

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

1 **OPENADE 1.0 SERVICE DEFINITION - WEB SERVICES EXTENSION**

2 *VERSION: DRAFT V0.6*

3 Release Date: 4/22/2010

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

4 **Acknowledgements**

5 The following individuals and their companies have contributed and/or provided support to the work of
6 the OpenADE 1.0 Service Definition - Web Services Extension:

- 7 • Chad Maglaque from Microsoft
- 8 • Charles Spirakis from Google
- 9 • Dave Mollerstuen from Tendril Networks
- 10 • Gerald Gray from CIMple Integrations
- 11 • Mark Ortiz from Consumers Energy
- 12 • Shawn Hu from Xtensible Solutions / SCE /Consumers Energy
- 13 • Steve Van Ausdall from Xtensible Solutions / SCE

14 The OpenADE Task Force wishes to thank all of the contributors to OpenADE, especially the above-
15 mentioned individuals and their companies for their support of this important endeavor, as it sets a key
16 foundation for an interoperable Smart Grid.

17

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

18 **Document History**

19 **Revision History**

20 Date of this revision: Apr. 15, 2010

Revision Number	Revision Date	Revision By	Summary of Changes	Changes marked
0.1	4/8/10	Gerald R. Gray	Initial draft discussion version.	N
0.2	4/14/10	Gerald R. Gray	Added example wsdl and xsds provided by Shawn Hu; example SOAP envelope structure	N
0.3	4/15/10	Steve Van Ausdall	Additional cleanup and updates	N
0.4	4/20/10	Steve Van Ausdall	Changes from reviews with SD team	N
0.5	4/20/10	Gerald R. Gray	Added reference to previous AMI-ENT work; additional clean-up from team discussion	N
0.6	4/22/10	Shawn Hu & Mark Ortiz	Added detailed WSDL information	N

21 **Open Issues Log**

22 Last updated: Apr. 8, 2010

Issue	Issue Date	Provided By	Summary of the Issue

23

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

Contents

24			
25	1	Introduction	5
26	1.1	Rights / Management / Governance	5
27	1.1.1	Intellectual Property Rights	5
28	1.1.2	CIM Object Models	5
29	1.1.3	Web Service Definitions	6
30	1.2	Referenced Specifications	6
31	1.3	Referenced Guidance	6
32	1.4	Namespaces	6
33	2	Web Services	7
34	1.1	Service Structure	7
35	1.2	Service Naming Convention	7
36	1.3	SOAP Binding	8
37	3	Versioning	10
38	4	Functional Areas	10
39	4.1	Common	10
40	4.1.1	Discover Services - (Sequence diagram)	11
41	4.2	Metering Consumption	12
42	4.2.1	Consumption Receive - (Sequence diagram)	12
43	5	Service Definitions	12
44	5.1	Web Service Details	13
45	5.1.1	Notification (ReceiveConsumptionUsage)	13
46	6	Appendix	13
47	6.1	Consumption WSDL/XSD	13
48	6.2	Large Size Data Exchange	14
49	6.3	Service Discovery	14

50

51

List of Figures

53	Figure 1: Discover Service Resources Sequence Diagram	11
54	Figure 2: ReceiveNotification Sequence Diagram	12
55	Figure 3: Web Service Interface	12

56

List of Tables

58	Table 1: Service Operations.....	13
----	----------------------------------	----

59

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - Web Services Extension

60

61 1 INTRODUCTION

62 This document contains only the extensions necessary to the OpenADE Common specification to build a WS-I Basic
63 Profile 1.1 implementation of the OpenADE Requirements Specification. The "OpenSG OpenADE SD – Common"
64 document should be thought of as the parent of this document, filling in sections not addressed in the Common
65 specification.

66 These extensions define a collection of services as a discoverable data service, using SOAP over HTTPS to send and
67 receive requests and information in XML. This architecture provides secure access to scalable methods and data
68 resources hosted by the provider, while maintaining concurrency and integrity. All data is secured at the user level,
69 so that access to individual user data can be provided or revoked to external services, and other users' data will
70 still be protected.

