OASIS 🕅

2 Web Services Distributed

3 Management: Management Using

4 Web Services (MUWS 1.0) Part 1

5 OASIS Standard, 9 March 2005

6 **Document identifier:**

wsdm-muws-part1-1.0

8 Location:

http://docs.oasis-open.org/wsdm/2004/12/wsdm-muws-part1-1.0.pdf

10 Editor:

7

9

11

1

William Vambenepe, Hewlett-Packard <vbp@hp.com>

12 Abstract:

- 13There are two specifications produced by the Web Services Distributed Management14technical committee: Management Using Web services (MUWS) and Management Of15Web services (MOWS, see [MOWS]). This document is part of MUWS.
- MUWS defines how an Information Technology resource connected to a network
 provides manageability interfaces such that the IT resource can be managed locally and
 from remote locations using Web services technologies.
- MUWS is composed of two parts. This document is MUWS part 1 and provides the fundamental concepts for management using Web services. MUWS part 2 [MUWS Part 2] provides specific messaging formats used to enable the interoperability of MUWS implementations. MUWS part 2 depends on MUWS part 1, while part 1 is independent from part 2.

24 Status:

- 25 This document is an OASIS standard.
- Committee members should send comments on this specification to the
 wsdm@lists.oasis-open.org list. Others should subscribe to and send comments to the
 wsdm-comment@lists.oasis-open.org list. To subscribe, send an email message to
 wsdm-comment-request@lists.oasis-open.org, with the word "subscribe" as the body of
 the message.
- 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 WSDM TC web page (http://www.oasisopen.org/committees/wsdm/).
- 35 The errata document for this specification is maintained at:
- 36 http://docs.oasis-open.org/wsdm/2004/12/wsdm-muws-part1-1.0-errata.pdf

37 Table of Contents

38	1	Intro	duction	3
39		1.1	Terminology	3
40		1.2	Notational conventions	4
41	2	Arch	itecture	6
42		2.1	Focus on Resources	6
43		2.1.1	Capabilities for Management	7
44		2.1.2	2 Isolation from Implementation	7
45		2.2	Composability	7
46		2.2.1	Low-end to High-end Manageability	9
47		2.3	Formal Representation of the Architecture	9
48	3	Usag	ge of the Web Services Platform	11
49		3.1	Properties	11
50		3.2	Operations	11
51		3.3	Events	11
52		3.4	Metadata	12
53		3.5	Addressing	12
54		3.6	Security	12
55	4	Com	mon Information Items	14
56		4.1	WSDM Event Format	14
57		4.1.1	XML Representation of the event	14
58		4.2	Manageability Endpoint Reference	15
59	5	Capa	abilities	16
60		5.1	Identity	16
61		5.1.1	Definition	16
62		5.1.2	Properties	16
63		5.2	Manageability Characteristics	18
64		5.2.1	Definition	18
65		5.2.2	2 Properties	18
66		5.3	Correlatable Properties	19
67		5.3.1	Definition	19
68		5.3.2	2 Information Markup Declarations	19
69		5.3.3	B Properties	20
70	6	Defir	ning a Manageability Interface	23
71	7	Refe	rences	24
72		7.1	Normative	24
73		7.2	Non-normative	24
74	A	ppendix	A. Acknowledgements	26
75	A	ppendix	B. Notices	27
76	A	ppendix		
77	A	ppendix	D. Properties Boolean Match Schema (Normative)	30
78				

79 **1** Introduction

Management Using Web Services (MUWS) enables management of distributed information
 technology (IT) resources using Web services. Many distributed IT resources use different
 management interfaces. By leveraging Web service technology, MUWS enables easier and more
 efficient management of IT resources. This is accomplished by providing a flexible, common
 framework for manageability interfaces that leverage key features of Web services protocols.
 Universal management and interoperability across the many and various types of distributed IT
 resources can be achieved using MUWS.

The types of management capabilities exposed by MUWS are the management capabilities generally expected in systems that manage distributed IT resources. Examples of manageability functions that can be performed via MUWS include:

- 90 monitoring the quality of a service
- 91 enforcing a service level agreement
- 92 controlling a task
- 93 managing a resource lifecycle

MUWS is designed to meet the requirements defined in the MUWS Requirements document
 [MUWS REQS]. Whenever possible, MUWS leverages existing Web services specifications to
 ensure interoperability, adoptability, and extensibility.

- There is a basic set of manageability capabilities defined in this specification. The only capability
 required by MUWS is the *Identity* capability defined in section 5.1.
- 99 To understand the various topics discussed in this specification, the reader should be familiar with 100 IT management concepts. In addition, the following assumptions are made:
- The reader is familiar with the Web Services Architecture [WSA].
- The reader is familiar with XML [XML 1.0 3rd Edition], XML Schema[XML Schema Part 1]
 [XML Schema Part 2], and XML Namespace [XNS].
- The reader is familiar with WSDL [WSDL], SOAP [SOAP] and WS-Addressing [WS-Addressing].
- The reader is familiar with WS SOAP Message Security [WSS].

The text of this specification, along with Appendix C and Appendix D, is normative with the
 following exception: the abstract, examples, UML diagrams and any section explicitly marked as
 non-normative.

110 **1.1 Terminology**

111 The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",

- 112 "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be 113 interpreted as described in RFC 2119 [RFC2119].
- 114 Furthermore, this specification defines and uses the following terms:
- 115 *Web service endpoint* an entity providing a destination for Web service messages. A Web
- service endpoint has an address (URI) and is described by the content of a WSDL 1.1 port
- element. This definition is consistent with the definition provided in the WS-Addressing specification [WS-Addressing].
- 119 *Web service interface* a group of operations described by the content of a WSDL 1.1 portType
- 120 element. These operations can provide access to resource properties and metadata.
- 121 Resource a logical or physical component of some subject domain, for example, a printer, a
- 122 magnetic storage disk, an application server, a CRM application or a car engine. wsdm-muws-part1-1.0

Copyright © OASIS Open 2003-2005. All Rights Reserved.

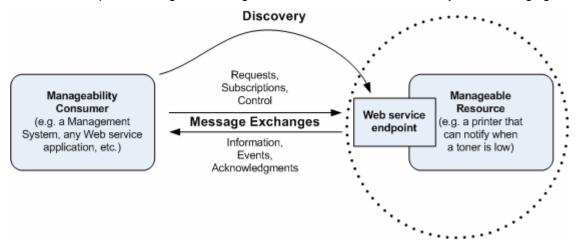
123 124	<i>Manageable resource</i> – a resource capable of supporting one or more standard manageability capabilities.			
125 126	Capability –a group of properties, operations, events and metadata, associated with identifiable semantics and information and exhibiting specific behaviors.			
127	Manageabi	<i>ility</i> – the ability to ma	nage a resource, or the ability of a resource to be managed.	
128	Manageabi	<i>ility capability –</i> a cap	pability associated with one or more management domains.	
129 130	Standard n specification		<i>lity</i> – a manageability capability that is defined by this	
131	Manageabi	<i>ility interface</i> –the cor	mposition of one or more manageability capability interfaces.	
132 133	<i>Manageabi</i> capability.	ility capability interfa	ce –a Web service interface representing one manageability	
134 135	-	ility consumer –a use e resources.	er of manageability capabilities associated with one or more	
136 137	<i>Manageabi</i> manageable		service endpoint associated with and providing access to a	
138 139			of knowledge relative to providing control over, and information cle, etc. of manageable resources.	
140	1.2 Not	ational conven	tions	
141 142 143			al syntax to describe the XML grammar of the information used bility interfaces. This syntax uses the following rules:	
	c	T he sumtain sum some s		
144 145	§	values.	as an XML instance, but data types appear instead of	
146 147	§	{any} is a placeholder in the XML Schema).	r for elements from some other namespace (like ##other	
148 149	§	The Cardinality of an characters to the item	attribute, element, or {any}, is indicated by appending as follows:	
150		?	none, or one	
151		*	none, or more	
152		+	one, or more	
153		no character	exactly one	
154	§	Items contained within	n the square brackets, [and], are treated as a group.	
155 156	§	Items separated by a syntactic alternatives.	and grouped within parentheses, (and), indicate	
157 158	§		onsecutive periods,, are used in XML start elements ites from some other namespace are allowed.	
159 160	§	The XML namespace of an attribute or an e	prefixes, defined in section 4, indicate the namespace element.	
161				
162 163	A full XML S specification		the XML information is available in Appendix C of this	
164				
105	When describing an instance of XML information, and in order to refer to an element or an attribute, this specification uses a simplified Xpath-like notation that is formally defined as follows:			
165 166 167				
			MI information and in order to refer to an element of	

