

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

1 **OPENADE 1.0 SERVICE DEFINITION - REST EXTENSION**

2 *VERSION: DRAFT V0.93*

3 Release Date: 8/5/2010

1 **Acknowledgements**

2 The following individuals and their companies have contributed and/or provided support to the work of
3 the OpenADE 1.0 Service Definition - REST Extension:

- 4 • Charles Spirakis from Google
- 5 • Dave Mollerstuen from Tendril Networks
- 6 • Gerald Gray from CIMple Integrations
- 7 • Jeffrey Kenward from DTE Energy
- 8 • Jeremy McDonald from SCE
- 9 • Mark Ortiz from Consumers Energy
- 10 • Shawn Hu from Xtensible Solutions / SCE
- 11 • Steve Van Ausdall from Xtensible Solutions / SCE

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

15

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

16 **Document History**

17 **Revision History**

18 Date of this revision: July, 28, 2010

Revision Number	Revision Date	Revision By	Summary of Changes	Changes marked
0.5	2/25/10	Steve Van Ausdall	Initial draft discussion version.	N
0.6	3/1/10	Steve Van Ausdall	Additional details about defined resources	N
0.8	4/8/10	Steve Van Ausdall	Broke REST parts out of Common doc	N
0.85	4/15/10	Steve Van Ausdall	Changes from first review meeting	N
0.9	4/22/10	Steve Van Ausdall	Updates from commenters	Y
0.92	7/28/10	Wayne Dennison Steve Van Ausdall	Additional Cleanup and Updates from F2F meeting and Review	N
0.93	8/6/10	Steve Van Ausdall	Addition of authorization issue	Y

19 **Open Issues Log**

20 Last updated: Aug. 6, 2010

Issue	Issue Date	Provided By	Summary of the Issue
50	8/5/10	Steve Van Ausdall	OpenADE SD - General-purpose authorization mechanism

21

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

Contents

22			
23	1	Introduction	6
24	1.1	Rights / Management / Governance	6
25	1.1.1	Intellectual Property Rights	6
26	1.1.2	CIM Object Models	6
27	1.1.3	Service Resource Definitions	7
28	1.2	Referenced Specifications	7
29	1.3	Referenced Guidance	7
30	1.4	Namespaces	7
31	2	Resources	7
32	2.1	URI Format / syntax	8
33	2.2	Message document format	8
34	2.3	Payload entities	8
35	2.3.1	Resources	9
36	3	Patterns	10
37	3.1	Creating, Updating, Deleting	10
38	3.2	Query, request and response (Retrieve) formats	10
39	3.2.1	Format	11
40	3.2.2	Category	11
41	3.2.3	Reference Expansion	11
42	3.2.4	Sorting	11
43	3.2.5	Filtering	11
44	3.2.6	Iteration	11
45	3.2.7	Conditional Retrieval	12
46	3.3	Event Notification (pub/sub)	12
47	3.4	Batch transfers	12
48	4	Discovery	12
49	5	Metadata	12
50	6	Extensibility	12
51	7	Versioning	12
52	8	Concurrency	13
53	9	Resource Definitions	13
54	9.1	Provider (Utility) Resources	13
55	9.2	Consumer (3 rd Party) Resources	14
56	9.3	Resource Definition	14
57	9.4	Resource Details	14

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

58	9.4.1	Collection (Feed)	14
59	9.4.2	Authorization	15

60

61

62

63

List of Tables

64	Table 1: Provider Resource Operations	13
----	---	----

65	Table 2: Consumer Resource Operations	14
----	---	----

66

67

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

68

69 1 INTRODUCTION

70 This document contains only the extensions necessary to the OpenADE Common specification to build an AtomPub
71 resource representation syndication server. It is based heavily on GData, an open specification of AtomPub
72 extensions required for general-purpose data synchronization.

73 These extensions define a collection of resource feeds as a discoverable, stateless data service, using HTTPS to
74 send and receive requests and information in AtomPub XML. This resource-oriented architecture is proposed,
75 similar to efforts elsewhere, such as web / internet of things, GData, and OData. This architecture provides secure
76 access to scalable methods and data resources hosted by the provider, while maintaining concurrency and
77 integrity. All data is secured at the user level, so that access to individual operations can be provided or revoked to
78 external services, and other users' data will still be protected.