71 1.1 RIGHTS / MANAGEMENT / GOVERNANCE

72 1.1.1 INTELLECTUAL PROPERTY RIGHTS

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

77 UCAIug requests any party that believes it has a patent claim that would necessarily be infringed by
78 implementations of this UCAIug work, to notify UCAIug immediately, so that fair and reasonable licensing terms
79 can be negotiated. UCAIug invites any party aware of applicable undisclosed patent claims to contact the UCAIug.
80 UCAIug may include such claims on its website, but disclaims any obligation to do so.

81 UCAIug takes no position regarding the validity or scope of any intellectual property or other rights that might be
82 claimed to pertain to the implementation or use of the technology described in this document or the extent to
83 which any license under such rights might or might not be available; neither does it represent that it has made any
84 effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of
85 licenses to be made available, or the result of an attempt made to obtain a general license or permission for the
86 use of such proprietary rights by implementers or users of this UCAIug recommendation, can be obtained from the
87 UCAIug. UCAIug makes no representation that any information or list of intellectual property rights will at any time
88 be complete, or that any claims in such list are, in fact, Essential Claims.

89 1.1.2 CIM OBJECT MODELS

90 The recommendations herein build on work owned by the IEC. Required extensions identified in this
91 recommendation will be submitted to the IEC, and will be tracked for inclusion in the model.

92 Information on the management of rights and governance can be found at the page below.
93 <http://www.iec.ch/tctools/patent-guidelines.htm>

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - Web Services Extension

94 1.1.3 WEB SERVICE DEFINITIONS

95 If necessary, UCAIug is willing to work with standards development organizations to incorporate additional aspects
96 of this recommendation into standards, including the specification of how to use profiled (restricted) CIM objects
97 within the SOAP over HTTP environment, and possibly the web service definitions themselves.

98 1.2 REFERENCED SPECIFICATIONS

- 99 • [1] IEC CIM (TC 57 61968/61970) - <http://tc57.iec.ch>
- 100 • [2] OAuth - <http://oauth.net/>
- 101 • [3] WS Basic Profile Version 1.0 <http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html>
- 102 • [4] OpenSG OpenADE SD – Common
- 103 • [5] UDDI: www.uddi.org
- 104 • [6] SOAP: <http://www.w3.org/TR/SOAP/>
- 105 • [7] IEC TC57 WG14 61968-1-2 – Profile for use of CIM with WS-I Basic Profile

106 1.3 REFERENCED GUIDANCE

- 107 • [G1] OpenADE B&UR 1.0 -
108 [http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2fsgsystems%2fOpenADE%2fShared%20Documents%2fBusiness%20and%20User%20Requirements)
109 [2fsgsystems%2fOpenADE%2fShared%20Documents%2fSRS](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2fsgsystems%2fOpenADE%2fShared%20Documents%2fSRS)
- 110 • [G2] OpenADE SRS 1.0 -
111 [http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
112 [2futilisec%2f](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
113 [hared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
- 114 • [G3] 3PDA – Security Profile for Third Party Data Access (ASAP-SG)
115 [http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
116 [2futilisec%2f](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
117 [hared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile](http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile)
- 118 • [G4] Service Definitions Technical Guide [http://www.smartgridipedia.org/images/a/af/AMI_ENT_Step-By-](http://www.smartgridipedia.org/images/a/af/AMI_ENT_Step-By-Step_Modeling_and_Artifacts_Generation_Guidelines.doc)
119 [Step_Modeling_and_Artifacts_Generation_Guidelines.doc](http://www.smartgridipedia.org/images/a/af/AMI_ENT_Step-By-Step_Modeling_and_Artifacts_Generation_Guidelines.doc)

118 1.4 NAMESPACES

119 The subject of namespaces is important, because the namespace identifies the domain managing the definitions of
120 protocol resources and formats. OpenSG proposes to use a temporary namespace until the final destination is
121 identified. In any case, namespaces already defined elsewhere and used directly within reference service
122 definitions will remain where they are, and will reference the identified body.

123 The proposed temporary namespace for definitions to be used in early implementations is below. (Service
124 definition will be updated with the final approved namespaces.)