168	Path = '/'?	(['@'? (NCName	QName '*')] ['(' (NCName QName '*'] ')') ['/' Path]?
169			
170	where:		
171 172 173	§	Schema [XMLS].	ML non-qualified name as defined by the XML In this case, the namespace is assumed to default e of this specification.
174 175	§	Q <i>Name</i> is an XM [XMLS].	IL qualified name as defines by the XML Schema
176	§	Symbol * denotes	s any name match.
177 178	§	•	s a path delimiter. When it appears as the first ath, it denotes the root of the XML document.
179 180	§		es a reference to an XML attribute. If absent then an the or * refer to an XML element.
181	§	Symbols (and)	denote a reference to an XML Schema type.
182			
183	For example:		
184 185	/E1/E2/@A	1	refers to an attribute, A1, of an element, E2, contained in element E1, which is a root of the XML document.
186			
187 188 189 190	E1/ns1:E2/	E3	refers to an element, E3, which is contained in element E2 which is contained in element E1, anywhere in the XML document. In this case element E2 belongs to the namespace mapped to the prefix ns1.
191 192 193 194 195	(ns2:T1)/E ⁻	1/ns1:E2/@A1	refers to an attribute, A1, on an element, E2, contained in element E1, as declared in the XML Schema type T1. In this case, the target namespace, T1, is mapped to the prefix ns2.

196 2 Architecture

197 This WSDM specification (MUWS) defines how the ability to manage, or, how the *manageability*

- of, an arbitrary *resource* can be made accessible via *Web services*. In order to achieve this goal,
- 199 MUWS is based on a number of Web services specifications, mainly for messaging, description,
- discovery, accessing properties, and notifications (section 3). Some of these Web services
 specifications are first presented in [MUWS Part 2].
- 202 The basic concepts of management using Web services can be illustrated by the following figure:



203 204

Figure 1: WSDM Concepts

A Web service *endpoint* provides access to a *manageable resource*. An example of a

manageable resource is a printer that indicates when its toner is low, or, a magnetic storage disk
 that reports its internal temperature,

208 A manageability consumer discovers the Web service endpoint and exchanges messages with

209 the endpoint in order to request information, subscribe to events, or, control the manageable

resource associated with the endpoint. An example of a manageability consumer is a

211 management system, or, a business automation process, or simply, any Web service application.

212 In order to discover the Web service endpoint providing access to a particular manageable

213 resource, a manageability consumer first obtains an Endpoint Reference (EPR), as defined by the

214 WS-Addressing specification [WS-Addressing], and then obtains any other required descriptions,

including, but not limited to, a WSDL document [WSDL], an XML Schema, or a policy document.
 MUWS uses the same mechanisms, for obtaining EPRs and their associated descriptions, as

217 used by regular Web service implementations, and their applications.

A Web service endpoint providing access to some manageable resource is called a *manageability endpoint*.

- To exchange messages with a manageability endpoint, a manageability consumer needs to understand all of the required descriptions for the endpoint. The manageability consumer sends messages targeted to the manageable resource by using information contained in the EPR, for
- example, an address and some reference properties (see [WS-Addressing]).

224 **2.1 Focus on Resources**

225 The WSDM specification focuses upon how access is provided to manageable resources.

226 Essentially, there exists a contract between a manageability consumer and a manageable

- 227 resource with respect to the ability of the consumer to understand what messages can be
- 228 exchanged between the consumer and the resource. Therefore, the central element and focal wsdm-muws-part1-1.0

Copyright © OASIS Open 2003-2005. All Rights Reserved.

point of the WSDM architecture is the manageable resource. The message patterns encapsulate
 access to resources into manageable resources instead of exposing message patterns to

231 indirectly access the resource through agents, proxies, observers, etc.

232 2.1.1 Capabilities for Management

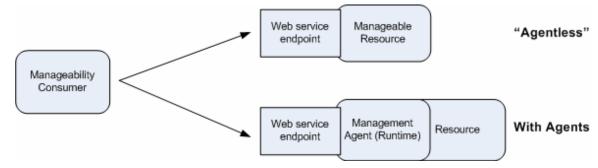
Manageability is one possible aspect of a resource. For example, a printer can (obviously!) print. Printing is the functional/operational aspect of the printer. However, the same printer may be able to indicate if it is on-line, or, if the toner has run out. Such indications compose manageability capabilities of the printer. A manageable resource may support some number of capabilities. Each capability has distinct semantics, for example, an ability to describe relationships among resources or an ability to indicate if the resource is on-line or off-line. An implementation of a manageable resource provides a set of manageability capabilities via Web service endpoints.

- 240 In WSDM terms, a manageability capability
- 241 § is uniquely identified,
- has defined semantics (such as those provided by any section in this specification that describes a new capability),
- is associated with a set of properties, operations, events (notifications) and metadata
 (including policies).

Each manageability capability defined in the WSDM specifications is extensible. New capabilities
 can be similarly defined, based on a particular resource manageability model, for example, DMTF
 CIM. MUWS provides mechanisms, patterns, and refinements, for defining new manageability
 capabilities and for discovering, identifying and using capabilities of a manageable resource.

250 **2.1.2 Isolation from Implementation**

The WSDM architecture focuses upon the manageable resource. This approach does not restrict choices of an implementation strategy. Moreover, WSDM isolates the manageability consumer from implementation specific aspects of a manageable resource or Web service endpoint. For example, a direct-to-resource, agent-less approach, or, an approach using management agents are equally valid implementations. Such implementation details are transparent to manageability consumers. Figure 2 illustrates this point:



257 258

Figure 2: Isolation from Implementation

259 2.2 Composability

Composability allows a manageable resource's implementation to support a non conflicting mix of
 some number of capabilities as well as features provided by the Web services platform. Parts of
 the composition incrementally enrich the implementation without incurring disruptions. For
 example, a SOAP message sent to a Web service endpoint may result in an order being placed.
 A similar SOAP message with WS-Security headers, signed and encrypted, may result in an
 order being placed in a secure manner. The mix of the order placement, plus the security
 implemented by a Web service endpoint, leveraged message-level composability. In other words,

wsdm-muws-part1-1.0 Copyright © OASIS Open 2003-2005. All Rights Reserved. the SOAP message is composed of an order placement request, plus the appropriate security headers, encryption and digital signatures.

