

305 4.2.3 Relative Proofness

306 Relative proofness indicates that web services are successful in exchanging and using the information
307 between two special system platforms. In real environments, services based on a technology platform
308 cannot be fully interoperable with other services on a different technology platform even if a standard
309 conformability test is passed. Whether the functions of web services are implemented by standards,
310 vendor specification or non-standards, relative proofness of service interoperability is tested and verified
311 in a real service platform that satisfies specific environments. VPI (Verified Platform Information)
312 represents the basic information of an opponent's platform and additional descriptions of the verification
313 when web services are tested and verified in real service environments of the opponent's platform.

314 *Relative Proofness* = $\{VPI_1, VPI_2, \dots, VPI_n\}$, where n is the number of platforms verified

315 **4.3 Relationships to other Standards**

- 316 • WS-I Basic Profile 1.1

317 This is a profile for interoperability of SOAP, WSDL, UDDI and is administered by WS-I.

318 URL: <http://www.ws-i.org/Profiles/BasicProfile-1.1.html>

- 319 • WS-I Basic Profile 1.2

320 This is a profile for interoperability of SOAP, WSDL, UDDI and is administered by WS-I.

321 URL: <http://ws-i.org/profiles/BasicProfile-2.0-WGD.html>

- 322 • WS-I Basic Security Profile Version 1.0

323 This is a profile for interoperability of web services security and is administered by WS-I.

324 URL: <http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0-2007-03-30.html>

- 325 • WS-I Basic Security Profile Version 1.1

326 This is a profile for interoperability of web services security and is administered by WS-I.

327 URL: <http://www.ws-i.org/Profiles/BasicSecurityProfile-1.1.html>

- 328 • WS-I Reliable Secure Profile 1.2

329 This is a profile for interoperability of reliable message and secured transmission and is
330 administered by WS-I. (WS-ReliableMessaging 1.1, WS-SecureConversation 1.3)

331 URL: <http://www.ws-i.org/Profiles/ReliableSecureProfile-1.0.html>

- 332 • WS-I Simple SOAP Binding Profile Version 1.2

333 This is a profile for interoperability of SOAP Binding and is administered by WS-I.

334 URL: <http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html>

335 5 Business Processing Quality

336 5.1 Definition

337 As the applying areas of web services are growing on a wide scale, the cases that use them in
338 communication between service units (i.e., enterprise, department, agency, program, division) are
339 increasing rapidly. Applying web services in business means that the service unit executing them has to
340 take responsibility of the execution result.

341 For applying web services in business, the intention of service providers and consumers has to be
342 reflected correctly in business results. A service unit can assure correctness and reliability in the business
343 context for business processing. In order to achieve this, a web service platform for business should
344 possess functions of reliable messaging, transaction processing, and collaborability. These functions
345 could be optionally used according to the requirement of a service unit. Accordingly, business processing
346 quality is defined as the capability of a web service platform for assuring correctness and reliability in
347 business processing.

348 5.2 Sub Quality Factors

349 5.2.1 Messaging Reliability

350 Most networks and their communication channels are not fairly reliable in real world. They are exposed to
351 unexpected circumstance variances, internal system errors and inexperienced users. As a result, the
352 messages for service requests and response could be lost and duplicated and their sequence confused.
353 These cases would cause serious results in business, which could give a fatal blow to a service unit.

354 Messaging reliability refers to the capability for messaging functionality which ensures the intention of
355 messaging for service units. The level of messaging reliability depends on the requirement of service
356 units. For example, a service unit could require a very restrict level of reliability in which a message is
357 transferred once and only once at any case (i.e. money transfer). In some cases, a service unit requires
358 at least one message transferred (i.e. request message for search). The sequence of messages could be
359 disregarded (i.e. request for an invoice). On the other hand, the sequence of messages could have major
360 effects to a business (i.e. request for stock trading).

361 The level of messaging reliability could be determined by the agreement of parties participating in a
362 business relationship. The business parties **MUST** provide the reliability functions which guarantee the
363 level of reliability more than the level agreed. The messaging reliability includes 4 basic factors, which can
364 be combined. Certain combinations are of particular interest due to their widespread application: exactly
365 once and ordered (also referred to as exactly once ordered).