79 1.1 RIGHTS / MANAGEMENT / GOVERNANCE

80 1.1.1 INTELLECTUAL PROPERTY RIGHTS

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

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

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

97 1.1.2 CIM OBJECT MODELS

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

100 Information on the management of rights and governance can be found at the page below.

101 <http://www.iec.ch/tctools/patent-guidelines.htm>

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

1.1.3 SERVICE RESOURCE DEFINITIONS

If necessary, UCAIug is willing to work with standards development organizations to incorporate additional aspects of this recommendation into standards, including the specification of how to use profiled (restricted) CIM objects within a RESTful HTTP environment, and possibly the resource definitions themselves.

1.2 REFERENCED SPECIFICATIONS

- [1] OpenADE B&UR 1.0 - <http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2fsgsystems%2fOpenADE%2fShared%20Documents%2fBusiness%20and%20User%20Requirements>
- [2] OpenADE SRS 1.0 - <http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2fsgsystems%2fOpenADE%2fShared%20Documents%2fSRS>
- [3] IEC CIM (TC 57 61968/61970) - <http://tc57.iec.ch>
- [4] OAuth - <http://tools.ietf.org/html/draft-hammer-oauth-10>
- [5] HTTP – IETF RFC 2616 - <http://www.ietf.org/rfc/rfc2616.txt>
- [8] PubSubHubbub - <http://code.google.com/p/pubsubhubbub/>
- [9] Atom Publishing Protocol (RFC 5023) – <http://tools.ietf.org/html/rfc5023>
- [10] OpenSG OpenADE SD – Common <http://osgug.ucaiug.org/sgsystems/OpenADE/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2fsgsystems%2fOpenADE%2fShared%20Documents%2fService%20Definition%2fOpenADE%201%2e0%20Service%20Definition>

1.3 REFERENCED GUIDANCE

- [G1] 3PDA – Security Profile for Third Party Data Access (ASAP-SG) <http://osgug.ucaiug.org/utilisec/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2futilisec%2fhared%20Documents%2fThird%20Party%20Data%20Access%20Security%20Profile>
- [G2] GData - <http://code.google.com/apis/gdata/docs/2.0/reference.html>
- [G3] OData - [http://www.odata.org/docs/\[MC-APDSU\].htm](http://www.odata.org/docs/[MC-APDSU].htm)

1.4 NAMESPACES

This document does not define any namespaces. Namespaces defined in referenced specifications shall be used as defined.

2 RESOURCES

Resource Oriented Architecture is nothing new; in fact the web we are all familiar with today provides restful (browser) access to internet resources. When you specify the Address URL of a page, you are providing the address of that resource that you requested. This architecture provides similar operations for external data consumer applications to request data and methods. Generally, data is made available as a feed, which is an agreement about how to query, create, update, request, and delete entries (individual object records, which have a defined schema according to their type).

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

139 Since this document is the first to define the general-purpose conventions, several resources were identified to
140 allow consumers to gain access to the resources they want. These extensions are listed below.

- 141 • **Service** – To verify service status
- 142 • **Enrollment** – To register for access and notifications
- 143 • **ActivityRecord** – To provide information about an activity (request)

144 In addition, addressable resources are defined for the following objects, defined in [10] OpenADE SD - Common.

- 145 • **Customer**
- 146 • **CustomerAgreement**
- 147 • **ServiceDeliveryPoint**
- 148 • **MeterReading**
- 149 • **IntervalReading**
- 150 • **ReadingType**

151

152 2.1 URI FORMAT / SYNTAX

153 The URIs of the resources defined in OpenADE take the general form below. The <baseUrl> does not need to be
154 the same across different implementations, since resource addresses include the entire string.

155 `http://<baseUrl>/<resource>?<query>`

156 Resource requests require inputs of user, key, and resource object. Additional path elements may be required for
157 some resources. Resources return a list (feed) or an individual entry.

158 Requests for protected resources require https, and require authorization token in HTTP header.

159 The query section contains additional inputs that can be specified to affect processing, passed as a list of
160 `name=value` pairs.

161 2.2 MESSAGE DOCUMENT FORMAT

162 Message documents shall use the extended version of the Atom Publishing Protocol ([9] AtomPub, which extends
163 the Atom Syndication Protocol) defined in [G2] GData for the elements described in this document, to fulfill this
164 need.

