Capabilities Selection CDMI Extension

Version 2.0

ABSTRACT: This CDMI Extension is intended for developers who are considering a standardized way to add functionality to CDMI. When multiple compatible implementations are demonstrated and approved by the Technical Working Group, this extension will be incorporated into the CDMI standard.

This document has been released and approved by the SNIA. The SNIA believes that the ideas, methodologies, and technologies described in this document accurately represent the SNIA goals and are appropriate for widespread distribution. Suggestion for revision should be directed to http://www.snia.org/feedback/.

SNIA Working Draft

June 17, 2020

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 https://www.snia.org/feedback/.

Contents

Clause 1

Capabilities Selection CDMI Extension

1.1 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.

1.2 Instructions to the Editor

To merge this extension into the CDMI 2.0.0 specification, make the following changes:

1. Add an entry immediately below the `domainURI` entry in the table starting on line 138 of cdmi_core/cdmi_data_object/create.txt, as follows:

Table 1: Request message body - Create a data object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The newly created data object shall inherit data system metadata values from the specified capabilities object. 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. The capabilitiesURI of the created data object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

2. Add the following example to the end of the examples section starting on line 394 of cdmi_core/cdmi_data_object/create.txt, as follows:

EXAMPLE 8: PUT to the container URI to create a “gold” object:

```
--> PUT /cdmi/2.0.0/MyContainer/MyGoldObject.txt HTTP/1.1
--> Host: cloud.example.com
--> Accept: application/cdmi-object
--> Content-Type: application/cdmi-object
-->
--> {
-->   "mimetype" : "text/plain",
```

(continues on next page)

(continued from previous page)

```
-->   "capabilitiesURI": "/cdmi_capabilities/dataobject/gold/",
-->   "value" : "This is the Value of this Data Object"
--> }

<-- HTTP/1.1 201 Created
<-- Content-Type: application/cdmi-object
<--
<-- {
<--   "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. Add an entry immediately below the `domainURI` entry in the table starting on line 110 of cdmi_core/cdmi_data_object/update.txt, as follows:

Table 2: Request message body - Update a CDMI data object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The updated data object shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present. 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. The capabilitiesURI of the updated data object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

4. Add the following examples to the end of the examples section starting on line 291 of cdmi_core/cdmi_data_object/update.txt, as follows:

EXAMPLE 13: PATCH to the data object URI to change a “gold” object to a “silver” object:

```
--> PATCH /cdmi/2.0.0/MyContainer/MyGoldObject.txt HTTP/1.1
--> Host: cloud.example.com
--> Content-Type: application/cdmi-object
-->
--> {
-->   "capabilitiesURI": "/cdmi_capabilities/dataobject/silver/"
--> }

<-- HTTP/1.1 204 No Content
```

EXAMPLE 14: PATCH 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):

```
--> PATCH /cdmi/2.0.0/MyContainer/MyGoldObject.txt HTTP/1.1
--> Host: cloud.example.com
```

(continues on next page)

(continued from previous page)

```
--> Content-Type: application/cdmi-object
-->
--> {
-->   "capabilitiesURI": "/cdmi_capabilities/dataobject/silver/",
-->   "metadata": {
-->     "cdmi_data_redundancy": "3"
-->   }
--> }

<-- HTTP/1.1 204 No Content
```

5. Add an entry immediately below the `domainURI` entry in the table starting on line 112 of cdmi_core/cdmi_container_object/create.txt, as follows:

Table 3: Request message body - Create a container object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The newly created container object shall inherit data system metadata values from the specified capabilities object. 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. The capabilitiesURI of the created container object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code `400 Bad Request` shall be returned.	Optional

6. Add the following example to the end of the examples section starting on line 425 of cdmi_core/cdmi_container_object/create.txt, as follows:

EXAMPLE 5: PUT to the container URI to create a “gold” container:

```
--> PUT /cdmi/2.0.0/MyContainer/ HTTP/1.1
--> Host: cloud.example.com
--> Accept: application/cdmi-container
--> Content-Type: application/cdmi-container
-->
--> {
-->   "capabilitiesURI" : "/cdmi_capabilities/container/gold/"
--> }

<-- HTTP/1.1 201 Created
<-- Content-Type: application/cdmi-container
<--
<-- {
<--   "objectType" : "application/cdmi-container",
<--   "objectID" : "00007ED900104E1D14771DC67C27BF8B",
<--   "objectName" : "MyContainer/",
<--   "parentURI" : "/",
<--   "parentID" : "00007E7F0010128E42D87EE34F5A6560",
<--   "domainURI" : "/cdmi_domains/MyDomain/",
<--   "capabilitiesURI" : "/cdmi_capabilities/container/gold/",
<--   "completionStatus" : "Complete",
<--   "metadata" : {
<--     <Data System Metadata Inherited from Gold Capabilities>
<--     ...
<--   },
<--   "childrenrange": "",
<--   "children": []
<-- }
```

7. Add an entry immediately below the `domainURI` entry in the table starting on line 111 of cdmi_core/cdmi_container_object/update.txt, as follows:

Table 4: Request message body - Update a container object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The updated container object shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present. 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. The capabilitiesURI of the updated container object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

8. Add the following example to the end of the examples section starting on line 275 of cdmi_core/cdmi_container_object/update.txt, as follows:

EXAMPLE 3: PATCH to the container object URI to change a “gold” container to a “silver” container:

```
--> PATCH /cdmi/2.0.0/MyContainer/ HTTP/1.1
--> Host: cloud.example.com
--> Content-Type: application/cdmi-container
-->
--> {
-->   "capabilitiesURI" : "/cdmi_capabilities/container/silver/"
--> }

<-- HTTP/1.1 204 No Content
```

9. Add an entry immediately below the `domainURI` entry in the table starting on line 158 of cdmi_core/cdmi_container_object/create_post_object.txt, as follows:

Table 5: Request message body - Create a new data object Using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The newly created data object shall inherit data system metadata values from the specified capabilities object. 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. The capabilitiesURI of the created data object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

10. Add an entry immediately below the `domainURI` entry in the table starting on line 146 of cdmi_core/cdmi_container_object/create_post_queue.txt, as follows:

Table 6: Request message body - Create a new queue object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The newly created queue object shall inherit data system metadata values from the specified capabilities object. 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. The capabilitiesURI of the created queue object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

11. Add an entry immediately below the `domainURI` entry in the table starting on line 265 of cdmi_advanced/cdmi_queue_object.txt, as follows:

Table 7: Request message body - Create a queue object using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The newly created queue object shall inherit data system metadata values from the specified capabilities object. 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. The capabilitiesURI of the created queue object may not match the specified `capabilitiesURI`, depending on the object type, specified data system metadata items, and other system configurations. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

12. Add an entry immediately below the `domainURI` entry in the table starting on line 1087 of cdmi_advanced/cdmi_queue_object.txt, as follows:

Table 8: Request message body - Update a queue object Using CDMI

Field Name	Type	Description	Requirement
`capabilitiesURI`	JSON string	URI of a capabilities object. The updated queue object shall inherit data system metadata values from the specified capabilities object, overwriting data system metadata values if already present. 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 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. If an invalid URI is specified, an HTTP status code of `400 Bad Request` shall be returned.	Optional

13. Replace the below text starting on line 1087 of cdmi_advanced/cdmi_capability_object.txt, as follows:

Capabilities cannot be altered by clients, but may be changed by the CDMI server to reflect configuration changes or operational changes. For example, if a CDMI server is upgraded or reconfigured, additional capabilities may become present, or existing capabilties may no longer be present. In practice, capabilities rarely change, and a client can assume that they shall remain constant for the duration of a client-server HTTP/HTTPS session.

Replace with:

Capabilities cannot be altered by clients, but may be created or changed by the CDMI server to reflect configuration changes or operational changes. For example, if a CDMI server is upgraded or reconfigured, additional capabilities may become present, or existing capabilties may no longer be present. In practice, capabilities rarely change, and a client can assume that they shall remain constant for the duration of a client-server HTTP/HTTPS session.

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 CDMI servers to create 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. Add an entry to the end of the table starting on line 135 of cdmi_advanced/cdmi_capability_object.txt, as follows:

Table 9: System-wide capabilities

Capability name	Type	Definition
`cdmi_capabilities_templates`	JSON string	If present and “`true`”, the CDMI server supports capabilities objects being used as templates when creating or modifying a CDMI object.
`cdmi_capabilities_exact_inherit`	JSON string	If present and “`true`”, the CDMI server supports inheriting data system metadata when a capabilities object is specified on create or update, which completely replace all existing data system metadata not explicitly specified in the object creation or update.

15. Add an entry immediately below the `capabilities` entry in the table starting on line 1015 of cdmi_advanced/cdmi_capability_object.txt, as follows:

Table 10: Response message body - Read a capabilities object using CDMI

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. Update example 1 in the examples section starting on line 1113 of cdmi_advanced/cdmi_capability_object.txt, as follows:

EXAMPLE 1: GET to the root container capabilities URI to read all fields of the container:

```
--> GET /cdmi/2.0.0/cdmi_capabilities/ HTTP/1.1
--> Host: cloud.example.com
--> Accept: application/cdmi-capability

<-- HTTP/1.1 200 OK
<-- Content-Type: application/cdmi-capability
<--
<-- {
<--   "objectType": "application/cdmi-capability",
<--   "objectID": "00007E7F00104BE66AB53A9572F9F51E",
<--   "objectName": "cdmi_capabilities/",
<--   "parentURI": "/",
<--   "parentID": "00007E7F0010128E42D87EE34F5A6560",
<--   "capabilities": {
<--     "cdmi_domains": "true",
<--     "cdmi_export_nfs": "true",
<--     "cdmi_export_iscsi": "true",
<--     "cdmi_queues": "true",
<--     "cdmi_notification": "true",
<--     "cdmi_query": "true",
<--     "cdmi_metadata_maxsize": "4096",
```

(continues on next page)

(continued from previous page)

```
<--     "cdmi_metadata_maxitems": "1024"
<--   },
<--   "metadata" : {
<--     <Data System Metadata for the capabiliity object>
<--     ...
<--   },
<--   "childrenrange": "0-3",
<--   "children": [
<--   "domain/",
<--   "container/",
<--   "dataobject/",
<--   "queue/"
<--   ]
<-- }
```