125 `http://osgug.ucaiug.org/ns/2010/06/wsoade`

UCAIug OpenSG OpenADE Task Force

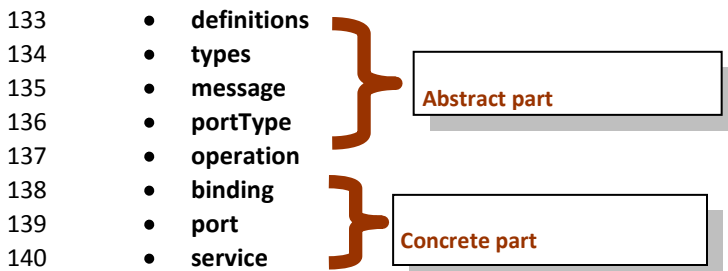
OpenADE 1.0 Service Definition - Web Services Extension

2 WEB SERVICES

The purpose of the section is to provide a set of recommendations for a Web Service definition based on OpenSG's Service Definition design patterns and Web Services Description Language (WSDL) from W3C. The audiences of the document are assumed to have basic knowledge of Web service and XML schema.

1.1 SERVICE STRUCTURE

W3C WSDL (v1.1) is followed to define OpenADE Web services. The services are made of two parts with following tags.



141

The web service design practices are summarized below:

- Standard SOAP envelope is used to avoid extra message enveloping.
- XSD as data type is imported instead of being embedded for better version control
- Wire signature issue is avoided by redefining element names such as CreatedConsumption and ChangedConsumption using a single XSD Consumption complexType
- Wrapped document WSDL style is used
- Operation name follows the Verb + Noun naming convention which can potentially avoid contend-based routing

150

1.2 SERVICE NAMING CONVENTION

Interfaces are defined using a specific set of verbs and nouns using Web service technology. Each service then has a subset of operations that are associated with information objects. Each operation is named following IEC 61968-1 verb + noun (information objects). The detail service and operation naming convention is covered in OpenSG Service Definition Technical Guide [G4]. In summary, the naming conventions are:

- **Service name:**

To follow *<Service pattern name>+<Information Object>* such as ReceiveConsumption

- **Operation name:**

To follow *<Operation pattern name>+<Information Object>* such as CreatedConsumption

160

UCAIug OpenSG OpenADE Task Force

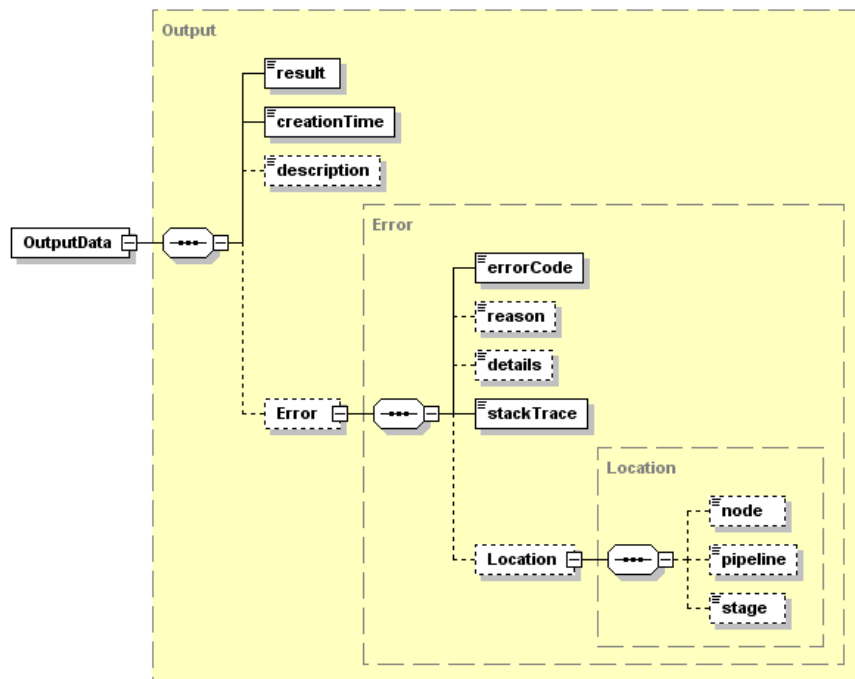
OpenADE 1.0 Service Definition - Web Services Extension

161

162 1.3 SOAP BINDING

163 The document style using SOAP body is the most common practice in WSDL design. It can fully utilize the benefits
164 of an XML schema for payload validation.

