



Capabilities Selection Extension

Version 1.1c

ABSTRACT:

"Publication of this Working Draft for review and comment has been approved by the Cloud Storage Technical Working Group. This draft represents a "best effort" attempt by the Cloud Storage Technical Working Group to reach preliminary consensus, and it may be updated, replaced, or made obsolete at any time. This document should not be used as reference material or cited as other than a 'work in progress.' Suggestion for revision should be directed to <http://snia.org/feedback>."

Working Draft

March 2, 2016

USAGE

The SNIA hereby grants permission for individuals to use this document for personal use only, and for corporations and other business entities to use this document for internal use only (including internal copying, distribution, and display) provided that:

Any text, diagram, chart, table or definition reproduced shall be reproduced in its entirety with no alteration, and,

Any document, printed or electronic, in which material from this document (or any portion hereof) is reproduced shall acknowledge the SNIA copyright on that material, and shall credit the SNIA for granting permission for its reuse.

Other than as explicitly provided above, you may not make any commercial use of this document, sell any or this entire document, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.

Permission to use this document for purposes other than those enumerated above may be requested by e-mailing tcmd@snia.org. Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

All code fragments, scripts, data tables, and sample code in this SNIA document are made available under the following license:

BSD 3-Clause Software License

Copyright © 2016, The Storage Networking Industry Association.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

DISCLAIMER

The information contained in this publication is subject to change without notice. The SNIA makes no warranty of any kind with regard to this specification, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this specification.

Suggestions for revisions should be directed to <http://www.snia.org/feedback/>.

Copyright © 2016 SNIA. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

Revision History

Date	Version	By	Comments
2016-02-05	1.1a	David Slik	First draft
2016-02-17	1.1b	Cloud TWG	TWG edits
2016-03-02	1.1c	Cloud TWG	Fixed typo, edits for public review

Capabilities Selection Extension

Overview

The capabilities objects in /<cdmi root>/cdmi_capabilities/ describe sets of capabilities supported for stored objects. This extension extends CDMI to permit capabilities objects to also contain data system metadata values that act as default values and permit the capabilitiesURI to be specified on create or update, which allows an object to inherit these data system metadata values.

Modifications to the CDMI 1.1 spec:

1) In Clause 8.2.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The newly created object shall inherit data system metadata values from the specified capabilities object.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the create operation, the contents of the data system metadata items in the create operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created object may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

2) In Clause 8.2.9, add example 5:

EXAMPLE 5 PUT to the container URI to create a "gold" object:

```

PUT /MyContainer/MyGoldObject.txt HTTP/1.1
Host: cloud.example.com
Accept: application/cdmi-object
Content-Type: application/cdmi-object
X-CDMI-Specification-Version: 1.1.1

{
  "mimetype": "text/plain",
  "capabilitiesURI": "/cdmi_capabilities/dataobject/gold/"
  "value": "This is the Value of this Data Object"
}

```

The following shows the response:

```

HTTP/1.1 201 Created
Content-Type: application/cdmi-object
X-CDMI-Specification-Version: 1.1.1

```

```

{
  "objectType": "application/cdmi-object",
  "objectID": "00007ED90010D891022876A8DE0BC0FD",
  "objectName": "MyGoldObject.txt",
  "parentURI": "/MyContainer/",
  "parentID": "00007E7F00102E230ED82694DAA975D2",
  "domainURI": "/cdmi_domains/MyDomain/",
  "capabilitiesURI": "/cdmi_capabilities/dataobject/gold/",
  "completionStatus": "Complete",
  "mimetype": "text/plain",
  "metadata": {
    "cdmi_size": "37",
    <Data System Metadata Inherited from Gold Capabilities>
  }
}

```

3) In Clause 8.4.3, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The updated object shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the update operation, the contents of the data system metadata items in the update operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created object may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

4) In Clause 8.4.8, add example 11:

EXAMPLE 11 PUT to the data object URI to change a "gold" object to a "silver" object:

```
PUT /MyContainer/MyGoldObject.txt HTTP/1.1
Host: cloud.example.com
Content-Type: application/cdmi-object
X-CDMI-Specification-Version: 1.1.1
```

```
{
  "capabilitiesURI": "/cdmi_capabilities/dataobject/silver/"
}
```

The following shows the response:

```
HTTP/1.1 204 No Content
```

EXAMPLE 12 PUT to the data object URI to change a "gold" object to a "silver" object, including a manually specified data system metadata item value (also required to preserve any data system metadata item values that were previously specified manually):

```
PUT /MyContainer/MyGoldObject.txt HTTP/1.1
Host: cloud.example.com
Content-Type: application/cdmi-object
X-CDMI-Specification-Version: 1.1.1
```

```
{
  "capabilitiesURI": "/cdmi_capabilities/dataobject/silver/",
  "metadata": {
    "cdmi_data_redundancy": "3"
  }
}
```

The following shows the response:

```
HTTP/1.1 204 No Content
```

5) In Clause 9.2.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The newly created container shall inherit data system metadata values from the specified capabilities object.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the create operation, the contents of the data system metadata items in the create operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created container may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code 400 Bad Request shall be returned.</p>	Optional

