S3 Exports CDMI Extension

Version 2.0.0

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.

SNIA Working Draft

March 29, 2024

DISCLAIMER

The information contained in this publication is subject to change without notice. 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. 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: S3 Exports CDMI Extension **1**
1.1 Overview . . . 1
1.2 Instructions to the Editor . . . 1
1.3 7.3 Read a container object using HTTP . . . 2
1.4 8.2.7 Data object representations . . . 2
1.5 9.2.6 Container object representations . . . 3
1.6 13.1 Overview . . . 3
1.7 13.2.2 Container object export fields . . . 3
1.8 13.8 S3 exported protocol . . . 3

Clause 1

S3 Exports CDMI Extension

1.1 Overview

The existing CDMI specification permits CDMI containers to be declaratively exported via network protocols such as NFS, CIFS and iSCSI. As S3 protocol access to objects has become widely used, the same “exports” facility can be extended to support exporting a CDMI container as an S3 bucket to allow S3 clients to transparently access data stored on a storage system managed using CDMI.

This extension proposes a new export protocol type of “s3”, along with export parameters necessary for specifying how the contents of a container are mapped to an S3 bucket. This extension also includes required changes to permit mapping of allowable names, handing for directories that contain a value, and values that contain a directory.

1.2 Instructions to the Editor

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

1.2.1 terms.txt

Insert into preamble/terms.txt, as follows:

x.1
S3 bucket
: **|br|** a distinctly named logical grouping of objects made available using the S3 protocol **|br|**

x.2
S3 bucket properties
: **|br|** behavioural settings associated with an S3 bucket. **|br|**

x.3
S3 protocol
: **|br|** a widely used proprietary object storage protocol developed by Amazon and used by the S3 “Simple Storage Service” **|br|**

1.2.2 cdmi_capability_object.txt

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

Table 1.1: System-wide capabilities

Capability name	Type	Definition
`cdmi_export_s3`	JSON string	If present and "`true`", the CDMI server supports S3 exports.
`cdmi_containers`	JSON string	If present and `true`", the CDMI server supports container objects.
`cdmi_dataobjects_as_containers`	JSON string	If present and `true`", the CDMI server supports accessing data objects as container objects.
`cdmi_containers_as_dataobjects`	JSON string	If present and `true`", the CDMI server supports accessing container objects as data objects.

Add an entry to the end of the table starting on line 612 of cdmi_advanced/cdmi_capability_object.txt, as follows:

Table 1.2: Capabilities for data objects

Capability name	Type	Definition
`cdmi_as_container`	JSON string	If present and "`true`", this capability indicates that the CDMI server shall support the ability to access the data object as a container.

Add an entry to the end of the table starting on line 668 of cdmi_advanced/cdmi_capability_object.txt, as follows:

Table 1.3: Capabilities for container objects

Capability name	Type	Definition
`cdmi_as_dataobject`	JSON string	If present and "`true`", this capability indicates that the CDMI server shall support the ability to access the container as a data object.

1.2.3 http_container_object/read.txt

1.3 7.3 Read a container object using HTTP

Replace the first paragraph with:

Reading a container object using HTTP is only defined if a data object representation exists for that container. If this is the case, the process described in clause 6.3. If a data object representation does not exist for the container, a server is allowed to return an error response, or to implement custom responses such as an Apache directory listing or an S3-style bucket listing.

1.3.1 cdmi_data_object/overview.txt

1.4 8.2.7 Data object representations

Add new paragraph to end of section:

If the `cdmi_dataobjects_as_containers` capability is supported by the CDMI server and the `cdmi_as_container` capability is present for the data object, CDMI servers shall return a container object representations of a data object when created updated or requested by specifying a `Content-Type` and/or `Accept` header the vaue of `application/cdmi-container`. The CDMI server shall also support the creation, modification and access to child objects by URI under any data object.

1.4.1 cdmi_container_object/overview.txt

1.5 9.2.6 Container object representations

Add new paragraph to end of section:

If the `cdmi_containers_as_dataobjects` capability is supported by the CDMI server and the `cdmi_as_dataobject` capability is present for the container object, CDMI servers shall return a data object representations of a container object when created updated or requested by specifying a `Content-Type` and/or `Accept` header the vaue of `application/cdmi-container`. The CDMI server shall also support storing and accessing a value associated with a container object.

1.5.1 cdmi_exports.txt

1.6 13.1 Overview

Add “S3 Bucket” to the end of the example list at the end of the second bullet.

1.7 13.2.2 Container object export fields

Replace “ACL” with “CDMI ACL” at the end of the third bullet.

1.8 13.8 S3 exported protocol

Add new section, with the following text:

An S3 export specifies the information required by an S3 server to provide S3 protocol access to a bucket.

Elements for an S3 export are described in Table 1.4.

Table 1.4: Elements of the S3 protocol export structure

Element	Type	Description	Requirement
`type`	JSON String	The export type is set to “`s3`”	Mandatory
`bucketname`	JSON String	The name of the bucket into which the container should be exported. This value shall be between 3 and 63 characters in length, and shall only include lowercase ASCII letters [a-z], ASCII numbers [0-9], non-consecutive repeating periods (”.”) and non-consecutive repeating hyphens (”-“).	Mandatory
`read_only`	JSON String	This value shall be either “`true`” or “`false`”. The default shall be “`false`”. When true, no modifications will be permitted to be made through the S3 protocol. If not present, the default value shall be used.	Optional
`public`	JSON String	This value shall be either “`true`” or “`false`”. The default shall be “`false`”. When true, no access control checks will be performed for operations performed using the S3 protocol. If not present, the default value shall be used.	Optional

EXAMPLE 6: S3 exports

```
{
    "exports" : {
        "1" : {
            "type" : "s3",
            "bucket" : "mybucket",
            "read_only" : "true"
```

(continues on next page)

(continued from previous page)

```
            }
        }
}
```