- 366 • Transmitting at least once (guaranteed delivery)

367 Every message **MUST** be transmitted at least once or an error **MUST** be raised on at least one
368 endpoint. Some messages **SHOULD** be delivered more than once until an Acknowledgement is
369 received from the receiver. Each message **MUST** be transmitted at least once, otherwise both
370 receiver and sender **MUST** issue an error message. In addition, the sender **MUST** keep
371 retransmitting the message until Ack is received from the receiver.

- 372 • Transmitting at most once (guaranteed duplicate elimination)

373 Each Message **MUST** be transmitted at most once without duplication. Otherwise an error will be
374 raised on at least one endpoint. The receiver **MUST** block the duplicated message. Each message
375 **MUST** be transmitted at most once. The receiver **MUST** block the duplicated message. Each
376 message **MUST** be transmitted precisely once, otherwise both receiver and sender **MUST** issue an
377 error message. In addition, the sender **MUST** keep retransmitting the message until Ack is received
378 from the receiver. The receiver **MUST** block the duplicated message.

- 379 • Transmitting precisely once (guaranteed delivery and duplicate elimination)

380 Each message MUST be transmitted precisely once, otherwise both receiver and sender MUST
381 issue an error message. In addition, the sender MUST keep retransmitting the message until an
382 Acknowledgement is received from the receiver. The receiver MUST block the duplicated message.
383 This delivery assurance is the logical "and" for the two prior delivery assurances.

384 • Transmitting sequentially(guaranteed delivery order)

385 Messages in sequence MUST be transmitted from where they are created to the receivers who the
386 messages are intended to be orderly delivered to. To do this, the sender MUST sequentially send
387 the message with message sequence information embedded in the messages. And the receiver
388 would have to be able to rearrange the messages according to sequence information embedded in
389 the messages.

390 5.2.2 Transaction Integrity

391 Transactions running across multiple services over multiple domains need to maintain business integrity.
392 Traditionally, a transaction is a business processing unit (a unit of work) that involves one or more
393 services and is either completed in this entirety or is not done at all.

394 Transaction integrity refers to whether a service has functionality for processing transactions or a
395 transaction integrity platform environment can be applied. The transaction model of web services can be
396 divided into either a short-term transaction (atomic transaction) or a long-term transaction (business
397 activity).

398 ▪ Short-term transaction (atomic transaction, all-or-nothing property)

399 Short-term transaction is a transaction which requires a service locked for a short period of time
400 such as purchasing a book online. The major function of the transaction is to reset to default when a
401 request of the transaction is not processed or a request of a transaction is processed so that all the
402 changes resulting from the transaction are applied. This transaction is also called an Atomic
403 transaction and MUST satisfy the following 4 ACID (Atomicity, Consistency, Isolation, and
404 Durability) requisites.

405 • Atomicity

406 The transaction completes successfully (commits) or if it fails (aborts), all of its effects are undone
407 (roll-back)

408 • Consistency

409 Transactions produce consistent results and preserve application specific invariants

410 • Isolation

411 The results of a task are not shared with other transactions unless it is successfully completed.

412 • Durability

413 Once a transaction is successfully completed, its results SHOULD be permanently applied to a
414 system.

415 ▪ Long-term transaction (long-running transaction)

416 Long-term transaction refers to a transaction which requires a longer processing time or its
417 resources cannot be locked exclusively during processing. It is also referred to business activity.
418 Because a long-term transaction consists of some short-term transactions or independent web
419 services, Commit or Roll-back mechanisms of short-term transactions cannot be used. Therefore,
420 long-term transaction quality is evaluated by the following criteria, not by ACID attributes.

421 • Consistency

422 Long-term transaction SHOULD be able to change consistently the status of participating systems.

423 • Compensatory

424 Long-term transaction MUST support an independent and alternative flow to compensate for failed
425 transactions. Because long-term transactions consist of some short-term transactions or
426 independent web services, an alternative flow of processing is needed without individual processes
427 being reset to default.

428 5.2.3 Collaborability

429 The application of web service collaboration is prevalent to implement business processes. A business
430 process can be defined as the execution of activities according to a defined set of rules in order to
431 achieve a common goal between participants. Collaborability is the capability of a service platform to
432 define, control and manage service flow between participants. There are two types of collaborability:
433 orchestration which is executed or coordinated by single conductor and choreography which is executed
434 or coordinated by multiple participants.