165 In addition to the recommended format, it is possible to support additional representations. An input can be
166 accepted to provide RSS or other formats, but these additional formats are all optional, and will only be considered
167 for this specification if needed.

168 2.3 PAYLOAD ENTITIES

169 Payload entities will be specializations (subclasses) of the message document “entry”. They will therefore inherit
170 all elements defined in the message document entry, as well as implement an XML schema representation of a
171 CIM class defining additional schema elements.

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

172 Feed payloads will contain a list of references to resource entries that match the request query criteria.

173 The batch payload defined in the Common document allows a number of documents or resource requests to be
174 included in a single request.

175 2.3.1 RESOURCES

176 Domain data objects build on the IEC CIM model. In general, resources will be named using the CIM class as the
177 resource part of the URI. Collections are returned if no specific entry ID is specified in the resource path. For
178 listings of fields, see the details for each resource, defined in Common [10].

179 Examples are shown below. Note that these are separate examples of a GET request issued to the resource path in
180 the comment before the example result.

181 Customer

```
182 <!-- /c/12345678910 -->
183 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
184   <id>12345678910</id>
185   <m:CustomerAgreement href="https://{IP Address}/c/12345678910/ca"/>
186 </entry>
```

187 CustomerAccount

```
189 <!-- /c/12345678910/ca/56421587 -->
190 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
191   <id>56421587</id>
192   <m:CustomerAuthorisation>
193     <m:validityInterval>
194       <m:end>2011-12-17T00:00:00Z</m:end>
195       <m:start>2010-12-17T00:00:00Z</m:start>
196     </m:validityInterval>
197   </m:CustomerAuthorisation>
198   <m:ServiceDeliveryPoint href="https://{IP Address}/c/12345678910/ca/56421587/sdp"/>
199 </entry>
```

200 ServiceDeliveryPoint

```
202 <!-- /c/12345678910/ca/56421587/sdp/85945261 -->
203 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
204   <id>85945261</id>
205   <m:MeterAsset>
206     <m:ID>19283746</m:ID>
207   </m:MeterAsset>
208   <m:MeterReading href="https://{IP Address}/c/12345678910/ca/56421587/sdp/85945261/mr"/>
209     <m:name>Guest House</m:name>
210     <m:ServiceCategory>
211       <m:kind>electricity</m:kind>
212     </m:ServiceCategory>
213 </entry>
```

214 MeterReading

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

```
216 <!-- /c/12345678910/ca/56421587/sdp/85945261/mr/1 -->
217 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
218   <id>1</id>
219   <m:IntervalReading href="https://{IP Address}/c/12345678910/ca/56421587/sdp/85945261/mr/1/ir"/>
220     <m:ReadingType>
221       <m:ID>1001</m:ID>
222     </m:ReadingType>
223 </entry>
```

IntervalReading

```
226 <!-- /c/12345678910/ca/56421587/sdp/85945261/mr/1/ir/2010-12-17T11_00_00Z -->
227 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
228   <m:endTimeStamp>2010-12-17T11:00:00Z</m:endTimeStamp>
229   <m:ReadingQuality>
230     <m:quality>interpolated</m:quality>
231   </m:ReadingQuality>
232   <m:timeStamp>2010-12-17T10:00:00Z</m:timeStamp>
233   <m:value>0.0035</m:value>
234 </entry>
```

ReadingType

```
237 <!-- /rt/1001 -->
238 <entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://osgug.ucaiug.org/ns/2010/06/oade">
239   <id>1001</id>
240   <defaultQuality>validated</defaultQuality>
241   <direction>delivered</direction>
242   <kindReading>energy</kindReading>
243   <multiplier>k</multiplier>
244   <name>Energy Delivered kWh</name>
245   <unitSymbol>Wh</unitSymbol>
246 </entry>
```

3 PATTERNS

249 This section contains guidance and decisions on how message exchanges flow for the general scenarios below. In
250 general, the constructs and operations defined in [9] AtomPub shall be used, including requests for Services,
251 Workspaces, Collections, Members, Categories, and Media Types. Extensions are generally refined subsets of the
252 full specifications detailed in [G2] GData, and full implementations should not break clients who only implement
253 these recommendations.

3.1 CREATING, UPDATING, DELETING

255 The POST method is to be used for creation of new entries, PUT is to be used for updates to existing entries, and
256 DELETE is to be used to delete an entry.