The implementer of a manageable resource may create an appropriate composition of aspects
and capabilities offered to a manageability consumer via one or more Web service endpoints.
Within the context of WSDM, there are two kinds of composition that can manifest in an
implementation of a manageable resource, as follows:

- Composition of aspects of a Web services implementation for example, messaging, description, discovery, security, asynchronous notifications, etc. These implementation aspects are provided by the Web services platform and the respective standards specifications (see section 3).
- 277
 2. Composition of manageability capabilities, which may be classified into one of two categories, as follows:
 - a. Composition of common manageability capabilities for example, the ability to identify manageable resources, the ability to report and notify on a change of resource availability, or, the ability to report on how resources are related to each other. Such common manageability capabilities are defined in this specification in section 4 and in [MUWS Part 2]. Essentially these are base-line enablers of a richer set of resource manageability. This is similar to how SOAP and HTTP may be considered baseline enablers of Web services.
- b. Composition of resource-specific manageability capabilities for example, an ability to manage printers, or, an ability to manage network-connected devices. Other specifications define these manageability capabilities based on the available resource management model, (e.g. DMTF CIM), based on the needs of the management applications, based on the abilities of the resource (e.g. WSDM MOWS), or based on the needs of the management application.

The whole composition as implemented by a manageable resource is then accessible via a Web service endpoint. This is illustrated in Figure 3.

294

279

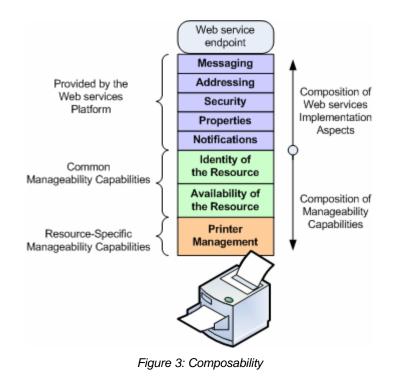
280

281

282

283

284 285



295 296

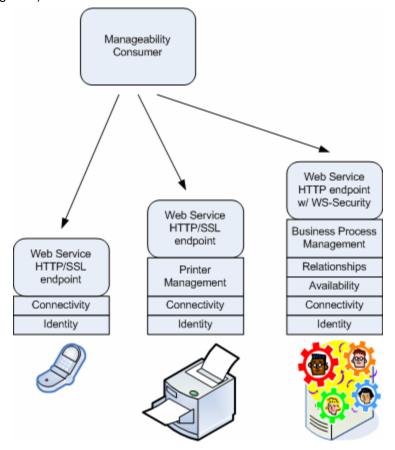
297

wsdm-muws-part1-1.0 Copyright © OASIS Open 2003-2005. All Rights Reserved.

298 2.2.1 Low-end to High-end Manageability

The WSDM architecture provides appropriate coverage from low-end manageability of small devices like mobile phones, to high-end manageability of very capable components like application servers and business processes. This range of coverage is achieved by the low barrier to entry placed upon a WSDM implementation: there are few normative requirements made by this specification and the specifications it depends on. Also, composability allows for additional manageability capabilities to be gradually introduced, based upon the availability of

- 305 management functions and processing power within an implementation of a manageable
- 306 resource. Manageability consumers can discover and make use of composed capabilities as 307 these capabilities become available. This flexibility is built into the foundation of the WSDM
- 308 architecture (Figure 4).



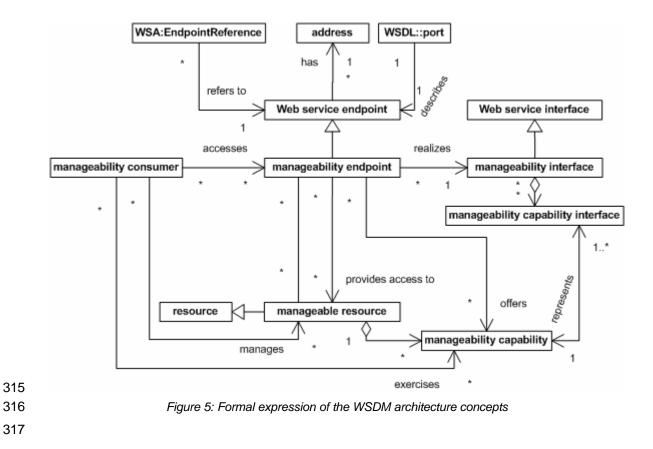
- 309
- 310

Figure 4: Low-end to High-end Manageability

311 **2.3 Formal Representation of the Architecture**

312 The following UML 2.0 model captures the WSDM MUWS concepts within the context of the

- 313 WSDL 1.1 [WSDL] and WS-Addressing [WS-Addressing] specifications. Figure 5 provides a
- 314 "mind map", or digest of the concepts described within the WSDM Architecture.



318 **3 Usage of the Web Services Platform**

As described in section 2, the foundation for MUWS is provided by the Web services platform. A number of Web services specifications may be composed with the WSDM specifications when implementing a manageability endpoint for a manageable resource. This and dependent specifications are used to represent different aspects of a capability: the properties, the operations, metadata, and events. [MUWS Part 2] introduces additional Web services

324 specifications to define an interoperable way to represent these capability aspects.

325 **3.1 Properties**

MUWS uses XML Schema ([XML Schema Part 1], [XML Schema Part 2]) to describe properties.
 A MUWS property is represented by a Global Element Declaration (GED). In order to create a
 property one MUST provide:

- the schema for the property,
- a description (in some form) of the semantics of the property,
- the cardinality of the property,
- any relevant metadata for the property.
- A manageable resource MUST expose an XML document containing, as top-level elements, all
 the properties of the manageable resource. This document is called the resource properties
 document for the resource.

336 **3.2 Operations**

MUWS uses [WSDL] to describe operations. The "operations" component of a capability
 corresponds to an operation, as defined by WSDL. In order to create an operation one MUST
 provide:

- a WSDL portType containing a WSDL operation corresponding to the operation,
- a description (in some form) of the semantics of the operation,
- any relevant metadata for the operation.

343 **3.3 Events**

344 Event types (as opposed to instances of event messages) are defined in MUWS by providing the 345 combination of a "topic" QName and a "message content" Global Element Declaration. The 346 "topic" QName need not be the QName of the "message content" element. A "topic" or a 347 "message content" element need not be exclusive to one event. However, the combination of a "topic" and a "message content" element MUST uniquely identify an event. The "message 348 349 content" element represents information that is transmitted as part of a notification message and 350 corresponds to an event instance. The "topic" provides information about why the event was 351 generated. In order to create a new event, one MUST provide:

- the corresponding "topic" and "message content" element,
- a description (in some form) of the semantics for the "topic" and "message content"
 element,
- any relevant metadata for the event.

A manageability endpoint SHOULD offer one or more events that correspond to a change in the properties it supports.

358 **3.4 Metadata**

359 MUWS allows definition of metadata on properties and operations. One such metadata item on 360 properties is whether it is *Mutable*. Mutability is defined as an indication of whether the value of a 361 property can change over time. Another metadata item on a property is whether it is *Modifiable*. 362 Modifiability is defined as an indication of whether the value of a property can be set explicitly, as 363 opposed to can not be set at all, or, can be set only as a side-effect of setting some other 364 property. Finally, a Capability is a metadata item that can be attached to a property, an operation 365 or an event. This metadata item contains a unique identifier for the capability. [MUWS Part 2] 366 describes additional metadata items.

For each property introduced in this specification, the value of these metadata items is described.
 However, MUWS does not specify if, or how, the value is made available to a consumer.

369 3.5 Addressing

MUWS makes use of the endpoint reference (EPR) construct, as defined in [WS-Addressing]. In addition, MUWS-compliant messages MUST comply with the rules in [WS-Addressing] regarding the use of SOAP headers, and, regarding how the content of the EPR constrains the messages sent to the endpoint.

374 **3.6 Security**

378

383

384

385

When evaluating the security requirements for resource management, it is important to delineate several aspects of Security technology;

- Identification: Presentation of a claimed identity
 - Authentication: Verification of proof of asserted identity
- Authorization: The information and mechanisms to allow appropriate authorized requests to resources and deny unauthorized requests.
- Message Integrity: The protection of messages in a message exchange from unauthorized modification.
 - Data Integrity: The protection of data from unauthorized modification.
 - Data confidentiality
 - Trust
- A complete security model addressing the requirements listed above needs to be provided for
 any management deployment. Profiles for different sets of requirements will be needed to ensure
 interoperable deployments.
- An explicit mapping to an authorization model at deployment time should be provided by aconformant management application.