- 435 • Orchestration

436 Orchestration is a technique used to compose hierarchical and self-contained service-oriented
437 business processes that are executed and coordinated by a single agent acting in a "conductor"
438 role [OASIS, Reference Architecture for Service Oriented Architecture]. In other words,
439 orchestration is the technique to define and execute a flow or procedure of services to achieve
440 business processing. An orchestration is typically implemented using a scripting approach to
441 compose service-oriented business processes. This typically involves use of a standards-based
442 orchestration scripting language. An example of such a language is the Web Services Business
443 Process Execution Language (WS-BPEL) [WS-BPEL].

- 444 • Choreography

445 Choreography is a technique used to characterize and to compose service-oriented business
446 collaborations based on ordered message exchanges between participants in order to achieve a
447 common business goal. [OASIS, Reference Architecture for Service Oriented Architecture] In other
448 words, choreography defines the sequence and dependencies of interactions between multiple
449 participants to implement a business process composing multiple web services. Choreography
450 differs from orchestration primarily in that each party in a business collaboration describes its part in
451 the service interaction in terms of public message exchanges that occur between the multiple
452 parties as standard atomic or composite services, rather than as specific service-oriented business
453 processes that a single conductor/coordinator (e.g., orchestration engine) executes [OASIS,
454 Reference Architecture for Service Oriented Architecture]. To be specific, choreography describes
455 the sequence of interactions for web service messages. WSDL describes the static interface and
456 choreography defines the dynamic behavior external interface. It is the Peer-to-Peer collaboration
457 model of exchanging messages among related partners as a part of a bigger business transaction
458 with many participants.

459 5.3 Relationship to other Standards

- 460 • WS-Reliability 1.1

461 WS-Reliability is a SOAP-based protocol for exchanging SOAP messages with guaranteed delivery,
462 no duplicates, and guaranteed message ordering.

463 URL: http://docs.oasis-open.org/wsrn/ws-reliability/v1.1/wsrn-ws_reliability-1.1-spec-os.pdf

- 464 • WS-ReliableMessaging 1.2

465 WS-Reliable Messaging is a protocol that allows messages to be transferred reliably between
466 nodes implementing this protocol in the presence of software component, system, or network
467 failures.

468 URL: <http://docs.oasis-open.org/ws-rx/wsrn/200702/wsrn-1.2-spec-os.html>

- 469 • WS-Context 1.0

470 WS-Context provides a definition, structuring mechanism, and service definitions for organizing and
471 sharing context across multiple execution endpoints

472 URL: <http://docs.oasis-open.org/ws-caf/ws-context/v1.0/wsctx.html>

- 473 • WS-Coordination 1.2

474 WS-Coordination specifies an extensible framework for providing protocols that coordinate the
475 actions of distributed applications.

476 URL: <http://docs.oasis-open.org/ws-tx/wstx-wscoor-1.2-spec.html>

- 477 • WS-AtomicTransaction 1.2
478 WS-AtomicTransaction specifies the definition of the Atomic Transaction coordination type that is to
479 be used with the extensible coordination framework described in WS-Coordination.
480 URL: <http://docs.oasis-open.org/ws-tx/wstx-wsat-1.2-spec-os/wstx-wsat-1.2-spec-os.html>
481 • WS-BusinessActivity 1.2
482 WS-BusinessActivity specifies the definition of two Business Activity coordination types:
483 AtomicOutcome or MixedOutcome, that are to be used with the extensible coordination framework
484 described in the WS-Coordination specification.
485 URL: <http://docs.oasis-open.org/ws-tx/wstx-wsba-1.2-spec-os/wstx-wsba-1.2-spec-os.html>
486 • WSBPEL 2.0
487 WSBPEL describes a business process activity using web services and defines the way they are
488 linked with each other. Using Orchestration, business process collaboration is composed. WS-BPEL
489 2.0 is approved as OASIS standard.
490 URL: <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>
491 • WS-CDL 1.0
492 Web Services Choreography Description Language is a language to describe XML based web
493 services collaboration as choreography. It is a standard for the decentralized business process.
494 URL: <http://www.w3.org/TR/2005/CR-ws-cdl-10-20051109/>

495 6 Manageability Quality

496 6.1 Definition

497 The more web services gain weight in business, the more the web service management scheme is
498 needed for maintaining web service quality. Web service can be managed not only locally by a web
499 service manager or provider, but also remotely by a consumer and a third party system. Web service
500 management may be a prerequisite for the foundation of trust between a service consumer and a
501 provider.

