OASIS

Management Using Web Services: Architecture

Working Draft 01, 5 November 2003

Document identifier:
WSDM-MUWS-Architecture-Draft

Location:

Editors:
John DeCarlo, The MITRE Corp. (jdecarlo@mitre.org)
Zulah Eckert, Hewlett-Packard, (zulah_eckert@hp.com)

Contributors:
Igor Sedukhin, Computer Associates (Igor.Sedukhin@ca.com)

Abstract:

Status:

Table of Contents

Deleted: 5
Deleted: 5
Deleted: 5
Deleted: 5
Deleted: 5
Deleted: 6
Deleted: 7
Deleted: 7
Deleted: 8
Deleted: 9
Deleted: 10

1 Introduction

[Provide an introductory chapter, indicating if any parts of it are non-normative.]

1.1 Terminology

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 **Error! Reference source not found.**

| Deleted: [RFC2119] |

39 **2 Context**

3 Concepts

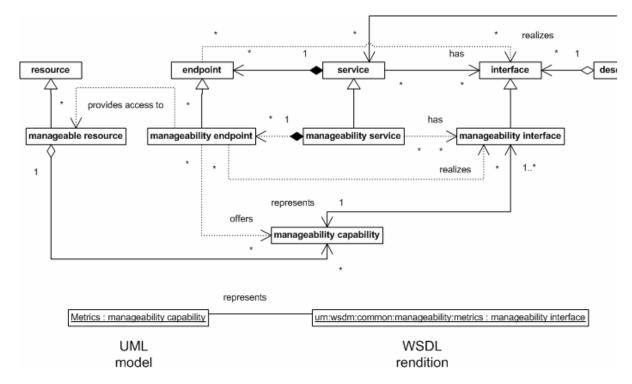## 3.1 MUWS Architecture Introduction

The MUWS Architecture being addressed in this document consists of the pieces needed for management using Web Services of generic Information Technology resources. This requires that manageability of the manageable resource be presented via Web Services, whether or not the resource is a Web Service itself. The Introduction/Context section (Section 1) placed this work in the larger context of Web Services Architecture and following sections will provide more detail about the components of the MUWS Architecture.

3.2 MUWS Architecture Scope

The MUWS Architecture being defined consists of the Provider of Manageability via Web Services (which consists of the Web Services endpoint(s), service(s), and interface(s) that expose the manageability capabilities for the manageable resource), the Consumer of Manageability, and other required infrastructure.

In addition to providing detailed information on the components that make up the Provider of Manageability, this document will address other items. The following items require specific notes on which parts are in and out of scope for the MUWS Architecture:

- The Consumer of Manageability (each manager which needs to manage some aspect of a manageable resource using MUWS is a consumer of Manageability). The Consumer must be able to make use of the manageability interface(s) provided by or on behalf of manageable resources. Conventional management applications that do not support MUWS will not be addressed at all in the MUWS Architecture. The Consumer of Manageability, like any Web Service consumer, must be able to send messages to, receive responses from, and possibly receive notifications from the manageability interface. There are no requirements imposed on the use of information received.

  NOTE: It is important to note that not every Consumer will have the same capabilities. For example, some may be able to process WSDL dynamically, others may not. Some may only be able to do monitoring, others may be able to do monitoring and configuring. This MUWS Architecture will refer to the Consumer in a generic sense, not requiring any particular implementation to provide any particular capability.

- The Manageable Resource. No constraints or requirements will be placed on the actual resource itself. In particular, the constraints and requirements will be put on the manageability endpoint and manageability interface to properly provide what manageability capabilities are available for that manageable resource via Web Services. It is entirely possible for there to be manageability capabilities that are not directly supplied by the manageable resource, but are inferred or calculated by another entity and offered by the manageability endpoint.

- Required infrastructure components. Examples include, but are not limited to, a Registry, a Policy Repository, or a Security service. They will be mentioned in the document where appropriate, and MUWS has requirements on these services, but they will not be defined here. Also, much of this work will be addressed via the MUWS Platform requirements.

3.3 MUWS Architecture Concept Diagram

85

86

87
88

89 A resource is a manageable resource if it has manageability, management
90 policy, and management related semantics [cite WSA]. In this document,
91 we are concerned with management using Web services which defines how
92 manageability of an arbitrary IT resource can be accessed via Web
93 services. Thus, manageability is one possible quality of a resource. '′
94 Manageability 'is composed of a number of capabilities. Each capability
95 has its own distinct semantics (e.g., could be expressed in a UML
96 model). Therefore, a manageable resource composes a set of
97 manageability capabilities. Figure ?, relates the concepts necessary
98 for management using Web services.

99

100 A Web service is an aggregate of endpoints each offering the service at
101 an address and accessible according to a binding. A service has a
102 number of interfaces that are realized by all of its endpoints. Each
103 interface describes a set of named messages that could be exchanged and
104 their format. Properly formatted messages could be sent to an
105 endpoint's address in a way prescribed by the binding. A description
106 (document, artifact) is composed of definitions of interfaces and
107 services. A description may contain both or either of the definitions.