To address security of messages, MUWS relies on generic Web services security mechanisms, including transport-level security (e.g. HTTP over SSL), OASIS Web Services Security messagelevel security [WSS], etc. The composition of appropriate security specifications and this specification provides a model for securing the messages exchanged during management using Web services realized by manageability endpoint implementations. The choice of concrete security mechanisms should be carried out by the implementers of the manageability endpoints and may conform to some profile.

- 398 Within an enterprise MUWS can be deployed like any other specification into the existing 399 enterprise security model. When managing between enterprises, security will need to be
- 400 developed in an ad hoc, pair-wise fashion at a messaging level.
- 401 This specification defines some metadata items for management. Whenever information related
- 402 to management metadata is being relied on, it is important to understand the environment in
- 403 which the metadata is being asserted. It may be needed to provide some data integrity
- 404 mechanisms to protect the information from unauthorized modification. It may also be needed to

- implement a set of authorization mechanisms to provide a way of identifying under what conditions information should be shared. 405
- 406

407 **4 Common Information Items**

408 4.1 WSDM Event Format

419

The WSDM Event Format defines an XML format to carry management event information. The format defines a set of basic, consistent data elements that allow different types of management event information to be carried in a consistent manner. The WSDM Event Format provides a basis for programmatic processing, correlation, and interpretation of events from different products, platforms, and management technologies.

The WSDM Event Format organizes management event data into three basic categories, the event reporter, the event source, and extensible, event-specific, situation data. Each category contains a few common properties, as found in most management events, and allows for

417 extensible, event-specific data. The WSDM Event Format has a flexible and extensible syntax..

- 418 To be effective, the WSDM Event Format MUST provides the following essential information:
 - the identification of the resource experiencing an event, called the source,
- the identification of the reporter of an event, known as the reporter. In most cases the source reports its own event, thus the identity of the reporter and the source are the same.
- 423 Typically, further information is also needed to describe the semantics of an event.

Additionally, an event MUST contain an *EventId* that is unique across event types within the source. An event may contain additional information related to the situation that has occurred or to the context within which it occurred. For example, message text, severity information or related

427 Application Response Measurement (ARM) instrumentation information. It is RECOMMENDED

428 that a container be used to encapsulate additional information that is significant to an event.

The base element of the WSDM Event Format is *muws-p1-xs:ManagementEvent*, as presented in the next section.

431 4.1.1 XML Representation of the event

432 The following is the XML representation of the WSDM MUWS management event container.

```
433
      <muws-pl-xs:ManagementEvent ...
434
        muws-p1-xs:ReportTime="xs:dateTime"?>
435
436
         <muws-pl-xs:EventId>xs:anyURI</muws-pl-xs:EventId>
437
438
         <muws-pl-xs:SourceComponent ...>
439
           <muws-pl-xs:ResourceId>xs:anyURI</muws-pl-xs:ResourceId> ?
440
           <muws-p1-xs:ComponentAddress>{any}</muws-p1-xs:ComponentAddress> *
441
           \{any\}*
442
        </muws-pl-xs:SourceComponent>
443
444
        <muws-p1-xs:ReporterComponent ...>
445
           <muws-pl-xs:ResourceID>xs:anyURI</muws-pl-xs:ResourceId> ?
446
           <muws-p1-xs:ComponentAddress>{any}</muws-p1-xs:ComponentAddress> *
447
           \{any\}^*
448
        </muws-pl-xs:ReporterComponent> ?
449
         \{any\}^*
450
      </muws-pl-xs:ManagementEvent>
```

451 Where the clauses are described as follows:

452 **muws-p1-xs:ManagementEvent**: The wrapper element used for management event messages.

- 453 muws-p1-xs:ManagementEvent/@muws-p1-xs:ReportTime: The date and time when the
 454 event was reported. If the value does not include a time zone designation, or use 'Z' for UTC,
 455 then the value MUST be interpreted as having a time zone of UTC. The value of *ReportTime*456 MUST provide a granularity as precise as is supported by the generating platform. This attribute
 457 is RECOMMENDED.
- 458 muws-p1-xs:ManagementEvent/muws-p1-xs:EventId: The primary identifier for an event. This 459 element MUST be unique within the scope provided by the manageability implementation for the 460 source resource. This element MAY be used as the primary key for the event. This element is 461 provided for management functions that require events to have an identifier. It is of type URI and 462 is REQUIRED.
- 463 **muws-p1-xs:ManagementEvent/muws-p1-xs:SourceComponent**: The identification of, or 464 reference to, the source associated with an event. This element is REQUIRED.
- 465 muws-p1-xs:ManagementEvent/muws-p1-xs:SourceComponent/ResourceId: A specification
- of an identifier of a manageable resource associated with an event. This is an OPTIONAL
 property. This property is intended as an identifier to be used, for example, in correlation, so that
- 468 management consumers can ensure that information contained in the *muws-p1-*
- 469 *xs:ManagementEvent* pertains to a given manageable resource. If provided, this element MUST 470 correspond to the *muws-p1-xs:ResourceId* property (defined in section 5.1.2) for the source
- 470 correspond to the *muws-p1-xs.Resourceid* property (defined in section 5.1.2) for the source 471 associated with an event.
- 472 muws-p1-xs:ManagementEvent/muws-p1-xs:SourceComponent/muws-p1-
- 473 xs:ComponentAddress: Contains the specific elements used to identify the address of a
- 474 component. If this element contains more than one child element, each child element represents475 an alternate address of the same source. This element is RECOMMENDED to improve
- 475 an alternate address of the same source. This element is RECOM476 interoperability.
- 477 muws-p1-xs:ManagementEvent/muws-p1-xs:SourceComponent/muws-p1-
- 478 xs:ComponentAddress/{any}: XML open content including any XML representation of the
 479 component address. One commonly used address type is a Web service address, such as an
 480 EPR as defined by [WS-Addressing]. In the case where the source is a manageable resource, it
 481 is RECOMMENDED that the *muws-p1-xs:ManageabilityEndpointReference* element, as defined
 482 in section 4.2, be used as the address type.
- 483 muws-p1-xs:ManagementEvent/muws-p1-xs:ReporterComponent: Provides the identification 484 of, or reference to, the reporter associated with an event. This is a REQUIRED property only if the 485 reporter is different from the source. Otherwise, this element is OPTIONAL. When this element is 486 absent the reporter is asserted to be the same as the source. The content of this element is the 487 same as the content of the *ManagementEvent/SourceComponent* element except that the 488 definitions apply to the reporter rather than the source.
- 489 muws-p1-xs:ManagementEvent/{any}: XML open content providing a container for additional
 490 data associated with an event. Among other things, this is where the "message content" Global
 491 Element Declaration introduced in section 3.3 is inserted. MUWS Part 2 defines some additional
 492 element that could be included using this wildcard.

493 **4.2 Manageability Endpoint Reference**

- 494 MUWS defines the following element to represent a reference to a manageability endpoint:
- 498 The element is an EPR as defined by [WS-Addressing]. The EPR provides a reference to a 499 manageability endpoint.

500 **5 Capabilities**

501 There is a minimum set of manageability capabilities that an implementation of a manageability 502 endpoint must support in order to comply with the MUWS specification.

503 A manageability capability defines properties, operations and events to support domain-specific 504 tasks. Details of a manageability capability are exposed by a manageable resource.