502 Manageability is defined as an ability which keeps a web service and its resources being manageable. At
503 this point, the web service resource includes the software and hardware components used by the web
504 service and a platform on which the web service operates. Manageability can be achieved by
505 implementing manageability capabilities, which are exposed as an access point, for each web service. A
506 manageability implementation means an implementation of a manageability endpoint and all of its
507 manageability capabilities. The manageability capability helps targeting a web service, provides a function
508 to monitor operational status, and controls operations along with web service protocol. As the
509 manageability capability enables a service consumer to use web services with reliability and stability, it
510 may be an important criterion for one to select a web service. The manageability is classified into 3 sub-
511 factors: informability, observability and controllability.

512 6.2 Sub Quality Factors

513 6.2.1 Informability

514 The management of a web service requires the primitive information to be settled in the implementation
515 phase in order to cope with troubles and to support a web service operation. Informability is a sub-quality
516 factor to measure whether the primitive information can provide enough to manage a web service. The
517 primitive information for managing a web service is divided into the manageability access information,
518 assessment of web service management capability and the web service primitive information.

- 519 • Manageability access information

520 The manageability capabilities are exposed as a web service access endpoint thus a manageability
521 consumer SHOULD get the access point for managing a web service and manageability access
522 information. There are two ways to achieve this. One is to implement an additional web service
523 whose functionality is to return a manageability endpoint for managing a web service. The other is
524 to implement each web service equipped with an operation which returns its own manageability
525 endpoint. The former has a disadvantage that a consumer has to know previously the reference of a
526 web service which informs the access point of manageability capabilities. In the latter, a web service
527 developer is burdened to implement an additional operation to inform the manageability endpoint.
528 By utilizing both, a manageability consumer SHOULD get the manageability endpoint precisely
529 through simple web service interface.

- 530 • Web service primitive information

531 The manageability capability can provide primitive information of a web service to be managed. The
532 primitive information includes web service properties (e.g. protocol version number, encryption
533 algorithm, messaging pattern, etc), description of web services and their resources, characteristics
534 of a manageability implementation, relation information between web services and resource. The
535 manageability endpoint SHOULD also have an identity capability to discriminate whether two web
536 services are the same by referring to their identity information.

537 6.2.2 Observability

538 Observability measures how effectively a manageability implementation can provide status information of
539 a web service. The status of a web service system can be revealed by gathering the values of

540 performance metrics. Therefore, observability can be evaluated by how effectively the metrics are taken
541 when gathering information. The effectiveness of gathering metrics is represented in three aspects:

- 542 • How much are metrics provided?
- 543 • How exactly metrics are provided?
- 544 • Are the metrics provided in real-time?

545 A manageability consumer can get the status information by two methods: monitoring and notification. In
546 the former, he can request the information anytime to a manageability endpoint actively, then a
547 corresponding manageability capability returns a metric value based on monitoring results. In the latter, a
548 manageability consumer subscribes previously significant events or issues to be notified. Then, when a
549 subscribed event or trouble occurs, he will be notified. The target information to be observed includes all
550 the operational status information such as performance metrics, a utilization ratio of memory, and a
551 history of messages.

552 **6.2.3 Controllability**

553 There may be a case that a web service and its resources have to be regulated for keeping a stable
554 status or coping with performance degradation. For example, a manageability consumer may increase
555 the size of a web service message queue when there are too many messages received at the same time.
556 If there is an error found in the messaging process, a web service platform SHOULD be stopped to cope
557 with the problem. Thus, Controllability measures whether a manageability implementation can provide
558 enough control functions to keep a web service in controllable status. The control functions are classified
559 into operation control functions and configuration control functions.

- 560 • Operation control functions

561 These are to change an operational status of a web service by executing commands such as start,
562 stop, fork, and exit to the web service or related resources.

- 563 • Configuration control functions

564 These are to modify the value of configuration parameters. For example, according to the change of
565 a circumstance, a manageability consumer wants to adjust the value of a web service configuration
566 such as a queue size, an encoding method and an encryption algorithm.