3.2 QUERY, REQUEST AND RESPONSE (RETRIEVE) FORMATS

258 This section specifies the input parameters that can be passed to GET method operations for format, category,
259 reference expansion, sorting, filtering, and iteration through list items.

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

260 3.2.1 FORMAT

261 The default, and only required format, will be CIM-extended AtomPub feed / entry XML.

262 3.2.2 CATEGORY

263 Specification of the category of entries is not included in this recommendation. However it can be accomplished
264 using the Atom element "term", and if needed could be supported as a qualifier in queries by accepting category
265 terms as inputs. In general, each CIM object class would be a category of entry, so that the representations of
266 entries can be specified with a schema.

267 3.2.3 REFERENCE EXPANSION

268 By default, feed queries will return a list of resource links. If the reference expansion flag is set, entries returned
269 will be expanded to contain their full representation.

270 (Need to determine if nested expansion is necessary / possible, and if so, how to specify to what level)

271 [G3] OData uses an `m:inline` extension to the `atom:link` element for this purpose.

272 3.2.4 SORTING

273 Ability to specify the sort order of resulting query / request entries is not necessary - subsequent processing of
274 received data can display or rearrange data however desired. However, ordering of entries shall remain consistent
275 across requests, so that an iterator can be used to page through results. Collection results shall be sorted by date
276 of last modification in descending order. (Latest to earliest updated)

277 3.2.5 FILTERING

278 Filtering requires inputs that allow the specification of the resource name and/or path, as well as a range of
279 publication or update date/times. Properties of the entry element (defined by its category type) could be defined
280 to be acceptable by default as filter terms for the associated resource. Need to determine if it is feasible to
281 implement all, or if identification is necessary of only the filter terms required for specific use cases. Possibly usage
282 patterns could determine the need for indexing, etc.

283 If a specific entry ID is specified, that entry is returned.

284 3.2.6 ITERATION

285 Iteration inputs allow consumers to request a subset of entries or references at a time, and then page through
286 them for processing. Inputs include the starting entry index, and number of entries per page.

287 Query results shall include the additional openSearch terms as defined in [G2] GData for iteration.

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

288 **3.2.7 CONDITIONAL RETRIEVAL**

289 If-None-Match in HTTP header shall be used to retrieve entries only if they have changed as defined in both [G3]
290 OData and [G2] GData.

291 **3.3 EVENT NOTIFICATION (PUB/SUB)**

292 The publish / subscribe pattern is incredibly useful, and is specified mostly in [9] AtomPub. However, there is no
293 mechanism defined in AtomPub to notify subscribers of new feed entries. This requires them to “poll” for new
294 data, and while this is sufficient in many cases, some business processes require ability to notify in order to achieve
295 reduced latency in client updates.

296 [8] PubSubHubbub defines a mechanism for this purpose, and may be implemented for this purpose.

297 **3.4 BATCH TRANSFERS**

298 If desired, a feed for each data service consumer could be provided through which all subscribed content would be
299 returned in a single request (or series of large chunks). This mechanism should allow any resource type to be
300 included within a single feed. The regular synchronization behavior shall be implemented as a regular feed request
301 or subscription notification with reference expansion specified, as defined in [G2] GData.

302 **4 DISCOVERY**

303 Discovery of available resources may utilize the [9] AtomPub constructs defining services, workspaces, and
304 collections. This is accomplished with a client request to GET the definition of all collections, followed by
305 enrollment / authorization, and finally subscription to the appropriate feeds.

306 **5 METADATA**

307 A “Resource” resource may allow retrieval of the representation of all available resources, and the currently
308 implemented version of each.

309 [9] AtomPub defines a “workspace collection” for this.

310 **6 EXTENSIBILITY**

311 [9] AtomPub is specified to be extensible, and implementations should be able to function even with different
312 versions of client or server. In addition, section 6.2 in AtomPub provides recommended behavior.

313 Extensions to the CIM objects will be associated with specific versions of the namespace, specified in the version
314 attribute of the schema element.

315 **7 VERSIONING**

316 As additional capabilities are added to the interface definition, a specification of the version of the definition will
317 be needed to help in service discovery negotiation. This should not change the namespace of any definitions.

UCAIug OpenSG OpenADE Task Force

OpenADE 1.0 Service Definition - REST Extension

318 8 CONCURRENCY

319 In most cases, Resources being utilized are Read Only. Should a need for an update to a Resource be determined;
 320 In order to ensure data integrity, clients may only update resources if they are updating the current version of the
 321 resource. If an update request fails due to conflict (not current version), the client must request the latest version,
 322 apply changes to that representation, and retry the update.

