Data Deduplication Metadata Extension

Version 1.1c

ABSTRACT: This document describes a proposed extension to the SNIA Cloud Data Management Interface (CDMI) International Standard.

Working Draft

December 10, 2014

USAGE

DISCLAIMER

Revision History

Date	Version	By	Comments
2014-05-13	1.0a	CDMI TWG	Document created at San Diego Face to Face meeting by Zhaoming Ding from ZTE corporation
2014-11-11	1.1a	Zhaoming Ding	Updated for sending of list of data fingerprints and supply examples
2014-11-20	1.1a	Ghazanfar Ali	Updated based on meeting discussions.
2014-12-01	1.1b	Marie McMinn	Performed technical edit.
2014-12-10	1.1c	CDMI TWG	Final edits in preparation for public review.

Data Deduplication Metadata Extension

Overview

Data deduplication is a capacity optimization technology that is used to dramatically improve storage efficiency. Data deduplication is the process of eliminating redundant copies of data. In the context of storage, deduplication refers to any algorithm that searches for duplicate data objects, e.g. blocks, chunks, and files, and stores only a single copy of those objects.

The benefits of data deduplication are clear: it can reduce the space needed to store data and increase the available space to retain data for longer periods of time.

The current CDMI protocol does not enable CDMI-aware clients to use or choose data deduplication techonologies. This extension extends CDMI to enable clients to transfer data to and from a CDMI server more efficiently using data deduplication.

Introduction for Data Deduplication Flows

For client backup deduplication, data fingerprints (hash-based calculations) are created on the client, and the CDMI server creates appropriate internal links to reference duplicated data.

Deduplication is accomplished by using an HTTP PUT to create an object and register fingerprints for chunks of that object, where a list of fingerprints used to create the object.

One or more fingerprints indicate that portions of the object being created may already be known to the CDMI server, and fingerprints together with data indicate that portions of the object being created are being registered with the CDMI server.

A client can reduce bandwidth by first performing a PUT containing only fingerprints in the fingerprintmap field, each corresponding to a portion of the object to be created, then performing a second PUT containing data along with fingerprints in the fingerprintmap field, each corresponding to previously unknown fingerprints returned in the 409 Conflict response to the first PUT.

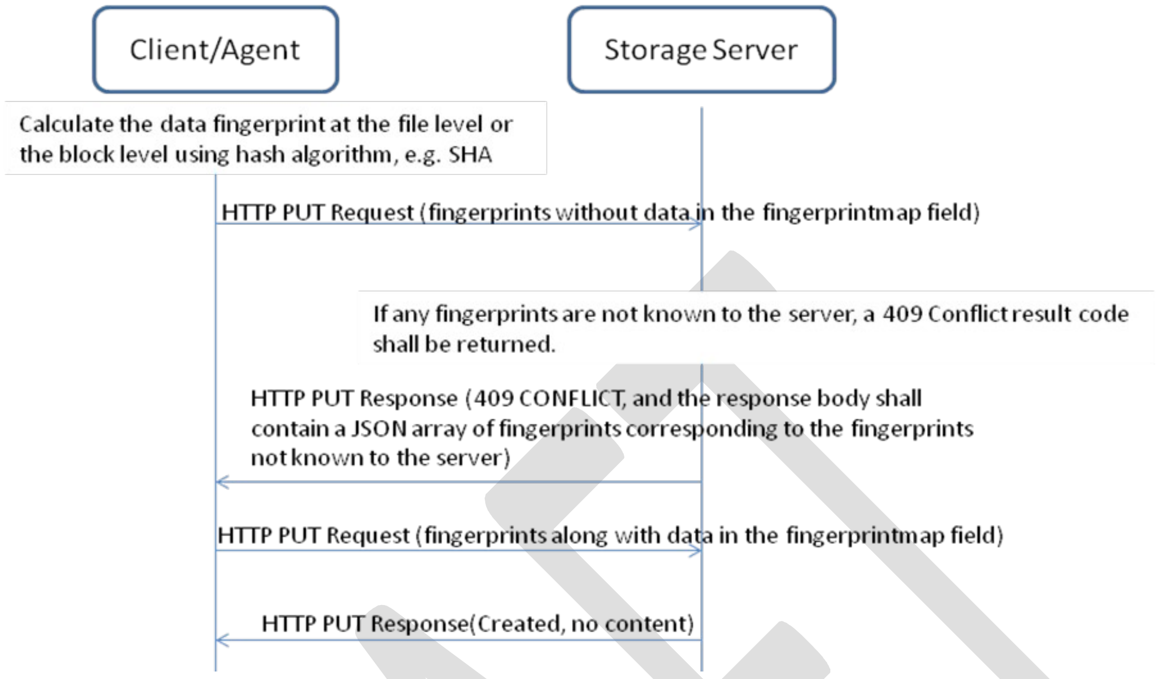
Client/Agent
Storage Server
Calculate the data fingerprint at the file level or the block level using hash algorithm, e.g. SHA
HTTP PUT Request (fingerprints without data in the fingerprintmap field)
If any fingerprints are not known to the server, a 409 Conflict result code shall be returned.
HTTP PUT Response (409 CONFLICT, and the response body shall contain a JSON array of fingerprints corresponding to the fingerprints not known to the server)
HTTP PUT Request (fingerprints along with data in the fingerprintmap field)
HTTP PUT Response(Created, no content)