165 Both <soap:binding> and <soap:operation> styles are defined as “document”. Also <soap:body> is used for both
166 input and output operations. Input data type is typically a payload such as Consumption data definition. Output
167 data follows a common XSD (OutputData.xsd) that is included for each operation in a WSDL. Each operation’s
168 OutputData adheres to the following XSD structure and is used as an acknowledgement return or a fault return
169 during a synchronous call.



170

171 The *wSDL:operation* is named the same as the input element name. As a result the WSDL is a wrapped document
172 style WSDL. Wrapped document style originates from Microsoft to mimic a RPC style. In a RPC style, an XML
173 payload is wrapped by its operation name.

174 Here is the WSDL section that illustrates the wrapped document style. Note the element name is the same as the
175 operation name (CreatedConsumption):

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - Web Services Extension

```
... ..  
  
<wsdl:message name="CreatedConsumptionInput">  
  <wsdl:part name="CreatedConsumption" element="typeIn:CreatedConsumption"/>  
</wsdl:message>  
  
... ..  
  
<wsdl:portType name="Consumption">  
  <wsdl:operation name="CreatedConsumption">  
    <wsdl:documentation>CreatedConsumption</wsdl:documentation>  
    <wsdl:input name="CreatedConsumptionInput" message="tns:CreatedConsumptionInput"/>  
    <wsdl:output name="CreatedConsumptionOutput" message="tns:outputData"/>  
    <wsdl:fault name="faultInfor" message="tns:faultReturn"/>  
  </wsdl:operation>  
</wsdl:portType>
```

176

177 One issue with the wrapped document style is when adding an “operation” like element in an XSD that may break
178 semantics in data definition. There can be also maintenance issue in a case of a new operation being added which
179 causes not only WSDL change but also XSD update. Therefore the recommendation is to create the operation like
180 elements within WSDL and decouple the original XSD element. Here is an example.

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - Web Services Extension

```
<wsdl:types>
  <xs:schema targetNamespace="http://<namespace prefix>/2009/09/ConsumptionIn">
    <xs:import namespace="http://<namespace prefix>/2009/09/Consumption" schemaLocation="Consumption.xsd"/>
    <xs:element name="CreatedConsumption" type="typeOrig:Consumption"/>
    <xs:element name="ChangedConsumption" type="typeOrig:Consumption"/>
  </xs:schema>
  ... ..
</wsdl:types>
```

181

182 Note that the operation-like element name is defined within wsdl:types section. This element references a
183 complexType within Consumption.xsd which does not need a change for this style.

184 3 VERSIONING

185 Versioning will be handled in the manner specified in the OpenADE Common document.

186 Additionally, WSDL targetNamespace needs to be updated whenever a change occurs to an XSD namespace. In
187 other words, a major XSD update will result in a WSDL namespace change and minor XSD update (no namespace
188 change) will have no impact on WSDL namespace.

189 4 FUNCTIONAL AREAS

190 4.1 COMMON

191 The flows in this section represent general-purpose functions that are needed for all protected resource
192 publications, in addition to those specified in [4] OpenADE 1.0 SD - Common.