6) In Clause 9.2.9, add example 5:

EXAMPLE 5 PUT to the container URI to create a "gold" container:

```
PUT /MyContainer/MyGoldContainer/ HTTP/1.1
Host: cloud.example.com
Accept: application/cdmi-container
Content-Type: application/cdmi-container
X-CDMI-Specification-Version: 1.1.1
```

```
{
  "capabilitiesURI": "/cdmi_capabilities/container/gold/"
}
```

The following shows the response:

```
HTTP/1.1 201 Created
Content-Type: application/cdmi-container
X-CDMI-Specification-Version: 1.1.1
```

```
{
  "objectType": "application/cdmi-container",
  "objectID": "00007ED900104E1D14771DC67C27BF8B",
  "objectName": "MyGoldContainer/",
  "parentURI": "/MyContainer/",
  "parentID": "00007E7F0010128E42D87EE34F5A6560",
  "domainURI": "/cdmi_domains/MyDomain/",
  "capabilitiesURI": "/cdmi_capabilities/container/gold/",
  "completionStatus": "Complete",
  "metadata": { <Data System Metadata Inherited from Gold
Capabilities>
  }
}
```


7) In Clause 9.4.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The updated container shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the update operation, the contents of the data system metadata items in the update operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the updated container may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

8) In Clause 9.4.9, add example 3:

EXAMPLE 3 PUT to the container object URI to change a "gold" container to a "silver" container:

```
PUT /MyContainer/MyGoldContainer/ HTTP/1.1
Host: cloud.example.com
Content-Type: application/cdmi-container
X-CDMI-Specification-Version: 1.1.1
```

```
{
  "capabilitiesURI": "/cdmi_capabilities/container/silver/"
}
```

The following shows the response:

```
HTTP/1.1 204 No Content
```

9) In Clause 9.6.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The newly created object shall inherit data system metadata values from the specified capabilities object.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the create operation, the contents of the data system metadata items in the create operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created object may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

10) In Clause 9.7.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The newly created queue shall inherit data system metadata values from the specified capabilities object.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the create operation, the contents of the data system metadata items in the create operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created queue may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

11) In Clause 11.2.5, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The newly created queue shall inherit data system metadata values from the specified capabilities object.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the create operation, the contents of the data system metadata items in the create operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the created queue may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

12) In Clause 11.4.4, add to table below domainURI:

Field Name	Type	Description	Requirement
capabilitiesURI	JSON String	<p>URI of a capabilities object. The updated queue shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present.</p> <p>If data system metadata items are specified in the capabilities object and in the metadata field included with the update operation, the contents of the data system metadata items in the update operation metadata field shall take precedence.</p> <p>The capabilitiesURI of the updated queue may not match the specified capabilitiesURI, depending on the object type, specified data system metadata items, and other system configurations.</p> <p>If an invalid URI is specified, an HTTP status code of 400 Bad Request shall be returned.</p>	Optional

13) In Clause 12.1, replace:

As an option, a CDMI implementation may map a container to a "gold_container" capabilities entry, if a data system metadata field is present and set to a given value, such as if the `cdmi_data_redundancy` field was set to the value of "4". This permits a cloud provider to create profiles of data system metadata fields and values.

Capabilities do not have a CDMI metadata field.

With:

Multiple capabilities objects can be created by a CDMI implementation and can contain different data system metadata items with different values. A CDMI implementation may map objects to specific capabilities URIs (such as the "gold_container" capabilities URI shown above) when data system metadata fields are present in the created or updated object and match to the data system metadata fields and values in a given capabilities object.

This mapping permits the creation of templates that can be used when creating or updating an object to specify a default set of data system metadata values. For example, the "gold_container" could include a `cdmi_data_redundancy` metadata item with the value set to "4". Any container created with this capabilities object specified shall inherit a "cdmi_data_redundancy" metadata item with the value "4", and, in the absence of additional and/or overriding data system metadata values, shall have a `capabilitiesURI` set to the "gold_container".

14) In Clause 12.1.1, add new rows at end of table "Table 100 - System-Wide Capabilities"

Capability Name	Type	Description
<code>cdmi_capabilities_templates</code>	JSON String	If present and "true", this capability indicates that capabilities objects can be used as templates when creating or modifying a CDMI object.
<code>cdmi_capabilities_exact_inherit</code>	JSON String	If present and "true", this capability indicates that the data system metadata from the capabilities object shall completely replace all existing data system metadata in an object when updated or created.

15) In Clause 12.2.6, add to table below capabilities:

Field Name	Type	Description	Requirement
Metadata	JSON Object	Metadata for the capabilities object. This field includes data system metadata associated with the capabilities object. See Clause 16 for a further description of data system metadata.	Mandatory

16) In Clause 12.2.8, update example 1 to include after "capabilities":

```
"metadata": {
```

```
    "cdmi_data_redundancy": "4"  
  },
```

DRAFT