Figure 1: PUT to create an object and register fingerprints for chunks of that object

The algorithm used to calculate the fingerprint value is chosen by the client. This extension requires that each fingerprint value represents a unique data block, and recommends that the beginning of the fingerprint is a self-describing indicator of the algorithm used to calculate the fingerprint.

Modifications to the CDMI 1.1.0 spec:

1) Add to end of clause 6.2.4 Request Message Body

A request message body using chunked encoding may contain a mixture of:

- non-zero-length chunks without a chunk extension.
- zero-length chunks with a fingerprint chunk extension to indicate that portions of the object being created may already be known to the CDMI server; and
- non-zero-length chunks with a fingerprint chunk extension to indicate that portions of the object being created are being registered with the CDMI server.

If the server does not know any zero-length fingerprint chunks, an HTTP status code of `409 Conflict` shall be returned, and the response body shall contain a JSON array of fingerprints unknown to the CDMI server.

A client can reduce bandwidth by performing

- a PUT containing only zero-length fingerprint chunk extensions that correspond to each chunk of the object to be created, and
- a second PUT containing a non-zero-length fingerprint chunk extension corresponding to each fingerprint returned in the `409 Conflict` response to the first PUT.

2) Add to end of clause 6.2.6 Response Message Body

If one or more non-zero-length fingerprint chunk extensions are received by the CDMI server that correspond to fingerprint chunks that are unknown to the CDMI server, a JSON array containing the fingerprint chunk extension values corresponding to the fingerprints unknown to the CDMI server shall be returned.

3) Add new examples to 6.2.8 Examples

EXAMPLE 2 PUT to request the creation of an object using fingerprints of possible previously stored data:

```
PUT /MyContainer/MyDataObject.txt HTTP/1.1
Host: cloud.example.com
Content-Type: text/plain;charset=utf-8
Content-Length: 37
Transfer-Encoding: chunked

0;fingerprint=SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205af
f1762d7301a
0;fingerprint=SHA256:30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f
5520f7b2460
```

The following shows the response.

```
HTTP/1.1 409 Conflict
Content-Type: application/json

[
"SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205aff1762d7301a",
"SHA256:30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f5520f7b2460"
]
```

EXAMPLE 3 PUT to create an object and register fingerprints for chunks of that object:

```
PUT /MyContainer/MyDataObject.txt HTTP/1.1
Host: cloud.example.com
Content-Type: text/plain;charset=utf-8
Content-Length: 37
Transfer-Encoding: chunked

4;fingerprint=SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205af
f1762d7301a
This
33;fingerprint=SHA256:30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718
f5520f7b2460
 is the Value of this Data Object
```

The following shows the response.

```
HTTP/1.1 201 Created
```

4) Add new row after "value" row of Table 21 in clause 8.2.5:

fingerprintmap	JSON Array of JSON Objects	A list of fingerprints used to create the object.
		One or more zero-length value fingerprint entries indicate that the CDMI server may already know portions of the object being created, and non-zero-length value fingerprint entries indicate that portions of the object being created are being registered with the CDMI server. If the CDMI server does not know any zero-length value fingerprint entries, an HTTP status code of `409 Conflict` shall be returned, and the response body shall contain a JSON array of fingerprints unknown to the server.
		A client can reduce bandwidth by first performing a PUT containing only zero-length value fingerprint entries in the fingerprintmap field, each corresponding to a chunk of the object to be created, then performing a second PUT containing non-zero-length value fingerprint entries for each corresponding fingerprint returned in the `409 Conflict` response to the first PUT.
		The transfer encoding of fingerprint entry values is determined by the valuetransferencoding field.
		Each fingerprint entry in the fingerprintmap JSON array shall have the following JSON object structure:
		`{`
		`    "fingerprint" : "<fingerprint>",`
			Optional[a]