108

109 In accordance with the Web services concepts, access to the
110 manageability for a resource must be provided by an endpoint. We call
111 such an endpoint a manageability endpoint. Implicitly, a manageability

112 endpoint belongs to a manageability service, which has a number of
113 manageability interfaces that are realized by manageability endpoints.
114 Thus, a single manageability interface represents all or part of a
115 manageability capability. Similarly, a single manageability capability
116 may be represented in one or more interfaces. The semantics of a
117 particular capability is represented in a set of possible message
118 exchanges and rendered in message formats grouped into one or more
119 interfaces.
120
121 For example, ability to offer metrics could be captured in a 'Metrics'
122 UML model which is, therefore, an instance of the manageability
123 capability concept. The semantics of offering metrics could be rendered
124 from the UML model into a WSDL interface description defined in a
125 "urn:wsdm:common:manageability:metrics" namespace. That would be an
126 instance of the manageability interface concept.

> **Comment:** This example needs to match our agreement on the "meta-model" for manageability capabilities. I would suggest that we put a place holder for an example and insert one once we have agreement on the meta model.

127
128 This specification defines the base set of manageability capabilities
129 that could be composed into a manageable resource or aggregated into
130 uber-capabilities. For example, a TotallyManagableResource uber-
131 capability could be defined that includes all of the base manageability
132 capabilities. Such uber-capability could also be composed into a
133 manageable resource, and in that sense, an uber-capability is
134 conceptually the same as any other capability. However, this
135 specification does not currently 5ttempt to define (identify) the
136 aggregated uber-capabilities and focuses on the definition of the base
137 set.

138 # 4 Logical Architecture

139 ## 4.1 Information Model

140 ## 4.2 Roles

141

142 This section documents the roles various components of the MUWS Architecture, as well as
143 related components, will have during management using Web Services. It is not intended to
144 constrain the locus of implementation, but instead is intended to document the required
145 components and which actions each is required to take.

146 NOTE: One application implementation may have many roles or a full role may be implemented
147 by a combination of many different applications.

148

149 The major roles are Consumer of the Manageability Service and Provider of the Manageability
150 Service. Related roles are Manageable Resource and related infrastructure components, such
151 as a Directory.

152

153 There are also actions only referred to here, because there is no direct relationship to the
154 manageability service, and standardization is not required. Such as getting a new manageability
155 service or component up and running for the first time.

156

157 ### 4.2.1 Consumer of Manageability Service

158

159 The Consumer of the Manageability Service plays a role in the management of manageable
160 resources. Because the Manageability Service is a Web Service, the Consumer must follow the
161 Web Services rules.

162 - The Consumer must send properly formatted messages (based on the WSDL describing
163   the service) to the appropriate Provider of the manageability service. .

164 - The Consumer must be able to locate the appropriate Provider for the manageable
165   resource being managed.

166 - The Consumer must be able to receive responses from the Provider.

167 - In order to receive Notifications, the Consumer must also provide a Web Service (making
168   it a specialized Provider of a Notification Receipt Web Service) that supports receiving
169   notifications from the Provider and responds appropriately.

170 - The Consumer may be capable of discovering manageable resources from a Provider
171   which has a relationship with another Provider or manageable resource or through a
172   Directory.

173 - The Consumer must follow the security requirements of the Provider and properly
174   authenticate with the Provider as well as using interoperable confidentiality and integrity
175   mechanisms.

176

4.2.2 Provider of Manageability

The Provider of the Manageability Service plays the largest role in the management of manageable resources via MUWS. The Provider supplies the Manageability Service for a manageable resource.

NOTE: The Provider may be implemented in the manageable resource or it may not. The Provider may supply the Manageability Service for more than one manageable resource. In other words, this is not intended to constrain the locus of implementation.

- The Provider must describe the Manageability Service provided for a manageable resource in WSDL.
- The Provider must be able to receive properly formatted messages as described in the WSDL.
- The Provider must be able to respond to properly formatted messages appropriately.
- The Provider may be able to generate Notifications and send them to a Consumer as indicated by the Consumer or via the Consumer's WSDL.
- The Provider must follow the security requirements of the environment.

4.2.3 Manageable Resource

The Manageable Resource must perform the business tasks it is normally required to do. Because there are no restrictions on the locus of implementation, the manageable resource may or may not implement the role of Provider of the Manageability Service.

4.2.4 Infrastructure Components

The Web Services Infrastructure Components are identified in this document as providing specific services that the Consumer or Provider requires in order to consume or provide the Manageability Service.

4.3 Process Model and Interaction Diagrams

4.4 Delegation Architecture

5 Implementation Architecture

6 References

6.1 Normative

.

214 # Appendix A. Acknowledgments

215 The following individuals were members of the committee during the development of this
216 specification:

217

218 # Appendix B. Revision History

Rev	Date	By Whom	What
1	30 October 2003	Zulah Eckert	Set up the original template
1	5 November 2003	Zulah Eckert and John DeCarlo	Add material on scope, roles, concept diagram, and other text

219

Appendix C. Notices

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's procedures with respect to rights in OASIS specifications can be found at 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 implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.

Copyright © **OASIS Open 2003.** *All Rights Reserved.*

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 paragraph are included on all such copies and derivative works. However, this document itself does not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document 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 RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.