323 [G2] GData and [G3] OData both use ETags for versioning / concurrency management, and are largely compatible.
 324 Additional details will be specified as identified during initial implementations.

325 See GData Resource Versioning <http://code.google.com/apis/gdata/docs/2.0/reference.html#ResourceVersioning>
 326 for more information.

327 9 RESOURCE DEFINITIONS

328 The following tables provide an overview of the service resources to be implemented by provider and consumer
 329 (3rd Party). Full definitions and expected behavior will be developed in a subsequent publication.

330 9.1 PROVIDER (UTILITY) RESOURCES

Resource	Method	Inputs (Resource, ...)	Outputs	Description
Service	GET	ResourceList	ServiceStatus	Synchronously check connectivity and current operational status of the service
Service	POST	ResourceList	RequestStatus	Asynchronously check connectivity and current operational status of the service
Service	POST	ServiceStatus	RequestStatus	Receive result of status check initiated by Utility
Enrollment	POST	Customer, Key, ResourceList	ActivityRecord	Initiate authorization of 3rd Party customer to receive Utility customer resources
Enrollment	POST	Customer, ResourceList	ActivityRecord	Notify Utility of new authorization completion (future)
Enrollment	POST	Customer, ResourceList	ActivityRecord	Initiate cancel authorization of customer resources
Enrollment	POST	Customer, ResourceList	ActivityRecord	Notify Utility of authorization cancellation
ActivityRecord	GET	ID	ActivityRecord	Receive status of an asynchronous request from Utility
Resource	GET	Format	Resource	Transfer customer usage information data (or other resources, future)
ActivityRecord	POST	ResourceList	RequestStatus	Notify Utility of current status of pending transfers

331 **Table 1: Provider Resource Operations**

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

332 9.2 CONSUMER (3RD PARTY) RESOURCES

333

Resource	Method	Inputs (Resource, ...)	Outputs	Description
Service	GET	ResourceList	ServiceStatus	Synchronously check connectivity and current operational status of the service
Service	POST	ResourceList	RequestStatus	Asynchronously check connectivity and current operational status of the service
Service	POST	ServiceStatus	RequestStatus	Receive result of status check initiated by 3rd Party
Enrollment	POST	Customer, Key, ResourceList	ActivityRecord	Initiate authorization of Utility customer to receive 3rd Party customer resources (future)
Enrollment	POST	Customer, ResourceList	ActivityRecord	Notify 3rd Party of new authorization completion (future)
Enrollment	POST	Customer, ResourceList	ActivityRecord	Initiate cancel authorization of customer resources
Enrollment	POST	Customer, ResourceList	ActivityRecord	Notify 3rd Party of authorization cancellation
ActivityRecord	GET	ID	ActivityRecord	Receive status of an asynchronous request from 3rd Party
Resource	POST	ResourceList	RequestStatus	Notify 3rd Party that resources were created or updated
Resource	POST	ID	RequestStatus	Notify 3rd Party that new and updated resource files are available

334

Table 2: Consumer Resource Operations

335 9.3 RESOURCE DEFINITION

336 All resources support the patterns in Section 3, returning a (possibly expanded) collection of data resource stream
 337 entries. Individual entries can be managed using POST, UPDATE, and DELETE. Permissions may be set according to
 338 policy, but guidance is provided regarding the typical configuration in resource details following this table.

339 9.4 RESOURCE DETAILS

340 Many of the resources below are necessary to support initial setup and authorization. Implementations shall
 341 conform to referenced specifications for details on these interfaces. Clarifications and refinements made to
 342 support these service resources are denoted where necessary.

343 All resources are to be implemented as collections, with the elements listed in the sections below. Additional
 344 allowed values may be specified, and will be included here as necessary.

345 9.4.1 COLLECTION (FEED)

346 Collections consist of feeds, and exhibit behavior as defined in [9] AtomPub.

UCAIug OpenSG OpenADE Task Force
OpenADE 1.0 Service Definition - REST Extension

347 9.4.2 AUTHORIZATION

348 The "Enrollment" resources shall use OAuth 2.0 definitions to allow user owners of provider data resources to
349 grant authorization to 3rd Party for access.