567 For measuring controllability, the scope, stability, voluntary, and easiness of control functions in a
568 manageability implementation MUST be evaluated.

569 **6.3 Relationship to other Standards**

- 570 • W3C Web Services Architecture

571 It defines the structure of web services including the state model of web services and the structure
572 for management. The standardization was completed in January, 2004. It consists of Web Services
573 Architecture for defining web services structure, Web Services Usage Scenarios, and Web Services
574 Management: Services Life Cycle for defining the state model of web services.

575 URL: <http://www.w3.org/TR/ws-arch/>

- 576 • OASIS Web Services Notification (WSN) v1.3

577 WSN is the OASIS standard to define the mechanism of asynchronous message exchange using
578 an event. It was completed in October, 2006. It consists of WS-BaseNotification to define the
579 message exchange mechanism of basic event method, WS-BrokeredNotification to define the
580 asynchronous message exchange mechanism using the broker like MOM and WS-Topics for
581 exchanging the event information.

582 URL: <http://www.oasis-open.org/specs/#wsnv1.3>

- 583 • OASIS Web Services Resource Framework (WSRF) v1.2

584 WSRF proposes the common framework for managing various resources that exist on networks.
585 It was completed on April, 2006. It consists of WS-Resource to define resources, WS-
586 ResourceProperties to define exchange method of the resources, WS-ResourceLifetime to define

587 lifecycle of the resources, WS-ServiceGroup for define the way for managing the numerous
588 resources as a group and WS-BaseFaults to define basic malfunctions that can occur during the
589 attributes management process. Although WS-Resource Metadata Descriptor is not approved as a
590 standard, it is very important for managing web services and is being used in TC such as WSDM.

591 URL: <http://www.oasis-open.org/specs/#wsrfv1.2>

592 • OASIS Web Services Distributes Management (WSDM) v1.1

593 WSDM proposes the framework for managing various resources on networks. It consists of MUWS
594 (Management Using Web Services) and MOWS (Management of Web Services).

595 URL: <http://www.oasis-open.org/specs/#wsdmv1.1>

596 7 Security Quality

597 7.1 Definition

598 Security quality is the degree of ability that can protect web services from various threats on
599 confidentiality, integrity and availability. Typical threats on web services environment are unauthorized
600 access, exposure, forgery and destruction of web services. These security threats can destroy web
601 services environment by identity theft, forgery of financial data and blockage of services. Therefore,
602 security quality is becoming significantly critical and essential for web service.

603 This security quality should be considered on two technical perspectives.

- 604 • Transport Level Security

605 Transport level security is to provide a secure data transfer on the transport layer. This is regardless
606 of the characteristics and complexities of web services because its implementation is based on
607 transport layer protocols and web services are applied on the application layer. Because security on
608 the parts of a message is not supported, it has a limitation that the security cannot be guaranteed
609 during intermediary processing.

- 610 • Message Level Security

611 Message level security is a method which provides security service using XML based message to
612 provide confidentiality and integrity of SOAP messages. Message level security uses End-To-End
613 model, thus it provides persistent security.

614

615 The sub quality factors of the security quality include encryption, authentication, authorization, integrity,
616 availability, audit, non-repudiation and privacy. Each quality factor is related to confidentiality, Integrity
617 and availability.

618 7.2 Sub Quality Factor

619 7.2.1 Encryption

620 Encryption is data protection and disposal control on web services with cryptographic functionality to
621 prevent unauthorized user access of confidential or sensitive information. Frequently 'encryption' is
622 regarded as 'confidentiality' in a narrow sense.

- 623 • Transport Level Encryption

624 Transport level encryption supports only the encryption of the entire message, using the
625 cryptographic features provided by SSL, TLS or IPSec protocol. It ensures confidentiality of data
626 when sending and receiving the data on the transport layer. However, if it has any intermediary
627 processing, the data should be decrypted and revealed to the intermediary.

- 628 • Message Level Encryption

629 To ensure confidentiality of web services' message, message level encryption is provided by either
630 the XML-Encryption or the cryptographic functionality (e.g. PGP for the attachment) in the WS-
631 Security. Especially, the XML-Encryption can encrypt the part of the message with the WS-
632 SecurityPolicy along with message protection policy, thus confidentiality of the service on transport
633 level can be improved.