505 A manageable resource MAY also define a new resource-specific manageability capability.

506 A manageable resource SHOULD extend a MUWS manageability capability with a resource-

507 specific manageability capability that uses similar semantics. A manageable resource is not

- 508 required to extend a MUWS manageability capability when a resource-specific manageability
- 509 capability uses different semantics than the set of MUWS manageability capabilities.
- 510 In this section the following namespaces are used unless otherwise specified. The table below 511 lists each prefix and a corresponding namespace URI.

Prefix	Namespace
muws-p1-xs	http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part1.xsd
pbm	http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-pbm.xsd
XS	http://www.w3.org/2001/XMLSchema
wsa	http://schemas.xmlsoap.org/ws/2004/08/addressing

512 **5.1 Identity**

- 513 The manageability capability URI for the *Identity* capability is
- 514 http://docs.oasis-open.org/wsdm/2004/12/muws/capabilities/Identity

515 **5.1.1 Definition**

- 516 The goal of the Identity capability is to establish whether two entities are the same. This is a 517 required capability and it MUST be provided by every manageability endpoint. Observe that this
- required capability and it model be provided by every managed bind point observe and requirement does not preclude the manageability endpoint from applying a security policy
- 519 preventing some requesters from accessing this, or another, capability.
- 520 In addition, this capability is used as a "marker" interface enabling a manageability consumer to 521 learn if an endpoint is a manageability endpoint.
- 522 Figure 6 shows the UML representation of MUWS Identity.
- 523

MUWS::Identity
Resourceld[1]

524 525

Figure 6: MUWS Identity

526 **5.1.2 Properties**

- 527 The following is the specification of the property defined by the Identity capability.
- 528 <muws-pl-xs:ResourceId>xs:anyURI</muws-pl-xs:ResourceId>
- 529 The following is an example property instance for the property defined by the Identity capability. wsdm-muws-part1-1.0

Copyright © OASIS Open 2003-2005. All Rights Reserved.

530 531 532	<muws-pl-xs:resourceid> http://example.com/resource/diskDrive/9F34AD35B </muws-pl-xs:resourceid>
533 534	Note that <i>Resourceld</i> is an opaque identifier of a resource managed through a manageability endpoint. <i>Resourceld</i> is a read-only, mandatory property with a cardinality of 1.
535	This property has the following metadata:
536 537 538	It is not <i>Mutable.</i> It is not <i>Modifiable.</i> Its <i>Capability</i> is "http://docs.oasis-open.org/wsdm/2004/12/muws/capabilities/Identity".
539	The following constraints are applicable to Resourceld:
540 541 542 543	• Globally unique: A manageability endpoint MUST create the <i>Resourceld</i> URI in a way that ensures that the <i>Resourceld</i> is unique to the resource managed through the manageability endpoint and globally unique. This specification does not prescribe the means by which global uniqueness is achieved.
544 545 546	 Uniqueness in time: A Resourceld MUST NOT be reused by the implementation of a manageability endpoint for another resource, even after the original resource no longe exists.
547 548 549 550 551 552 553	 Consistency across endpoints: An implementation of a manageability endpoint SHOUL use a <i>Resourceld</i> that is suggested by the characteristics of a resource. This is possib when, for example, a <i>Resourceld</i> is retrievable from a resource by a manageability endpoint, or, an application of MUWS to a given domain specifies a method for building <i>Resourceld</i> based upon characteristics of resources populating the domain. It is not guaranteed that different manageability endpoints associated with the same resource v in all cases, return the same <i>Resourceld</i>.
554 555 556	 Consistency within an endpoint: An implementation that exposes several manageabilit endpoints for the same resource MUST report the same <i>ResourceId</i> at each manageability endpoint.
557 558 559 560 561	• Persistence: A manageability endpoint SHOULD return the same <i>ResourceId</i> during the entire lifetime of the manageability endpoint, including across power cycles of the manageability endpoint. Resources that are not able to persist a <i>ResourceId</i> across power cycles of a manageability endpoint SHOULD try to provide a consistent <i>ResourceId</i> via predictable identifier generation or delegation of identity assignment.
562 563 564 565 566 567 568 569	 Equality: If two reported <i>Resourcelds</i> are equal, then the consumer knows that the two manageability endpoints represent the same resource. The converse proposition is no necessarily true: two different <i>Resourcelds</i> could conceivably correspond to the same resource. It is strongly RECOMMENDED that this condition be avoided in a conscious and deliberate manner, as some managers may not be able to distinguish that two different reported identifiers are, in fact, associated with the same manageable resource Thus, manageability consumers would be forced to treat every identifier as correspond to a unique manageable resource.
570 571	Note that a manageability consumer MUST NOT assume that two manageability endpoints represent two different resources solely because the two reported <i>Resourcelds</i> are different.
572 573 574 575 576	Since the <i>Resourceld</i> is defined as opaque, this specification does not allow a consumer to inferring any characteristic of a resource by examining a <i>Resourceld</i> , other than comparing the <i>Resourceld</i> to another <i>Resourceld</i> as one way of establishing oneness. For example, one possible way to construct a <i>Resourceld</i> and ensure its uniqueness is to use a UUID wrapped in URI.
577 578	Note that this specification does not define equivalence of URIs and the consumer should deci which level of the comparison ladder, as defined in section 6 of [RFC2396bis], is appropriate to

578 which level of the comparison ladder, as defined in section 6 of [RFC2396bis], is appropriate to 579 use for this comparison. 580 MUWS defines an additional mechanism for establishing oneness of two resources. This 581 mechanism, called *Correlatable Properties* is defined in the section 5.3.

582 **5.2 Manageability Characteristics**

583 The manageability capability URI for the *Manageability Characteristics* capability is 584 http://docs.oasis-open.org/wsdm/2004/12/muws/capabilities/ManageabilityCharacteristics

585 **5.2.1 Definition**

- 586 The Manageability Characteristics capability defines properties providing information about the 587 characteristics of a manageability endpoint implementation rather than the resource.
- 588 Figure 7 shows the UML representation of MUWS Manageability Characteristics.
- 589

MUWS::ManageabilityCharacteristic	cs
ManageabilityCapability[0*]	

- 590
- 591

Figure 7: MUWS Manageability Characteristics

592 **5.2.2 Properties**

593 The following is the specification of the property defined by the Manageability Characteristics 594 capability.

595	<muws-p1-xs:manageabilitycapability></muws-p1-xs:manageabilitycapability>
596	xs:anyURI
597	

598 The following are example of property instances for the property defined by the *Manageability* 599 *Characteristics* capability.

600 <muws-p1-xs:ManageabilityCapability>

```
601 http://docs.oasis-open.org/wsdm/2004/12/muws/capabilities/Identity
```

602 </muws-p1-xs:ManageabilityCapability> 603 <muws-p1-xs:ManageabilityCapability>

- 604 http://example.com/capabilitys/FooCapability
- 605 </muws-p1-xs:ManageabilityCapability>
- 606 Note that **ManageabilityCapability** contains a URI identifying a manageability capability that is 607 supported by a manageable resource. The cardinality of this property is zero to unbounded.
- 608 This property has the following metadata:
- 609 It is not *Mutable*.
- 610 It is not *Modifiable*.
- 611 Its Capability is "http://docs.oasis-
- 612 open.org/wsdm/2004/12/muws/capabilities/ManageabilityCharacteristics".
- A manageability interface is said to provide a capability if it supports all of the required properties,
- 614 events, operations and metadata defined by the capability. This does not preclude the
- 615 manageability endpoint from applying a security policy preventing some requesters from
- 616 accessing this, or another, capability.
- 617 There SHOULD be one ManageabilityCapability property instance for each manageability
- 618 capability provided by a manageability interface. For capabilities extending a base capability, both
- 619 the extension and the base capability MUST be listed. Marking a property, operation or event as
- 620 part of a capability is considered a hint for the consumer of a manageability endpoint. The
- 621 meaning of such a hint is defined by the capability. As a result, the *ManageabilityCapability*

622 property facilitates discovery and introspection by providing a hint to the manageability consumer 623 about what requests can be sent to the manageability endpoint.