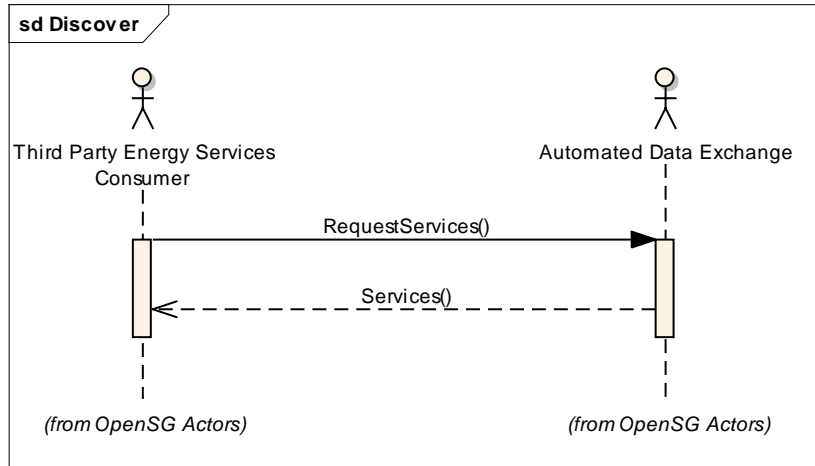
UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - Web Services Extension

193 4.1.1.1 DISCOVER SERVICES - (SEQUENCE DIAGRAM)

194 This flow shows retrieval of service definitions from UDDI. Full details are not specified in this document. This
195 capability is purely optional in this release.

196 Addresses OpenSG OpenADE 1.0 SRS 3.2.1, bullet 1.2, 1.3



197

198

Figure 1: Discover Service Resources Sequence Diagram

UCAIug OpenSG OpenADE Task Force

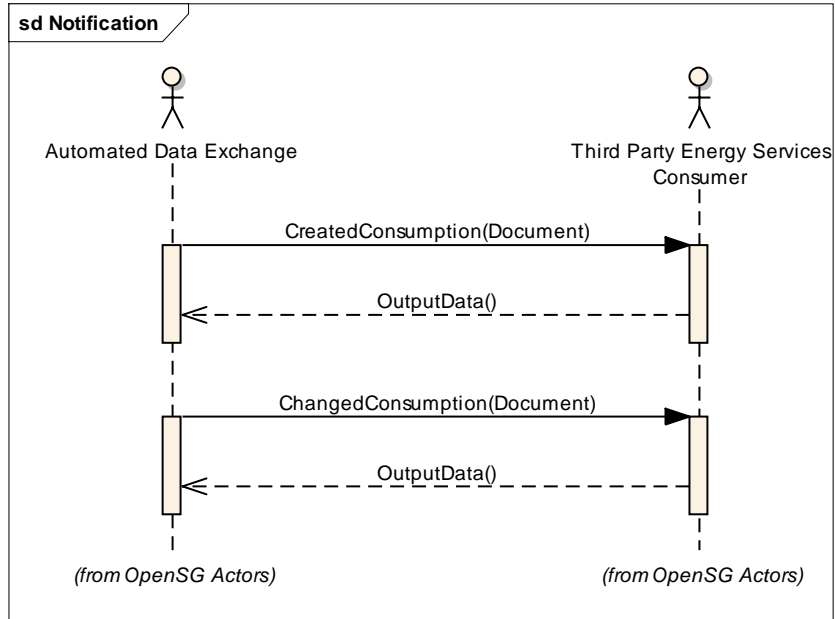
OpenADE 1.0 Service Definition - Web Services Extension

199 4.2 METERING CONSUMPTION

200 4.2.1 CONSUMPTION RECEIVE - (SEQUENCE DIAGRAM)

201 This is the flow used to transfer authorized data from the ADE system to the 3rd party via the notification delivery
 202 service operation. This operation can be called with varying frequency, and each transfer shall include all new and
 203 updated authorized resources since the last transfer.