634

635 Firstly, the encryption quality can be measured by whether the encryption feature is applied or not. If the
636 encryption feature is applied, the strength of the encryption function (e.g. AES-128-CBC is a stronger
637 encryption algorithm than 3DES-CBC.), the size of encryption key, or the life cycle of key will affect the
638 encryption quality.

639 7.2.2 Authentication

640 Authentication is the identification of services' consumer/ provider and the verification of the credential
641 that can be assured by the identification and trusted for the transmission.

- 642 • Transport Level Authentication

643 Transport level authentication is the same method as traditional web environment. Normally the
644 authentication method is ID/Password, X.509 based certificate, Kerberos and so on. And transport
645 level authentication is an authentication under point-to-point model. Thus, if it has no intermediary
646 processing, it can be trusted and provide many vendors with good interoperability. But, if
647 intermediary processing exists, the consumer's credential cannot be trusted and the provider cannot
648 know who the first origin is, because the identification is propagated and changed.

- 649 • Message Level Authentication

650 Message level authentication is an XML message based authentication using standard of W3C
651 such as WS-Security. Message level authentication method is sometimes similar to transport level
652 authentication. It can use ID/Password, X.509 based certificate token, or Kerberos token just
653 inserted in XML. And for more secure requirements and standard based interoperability, a more
654 special authentication method is used with XML token such as SAML, WS-Federation, Liberty, and
655 so on. But, because the interconversion is not supported among XML tokens, the provider must
656 consider STS (Security Token Service) in WS-Trust specification under the mixed XML token
657 environment.

658

659 The authentication quality can be measured by whether the authentication feature is applied or not. And it
660 can be the strength of the implementation of authentication method such as password policy, multi-factor
661 authentication or the possibility of bypass in an authentication mechanism.

662 7.2.3 Authorization

663 Authorization is the control over access on service/message for each actor's right. It is used to support
664 Confidentiality and Integrity. It uses various policies, access control models and security levels as means
665 of support of Authorization.

- 666 • Transport Level Authorization

667 Transport level authorization refers to the access control on the resources of users of the transport
668 channel. It is implemented on application, middleware – web application server, directory, or
669 security device using various security models such as RBAC (Rule based access control) and so on.
670 In this case, the Transport level authorization has the same limitation as the point-to-point model. In
671 essence the transport level authorization is decentralized control. Thus, Origin authorization policy
672 cannot persist under transport level authorization with intermediary processing.

- 673 • Message Level Authorization

674 Message level authorization is the XML messaged based authorization using standard of W3C such
675 as WS-Security. It is represented with XACML as service authorization policy and WS-
676 SecurityPolicy as message protected policy. Moreover, it can be encapsulated with the access right
677 defined as XACML or SAML to carry XACML. Using this function, the access right over the actual
678 resources is controlled.

679

680 The authorization quality can be measured by whether the authorization feature is applied or not. It can
681 be the possible bypass in an authorization mechanism.

682 7.2.4 Integrity

683 Integrity is to protect from unauthorized service/message modify, delete and create. It uses access
684 control and briefing message.

- 685 • Transport Level Data Integrity

686 Transport level data integrity refers to a feature such as the packet comparison and message digest
687 provided by IPSEC or TLS to provide the data integrity when sending and receiving data between
688 transport channels.

- 689 • Message Level Data Integrity

690 Message level data integrity refers to the data integrity of SOAP message level. It can be
691 guaranteed by XML-Signature in WS-Security. Also, XKMS to manage the digital signature for the
692 data integrity can be used.

693 The Integrity quality can be measured by whether the integrity feature is applied or not.

694 **7.2.5 Non-Repudiation**

695 Non-repudiation is to prevent receivers and senders from denying that they send and receive messages.
696 It uses digital-signature for non-repudiation.

- 697 • Transport Level Non-Repudiation

698 Under the transport layer, non-repudiation cannot be built using digital-signature. Usually it is
699 included in application or business logic. Hence, we do not mention non-repudiation of transport
700 level here.