624 **5.3 Correlatable Properties**

625 The manageability capability URI for the *Correlatable Properties* capability is 626 http://docs.oasis-open.org/wsdm/2004/12/muws/capabilities/CorrelatableProperties

627 **5.3.1 Definition**

628 The Correlatable Properties capability allows a manageability endpoint to expose its 629 understanding of which property values could be compared when establishing that the 630 manageability endpoint in guestion and another manageability endpoint correspond to the same 631 resource. This is especially useful in the case where the two manageability endpoints are unable 632 to return the same ResourceId for a resource. For example, one manageability endpoint may 633 enable a temperature control capability for a SCSI hard disk drive, and another manageability 634 endpoint may enable a capacity management capability for the same SCSI hard disk drive. Each 635 manageability endpoint may return its own unique Resourceld due to implementation 636 requirements or constraints (e.g. firmware). However, implementers of a manageability endpoint 637 may be aware of some unique resource-specific property values that can indicate if two 638 manageability endpoints correspond to the same resource. In the SCSI example, correlatable 639 properties could be host IP, bus #, channel #, SCSI ID, LUN ID. If the values of those property 640 instances match, then one could be fairly certain that multiple manageability paths are provided to the same SCSI resource. 641

Using the CorrelatableProperties capability, both manageability endpoints may expose their
understanding of what resource property values need to match in order to establish a correlation
between manageable resources. The manageability consumer uses this information to evaluate
and establish such a correlation.

646 Note that if the *Resourcelds* returned by both manageability endpoints are the same but the 647 correlatable properties do not match, then the resources should be considered the same, as the

648 Identity capability takes precedence over *Correlatable Properties* capability. Typically,

649 manageability consumers will not evaluate correlatable properties if the two manageability

650 endpoints return the same *ResourceId*.

- 651 Figure 8 shows the UML representation of MUWS Correlatable Properties.
- 652

MUWS::CorrelatableProperties
CorrelatableProperties[0*]

653

654

Figure 8: MUWS Correlatable Properties

The exposure of the information provided as part of this capability allows clients to understand the information used to uniquely identify the resource. This may allow a nefarious client to spoof the presence of the resource. This is particularly true if it is obvious how to generate or construct the *Resourceld* from these properties. These properties should be used and exposed with this risk in mind. The *CorrelateableProperties* property should receive the same level of protection as the *ResourceID*.

661 **5.3.2 Information Markup Declarations**

662 There are three elements, as defined by this specification, providing a simple property boolean 663 match (PBM) dialect that can be used to express a correlation condition for correlatable 664 properties. This condition is expressed based on values of properties of the two resources that 665 are compared through the correlatable properties mechanism. These elements are defined in a 666 separate namespace, from the rest of the MUWS specification, as follows: wsdm-muws-part1-1.0

Copyright © OASIS Open 2003-2005. All Rights Reserved.

667	<pbm:match>xs:QName</pbm:match>
-----	---------------------------------

- 668 This element evaluates to true if the values of the properties for the given QName match for the 669 two resources.
- 670 <pbm:MatchAny>(<pbm:Match/>|<pbm:MatchAll>)*</pbm:MatchAny>
- This element evaluates to true if any of the enclosed *Match* and/or *MatchAll* conditions evaluate to true.
- 673 <pbm:MatchAll>(<pbm:Match/>|</pbm:MatchAny>)*</pbm:MatchAll>
- This element evaluates to true if all of the enclosed *Match* and/or *MatchAny* conditions evaluate to true.

676 **5.3.3 Properties**

677 The following is a definition of the property defined by the *Correlatable Properties* capability.

678	<muws-p1-xs:correlatableproperties< th=""></muws-p1-xs:correlatableproperties<>
679	Dialect ="xs:anyURI"
680	NegativeAssertionPossible="xs:boolean"
681	$\{any\}$ *
682	*

This property indicates, from the perspective of the manageability representation, which property values, conditions and expressions are used to correlate a manageable resource. The cardinality of the property is zero to unbounded.

?>

- 686 This property has the following metadata:
- 687 It is Mutable.
- 688 It is not Modifiable.
- 689 Its Capability is "http://docs.oasis-
- 690 open.org/wsdm/2004/12/muws/capabilities/CorrelatableProperties".
- The value of this property is the correlation expression. The format of the correlation expression is determined by the *Dialect* attribute. This specification defines three possible dialect values. An additional dialect value can be defined to provide additional functionality. A manageability
- representation can offer several instances of the *muws-p1-xs:CorrelatableProperties* property,
- 695 using the same, or different, dialects. A manageability consumer may evaluate a *muws-p1-xs:*
- 696 *CorrelatableProperties* property in any dialect that it understands. Support for a particular dialect 697 is optional.
- 698 The dialects defined by this specification are:
- Simple Property Boolean Match
- The URI for this dialect is http://docs.oasis-open.org/wsdm/2004/12/pbm.
 The content of the property is as described in section 5.3.2. If all top-level match conditions evaluate to true, then a correlation between manageable resources is
- 703 established.

704

- XPath 1.0
- 705The URI for this dialect is http://www.w3.org/TR/1999/REC-xpath-19991116.706The content of the property is an [XPath 1.0] expression. When retrieved as a property707form a manageable resource, the XPath expression is evaluated on properties of another708manageability resource. If the XPath expression evaluates to a Boolean value of *true*, or709if it evaluates to a non-empty, non-boolean value, without any errors, then a correlation is710established between the manageable resources.
- XPath 2.0
- The URI for this dialect is http://www.w3.org/TR/xpath20/.
- 713 The content of the property is an [XPath 2.0] expression. This XPath expression is

- evaluated on a resource properties document of another manageability representation. If
 the XPath expression evaluates to a Boolean value of *true*, or if it evaluates to a nonempty, non-boolean value, without any errors, then a correlation is established between
 the manageable resources.
- The optional *NegativeAssertionPossible* attributes express whether a negative result from the
 evaluation of the correlation expression implies that the resources are necessarily different.
 The default value is false.
- If NegativeAssertionPossible is false, only a positive match is meaningful to the consumer. In other words, if the correlation expression evaluates successfully, according to the evaluation rules defined by the dialect, then a consumer can consider the resource representations to represent the same resource. If the correlation expression does not evaluate successfully, then the consumer can not infer whether the resource representations represent different resources.
- If NegativeAssertionPossible is true, a positive match still means that the resources are the same. But a negative match now means that the resources are guaranteed to NOT be the same.