204 Addresses OpenSG OpenADE 1.0 SRS 3.2.1, bullet 3.1



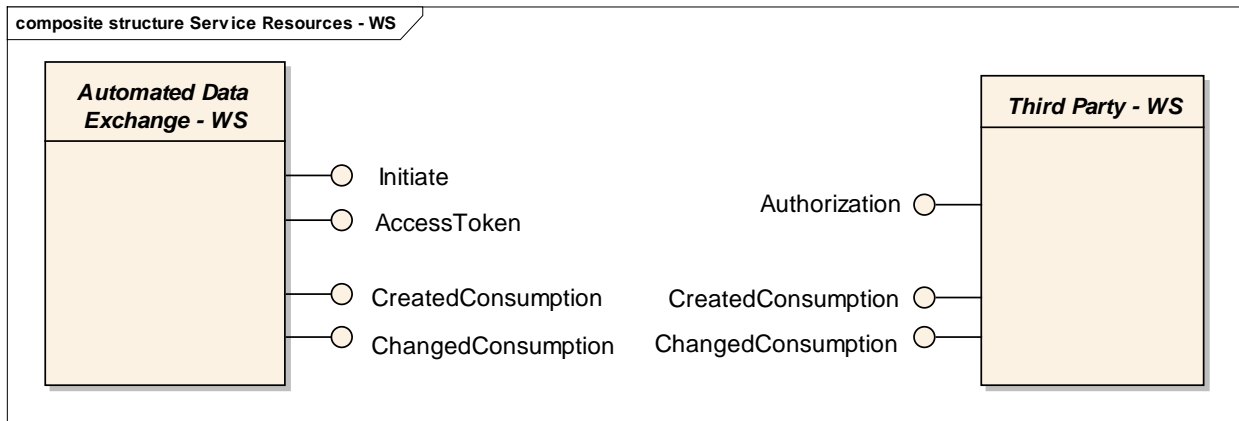
205

206

Figure 2: ReceiveNotification Sequence Diagram

207 5 SERVICE DEFINITIONS

208 The following diagram provides a view of the service operations defined.



209

210

Figure 3: Web Service Interface

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

211 The following table lists the resources defined for OpenADE.

Service Name	Service Operation	Service Provider / Data Consumer	Service Consumer / Data Provider	Use
ReceiveConsumptionUsage	CreatedConsumption	3 rd Party	ADE	Send new data objects in batch document
ReceiveConsumptionUsage	ChangedConsumption	3 rd Party	ADE	Send updated data objects
ReceiveConsumptionUsage	CreatedConsumption	ADE	3 rd Party	Send new data objects in batch document (normally not used)
ReceiveConsumptionUsage	ChangedConsumption	ADE	3 rd Party	Send updated data objects (primarily for termination)

212 Table 1: Service Operations

213 **5.1 WEB SERVICE DETAILS**

214 Examples of the consumption wsdl, and associated consumption and output data xsds can be found in the
 215 Appendix.

216 **5.1.1 NOTIFICATION (RECEIVECONSUMPTIONUSAGE)**

217 These operations are to be used to transmit the creation or modification of resources.

Operation	Use	Element
CreatedConsumption	Input	Document
CreatedConsumption	Output	OutputData
ChangedConsumption	Input	Document
ChangedConsumption	Output	OutputData

218

219 **6 APPENDIX**

220 **6.1 CONSUMPTION WSDL/XSD**

221 Note that the schema in this example is the schema contained in [4] "OpenADE 1.0 SD - Common" as it is finalized.



222 **OpenADE SD WS.zip**

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - Web Services Extension

223 **6.2 LARGE SIZE DATA EXCHANGE**

224 It is recommended to use MTOM for large data transaction. MTOM stands for Message Transmission Optimization
225 Mechanism. It is often used for a binary data transaction and usually used with XOP (XML-binary Optimized
226 Packaging). Using MTOM, the SOAP binding has no significant change in comparison with the conventional SOAP
227 binding in document style. Currently there is no requirement on a large size payload data transaction. Should this
228 be a case in the future, a new operation based on MTOM will be provided.

229 **6.3 SERVICE DISCOVERY**

230 Universal Description, Discovery, and Integration (UDDI) is a specification designed to allow businesses to enter
231 details about themselves and the services they provide in a registry. Searches can be typically be performed by
232 company name, specific service, or types of service. This allows companies providing or needing web services to
233 discover each other, define how they interact over the Internet, and share information in a standardized fashion.

234 Since a WSDL defines the XML grammar for describing services as collections of communication endpoints capable
235 of exchanging messages, utilities and third parties can publish WSDLs for services they provide and links to the
236 WSDLs are usually offered in a company's profile in a UDDI registry.

237