- 701 • Message Level Non-Repudiation

702 Message level non-repudiation- can be built using XML-Signature in WS-Security. XKMS is used to
703 manage the digital signature for non-repudiation on the transport level. The non-repudiation quality
704 of the message level can be measured by whether the non-repudiation feature is applied or not.

705 **7.2.6 Availability**

706 Availability is to allow only authorized consumers to access services whenever they need. The techniques
707 such as IDS, IPS or Anti-DoS(Denial of Services) can be implemented to ensure availability.

- 708 • Transport Level Availability

709 Transport level availability refers to service continuity on the transport layer to prevent exhausting
710 web resources by excessive or malicious requests which can make services unavailable. It is
711 established by surveilling the packets with IDS, IPS or Anti-DoS equipment on the transport layer.

- 712 • Message Level Availability

713 Message level availability refers to service continuity on the message level to protect malicious XML
714 messages which can make services unavailable. It is accomplished by filtering and verifying
715 messages on the application level or XML firewall.

716 The availability quality can be measured by whether the availability feature is applied or not.

717 **7.2.7 Audit**

718 Audit is the capability to trace and verify activities and events on web services providing and consuming.
719 During the security audit process, security vulnerability or security attack can be identified from the traced
720 information.

- 721 • Transport Level Audit

722 Transport level audit is to trace and verify send/receive information on transport layer.

- 723 • Message Level Audit

724 Message level audit is to trace and verify request/response message on the application layer.

725 The audit quality can be measured by whether the audit feature is applied or not.

726 **7.2.8 Privacy**

727 Privacy is the protection of sensitive information of web services consumers and providers. To support
728 privacy, it is necessary to guarantee above, the sub quality factors of security quality. Additionally privacy
729 policy is required in compliance with law and regulations related with privacy.

730 The privacy quality can be measured by whether the protection of privacy information is implemented and
731 the privacy policy is defined appropriately.

732 **7.3 Relationship to other Standards**

- 733 • W3C XML Signature

734 XML digital signature standard

735 URL: <http://www.w3.org/TR/xmlsig-core/>

- 736 • W3C XML Encryption

737 XML encryption standard

738 URL: <http://www.w3.org/TR/xmlenc-core/>

- 739 • W3C XKMS (XML Key Management Specification)

740 Key management service standard which makes the integration between PKI and XML application
741 easy

742 URL: <http://www.w3.org/2001/XKMS/>

- 743 • OASIS WS-Security (Web Services Security: SOAP Message Security)

744 The standard to provide the authentication, integrity, non-repudiation, and confidentiality of SOAP

745 URL: [http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-
746 SOAPMessageSecurity.pdf](http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf)

- 747 • MS, VeriSign, IBM WS-SecurityPolicy (Web Services Policy)

748 The standard to provide the security policy applied on WS-Security.

749 URL: <http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/ws-securitypolicy-1.2-spec-os.html>

- 750 • OASIS SAML (Security Assertion Markup Language)

751 The standard to reliably exchange the authentication and approval information based on XML

752 URL: <http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf>

- 753 • OASIS XACML (eXtensible Access Control Markup Language)

754 The access control standard that consists of XML based policy language and access control
755 decision, request/response language

756 URL: <http://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf>

- 757 • MS, VeriSign, BEA, IBM WS-Trust (Web Services Trust Language)

758 The standard about issuing and exchange the security token and configuring trust relationship in
759 various trust domains

760 URL: <http://docs.oasis-open.org/ws-sx/ws-trust/v1.4/os/ws-trust-1.4-spec-os.html>

- 761 • MS, VeriSign, BEA, IBM WS-Federation (Web Services Federation Language)

762 The mechanism definition to intervene user identification, attributes, and authentication among web
763 services applications what belong to different security domains.

764 URL: <http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.html>

765 **8 Conformance**

766 A product, document or service conforms to this specification if it adopts or provides all the quality factors
767 and sub quality factors in this specification. If there are some quality factors or sub quality factors with no
768 value or not adopted in a product, then they should be supplied with no value or the value indicating “not
769 applicable” such as N/A.

770 The name and meaning of each quality factor or sub quality factor should not be changed. It is possible to
771 include additional quality factors or sub quality factors which are not given in this specification. For the
772 adopted or provided quality factors, a product, document or service should satisfy all of the MUST or
773 REQUIRED level requirements defined in the part of the quality factors in this specification.