730 **5.3.3.1 Examples of use**

- Consider the following two simplified sets of properties, obtained through two differentmanageability endpoints:
- 733 Properties obtained through manageability endpoint ME1:

734	<print:printerresourcepropdoc></print:printerresourcepropdoc>
735	
736	<print:printermodel>PrintCo SuperJet 5000</print:printermodel>
737	<print:location>Building 42 lower pillar D4</print:location>
738	<print:owner>Sir Printalot</print:owner>
739	<print:ipaddress>15.244.62.41</print:ipaddress>
740	<foo:name>Baby got ink</foo:name>
741	<muws-pl-xs:correlatableproperties< th=""></muws-pl-xs:correlatableproperties<>
742	Dialect="http://docs.oasis-open.org/wsdm/2004/12/pbm">
743	<pre><pbm:matchany></pbm:matchany></pre>
744	<pbm:match>print:IPAddress</pbm:match>
745	<pre><pre>matchAll></pre></pre>
746	<pre><pbm:match>foo:Name</pbm:match></pre>
747	<pbm:match>print:PrinterModel</pbm:match>
748	<pre><pre>cpbm:Match>print:Location</pre></pre>
749	<pbm:match>print:Owner</pbm:match>
750	
751	
752	
753	

754 Properties obtained through manageability endpoint ME2:

755	<print:printerresourcepropdoc></print:printerresourcepropdoc>
756	
757	<print:printermodel>PrintCo UltraJet 40</print:printermodel>
758	<print:location>Building 42 lower pillar D4</print:location>
759	<print:owner>Sir Printalot</print:owner>
760	<pre><pre><pre><pre>int:IPAddress>15.244.10.89</pre></pre></pre></pre>
761	<foo:name>Baby got ink</foo:name>
762	

- 763 The CorrelatableProperties property, as provided through manageability endpoint ME1, asserts
- that if a manageability representation provides a view of a resource which either has the same
- 765 *IPAddress* as ME1, or, has the same *Name*, *PrinterModel*, *Location*, and *Owner* as ME1, then 766 these two manageability endpoints represent are the same printer. In this example, since the

- 767 IPAddress doesn't match and the PrinterModel is different, the correlation is not established and 768 the consumer cannot deduce that the two printers are the same.
- 769 Note that since the NegativeAssertionPossible attribute is not specified on CorrelatableProperties 770 it takes the default value of false. Therefore, the consumer cannot assume that the resources are
- 771 indeed two different printers. At this point, the consumer still cannot infer whether the two
- manageability endpoints correspond to the same printer or not. 772
- Properties obtained through manageability endpoint ME3: 773

774 775	<pre><print:printerresourcepropdoc></print:printerresourcepropdoc></pre>
776	<muws-pl-xs:correlatableproperties< th=""></muws-pl-xs:correlatableproperties<>
777	Dialect=http://www.w3.org/TR/1999/REC-xpath-19991116
778	NegativeAssertionPossible="false">
779	boolean(/print:PrinterResourcePropDoc/print:LastJob/print:JobID="5622654845
780	1262") and
781	<pre>boolean(/print:PrinterResourcePropDoc/print:LastJob/print:JobOriginator="15</pre>
782	.244.30.30")
783	
784	
785	Properties obtained through manageability endpoint ME4:
100	
786	<print:printerresourcepropdoc></print:printerresourcepropdoc>
787	·
788	<print:lastjob></print:lastjob>
789	<print:jobid>56226548451262</print:jobid>
790	<print:joboriginator>15.244.30.30</print:joboriginator>

- <print:JobDate>2004-03-11T11:30:56Z</print:JobDate>
- 791 792 </print:LastJob> 793 </print:PrinterResourcePropDoc>

794 The CorrelatableProperties property, as provided through manageability endpoint ME3, asserts that if a manageability endpoint provides a view of a resource for which the JobID of the last job is 795 796 56226548451262, and the JobOriginator of the last job is 15.244.30.30, then these manageability 797 endpoints represent the same printer. In this example, the condition is satisfied, so the consumer knows that ME3 and ME4 correspond to the same physical printer. Note that, as the example 798 799 shows, with this dialect the consumer only needs to retrieve the CorrelatableProperties property and no other property from ME3 to check correlation. From ME4 it needs to retrieve the 800 801 properties needed to evaluate the XPath expression. In this example, NegativeAssertionPossible 802 is set to false, thus a negative result would not have guaranteed that the printers behind ME3 and ME4 are indeed different. 803

6 Defining a Manageability Interface

Implementers of manageability endpoints are free to expose additional manageability capabilities
beyond those defined in MUWS. An additional capability is represented by a set of manageability
capability interfaces. The properties defined in a new capability must be defined as XML Schema
Global Element Declarations. The operations defined in a new capability are represented as
WSDL 1.1 operations. Furthermore, a manageability endpoint offering a new capability is free to
ignore all standard manageability capabilities defined by MUWS except for the *Identity* capability.
The MUWS *Identity* capability is REQUIRED.

812 MUWS-compliant manageability endpoints SHOULD also comply with the WS-I Basic Profile 813 version 1.1 [BP].

wsdm-muws-part1-1.0 Copyright © OASIS Open 2003-2005. All Rights Reserved.

7 References 814

7.1 Normative 815

816	[XML1.0 3 rd Edition	[XML1.0 3 rd Edition]			
817	-	Tim Bray, et al., <i>Extensible Markup Language (XML) 1.0 (Third Edition)</i> ,			
818		W3C Recommendation, February 2004, http://www.w3.org/TR/REC-xml			
819					
820	[XML Schema Par	t 1]			
821		Henry S. Thompson, et al. XML Schema Part 1: Structures, W3C			
822		Recommendation, May 2001, http://www.w3.org/TR/xmlschema-1/			
823					
824	[XML Schema Par	t 2]			
825	-	Paul V. Biron, et al. XML Schema Part 2: Datatypes, W3C			
826		Recommendation, May 2001, http://www.w3.org/TR/xmlschema-2/			
827					
828	[XNS]	Tim Bray, et al., Extensible Namespaces in XML, W3C			
829	[]	Recommendation, January 1999, http://www.w3.org/TR/REC-xml-			
830		names/			
831					
832	[WSDL]	Erik Christensen, et al., Web services Description Language (WSDL)			
833		1.1, W3C Note, March 2001, http://www.w3.org/TR/wsdl			
834					
835	[WS-Addressing]	Don Box, et al., Web services Addressing (WS-Addressing), W3C			
836		Member Submission, August 2004,			
837		http://www.w3.org/Submission/2004/SUBM-ws-addressing-20040810/			
838					
839	[RFC2119]	S. Bradner, Key words for use in RFCs to Indicate Requirement Levels,			
840		http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.			
841					
842	[RFC2396bis]	T. Berners-Lee, et al., Uniform Resource Identifier (URI): Generic			
843		Syntax, IETF RFC 2396bis-04, February 2004,			
844		http://www.ietf.org/internet-drafts/draft-fielding-uri-rfc2396bis-04.txt			
845	7.2 Non-norma	tive			
0.40	[MOW6]	Lear Cadulphia Web Convises Distributed Management Management of			
846 847	[MOWS]	Igor Sedukhin, Web Services Distributed Management:Management of Web Services (WSDM-MOWS) 1.0, OASIS Committee Draft, December			
848		2004, http://docs.oasis-open.org/wsdm/2004/12/wsdm-mows-1.0.pdf			
849					
850	[MUWS Part 2]	William Vambenepe, Web Services Distributed Management:			
851		Management using Web Services (MUWS 1.0) Part 2, OASIS Committee			
852		Draft, December 2004, http://docs.oasis-open.org/wsdm/2004/12/wsdm-			
853		muws-part2-1.0.pdf			
854					
855	[MUWS REQS]	Pankaj Kumar, et al., Requirements – Management Using Web Services,			
856	[Committee Draft, October 2003, http://www.oasis-			
857		open.org/apps/org/workgroup/wsdm/download.php/6185/WSDM-MUWS-			
858		Req-committee-draft-1.0-20031002.pdf			

859 860 861 862	[SOAP]	Don Box, et al., <i>Simple Object Access Protocol (SOAP) 1.1</i> , W3C Note, May 2000, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
863 864 865 866	[WSA]	David Booth, et al. <i>Web Services Architecture</i> , W3C Working Group Note, February 2004, http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/
867 868 869 870	[wss]	Anthony Nadalin, et al. <i>Web Services Security: SOAP Message Security</i> 1.0, OASIS Standard, March 2004, http://docs.oasis- open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf
871 872 873	[BP]	Keith Ballinger, et al. <i>Basic Profile Version 1.1</i> , WS-I Final Material, August 2004, http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08- 24.html

874 Appendix A. Acknowledgements

The following individuals were members of the committee when the specification was approved by the technical committee

877 Guru Bhat, Jeff Bohren, Winston Bumpus, Nick Butler, Brian Carroll, Fred Carter, Michael

878 Clements, David Cox, John DeCarlo, Andreas Dharmawan, Mark Ellison, John Fuller, Paul

Lipton, Heather Kreger, Hal Lockhart, Frederico Maciel, Tom Maguire, Bryan Murray, Richard

880 Nikula, Mark Peel, Richard Pelavin, Homayoun Pourheidari, Warren Roberts, Karl Schopmeyer,

881 Igor Sedukhin, David Snelling, Thomas Studwell, William Vambenepe, Andrea Westerinen, Jim

- 882 Willits, Zhili Zhang.
- 883 In addition, the following non-member employees of member companies made contribution to the 884 specification: Maryann Hondo, Ian Springer.

Appendix B. Notices

886 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 887 888 document or the extent to which any license under such rights might or might not be available: 889 neither does it represent that it has made any effort to identify any such rights. Information on 890 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 891 892 to be made available, or the result of an attempt made to obtain a general license or permission 893 for the use of such proprietary rights by implementers or users of this specification, can be

- 894 obtained from the OASIS Executive Director.
- 895 OASIS invites any interested party to bring to its attention any copyrights, patents or patent 896 applications, or other proprietary rights which may cover technology that may be required to 897 implement this specification. Please address the information to the OASIS Executive Director.
- 898 Copyright © OASIS Open 2003-2005. 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,
- 901 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
- 904 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
- 907 than English.
- 908 The limited permissions granted above are perpetual and will not be revoked by OASIS or its 909 successors or assigns.
- 910 This document and the information contained herein is provided on an "AS IS" basis and OASIS
- 911 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO
- 912 ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE
- 913 ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
- 914 PARTICULAR PURPOSE.

Appendix C. MUWS Part 1 Schema (Normative) 915 916 <?xml version="1.0" encoding="utf-8"?> 917 <xs:schema 918 targetNamespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-919 part1.xsd" 920 xmlns:muws-pl-xs="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-921 part1.xsd" 922 xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" 923 xmlns:xs="http://www.w3.org/2001/XMLSchema" 924 elementFormDefault="qualified" attributeFormDefault="unqualified"> 925 926 <xs:import namespace="http://schemas.xmlsoap.org/ws/2004/08/addressing"</pre> 927 928 schemaLocation="http://schemas.xmlsoap.org/ws/2004/08/addressing"/> 929 930 <xs:element name="ResourceId" type="xs:anyURI"/> 931 932 <xs:complexType name="IdentityPropertiesType"> 933 <xs:sequence> 934 <xs:element ref="muws-pl-xs:ResourceId"/> 935 </xs:sequence> 936 </xs:complexType> 937 938 <xs:element name="IdentityProperties"</pre> 939 type="muws-pl-xs:IdentityPropertiesType"/> 940 941 942 <xs:element name="ManageabilityCapability" type="xs:anyURI"/> 943 944 <xs:complexType name="ManageabilityCharacteristicsPropertiesType"> 945 <xs:sequence> 946 <xs:element ref="muws-pl-xs:ManageabilityCapability"</pre> 947 minOccurs="0" maxOccurs="unbounded"/> 948 </xs:sequence> 949 </xs:complexType> 950 951 <xs:element name="ManageabilityCharacteristicsProperties"</pre> 952 type="muws-pl-xs:ManageabilityCharacteristicsPropertiesType"/> 953 954 955 <xs:complexType name="CorrelatablePropertiesType"> 956 <xs:sequence> 957 <xs:any minOccurs="0" maxOccurs="unbounded" 958 namespace="##other" processContents="lax"/> 959 </xs:sequence> 960 <xs:attribute name="Dialect" type="xs:anyURI"/> 961 <xs:attribute name="NegativeAssertionPossible" type="xs:boolean"/> 962 <xs:anyAttribute namespace="##other"/> 963 </xs:complexType> 964 965 <xs:element name="CorrelatableProperties"</pre> 966 type="muws-pl-xs:CorrelatablePropertiesType"/> 967 968 <xs:complexType name="CorrelatablePropertiesPropertiesType"> 969 <xs:sequence> 970 <xs:element ref="muws-pl-xs:CorrelatableProperties"</pre> 971 minOccurs="0" maxOccurs="unbounded"/> 972 </xs:sequence> 973 </xs:complexType> 974

wsdm-muws-part1-1.0 Copyright © OASIS Open 2003-2005. All Rights Reserved.

```
975
         <xs:element name="CorrelatablePropertiesProperties"</pre>
 976
                      type="muws-pl-xs:CorrelatablePropertiesPropertiesType"/>
 977
 978
 979
         <xs:complexType name="ComponentAddressType">
 980
           <xs:sequence>
 981
              <xs:any namespace="##any" processContents="lax"/>
 982
           </xs:sequence>
 983
         </xs:complexType>
 984
 985
         <xs:complexType name="ComponentType">
 986
           <xs:sequence>
 987
              <xs:element name="ResourceId" type="xs:anyURI"</pre>
 988
                          minOccurs="0"/>
 989
              <xs:element name="ComponentAddress"</pre>
 990
                          type="muws-pl-xs:ComponentAddressType"
 991
                          minOccurs="0" maxOccurs="unbounded"/>
 992
              <xs:any minOccurs="0" maxOccurs="unbounded"
 993
                      namespace="##other" processContents="lax"/>
 994
            </xs:sequence>
 995
            <xs:anyAttribute namespace="##other"/>
 996
          </xs:complexType>
 997
 998
         <xs:complexType name="ManagementEventType">
 999
            <xs:sequence>
1000
              <xs:element name="EventId" type="xs:anyURI"/>
1001
              <xs:element name="SourceComponent" type="muws-p1-xs:ComponentType"/>
1002
              <xs:element name="ReporterComponent" type="muws-pl-xs:ComponentType"</pre>
1003
                          minOccurs="0"/>
1004
              <xs:any minOccurs="0" maxOccurs="unbounded"
1005
                      namespace="##other" processContents="lax"/>
1006
            </xs:sequence>
1007
            <xs:attribute name="ReportTime" type="xs:dateTime" use="optional"/>
1008
            <xs:anyAttribute namespace="##other"/>
1009
         </xs:complexType>
1010
1011
         <xs:element name="ManagementEvent"</pre>
1012
                      type="muws-pl-xs:ManagementEventType"/>
1013
1014
          <xs:element name="ManageabilityEndpointReference"</pre>
1015
                      type="wsa:EndpointReferenceType"/>
1016
1017
       </xs:schema>
1018
```

Appendix D. Properties Boolean Match Schema (Normative)

```
<?xml version="1.0" encoding="utf-8"?>
1021
1022
       <xs:schema
1023
           targetNamespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-pbm.xsd"
1024
           xmlns:pbm="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-pbm.xsd"
1025
           xmlns:xs="http://www.w3.org/2001/XMLSchema"
1026
           elementFormDefault="qualified" attributeFormDefault="unqualified">
1027
1028
         <xs:element name="Match" type="xs:QName"/>
1029
1030
         <xs:complexType name="MatchAllType">
1031
           <xs:choice>
1032
             <xs:element ref="pbm:Match"/>
1033
             <xs:element ref="pbm:MatchAny"/>
1034
           </xs:choice>
1035
         </xs:complexType>
1036
1037
         <xs:complexType name="MatchAnyType">
1038
           <xs:choice>
1039
             <xs:element ref="pbm:Match"/>
1040
             <xs:element ref="pbm:MatchAll"/>
1041
           </xs:choice>
1042
         </xs:complexType>
1043
1044
         <xs:element name="MatchAll" type="pbm:MatchAllType"/>
1045
1046
         <xs:element name="MatchAny" type="pbm:MatchAnyType"/>
1047
1048
       </xs:schema>
1049
1050
```