		`"value" : "<optional value>",` `"part" : "<optional multi-part MIME value reference>"` `}` If a CDMI client includes a “part” in place of a “value”, the CDMI server shall use the contents of the MIME part that corresponds to the MIME value reference in place of the value. The “part” JSON string shall contain an integer greater than one, and the HTTP Request Body shall contain at least that number of multi-part MIME parts.	

5) Add new row after "metadata" row of Table 23 in clause 8.2.7:

If one or more zero-length value fingerprint entries are received that correspond to fingerprints that are unknown to the CDMI server, the response body shall contain the fingerprintmap field with only zero-length value fingerprint entries that are unknown to the CDMI server.

6) Add new examples to 8.2.9 Examples

EXAMPLE 5 PUT to request the creation of an object using fingerprints of possible previously stored data, where the CDMI server knows the first fingerprint:

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

{
    "fingerprintmap": [
        {
            "fingerprint":
"SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205aff1762d7301a",
            "value": ""
        },
        {
            "fingerprint": ""SHA256:
30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f5520f7b2460",
"value" : ""
}
        ]
}
```

The following shows the response.

```
HTTP/1.1 409 Conflict

{
    "fingerprintmap": [
        {
            "fingerprint": ""SHA256:
30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f5520f7b2460",
"value" : ""
}
        ]
}
```

EXAMPLE 6 PUT to create an object by including value for the fingerprints unknown to the CDMI server:

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

{
    "fingerprintMap": [
        {
            "fingerprint":
"SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205aff1762d7301a",
            "value": ""
        },
        {
            "fingerprint": ""SHA256:
30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f5520f7b2460",
"value" : "istheValueofthisDataObject"
}
        ]
}
```

The following shows the response.

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

{
    "objectType": "application/cdmi-object",
    "objectID": "00007ED90010D891022876A8DE0BC0FD",
    "objectName": "MyDataObject.txt",
    "parentURI": "/MyContainer/",
    "parentID": "00007E7F00102E230ED82694DAA975D2",
    "domainURI": "/cdmi_domains/MyDomain/",
    "capabilitiesURI": "/cdmi_capabilities/dataobject/",
    "completionStatus": "Complete",
    "mimetype": "text/plain",
    "metadata": {
        "cdmi_size": "37"
   }
}
```

7) Add new row after "value" row of Table 27 in clause 8.3.6:

fingerprintmap	JSON Array of JSON Objects	A list of fingerprints associated with the object. Each fingerprint entry contains an offset and length corresponding to the data associated with the fingerprint. If a client already has part of the object stored locally, the client can retrieve the fingerprintmap, and subsequently only GET the ranges of the object that are not already locally stored. Byte offsets and lengths are defined according to the object valuetransferencoding.
		Each fingerprint entry in the fingerprintmap JSON array shall have the following JSON object structure:
		`{`
		`"fingerprint" : "<fingerprint >",`
		`"offset" : "<byte position of beginning of range>",`
		`"length" : "<byte length of the range>"`
		`}`
			Optional

8) Add new examples to 8.3.8 Examples

EXAMPLE 8 GET to retrieve the fingerprintmap of an object:

```
GET /MyContainer/MyDataObject.txt?fingerprintmap HTTP/1.1
Host: cloud.example.com
Accept: application/cdmi-object
X-CDMI-Specification-Version: 1.1
```

The following shows the response.

```
HTTP/1.1 200 OK
Content-Type: application/cdmi-object
X-CDMI-Specification-Version: 1.1

{
    "fingerprintmap": [
        {
            "fingerprint":
"SHA256:86e1de74820a9b252ba33b2eed445b0cd02c445b5f4b8007205aff1762d7301a",
            "offset": "0",
            "length": "4"
        },
        {
            "fingerprint": "SHA256:
30e70dda3fb3acd5aafd3e6426613247f2c88b2384ad048ad718f5520f7b2460",
            "offset": "4",
            "length": "33"
        }
    ]
}
```

9) In Clause 12.1.1, add a new row at end of "Table 100 – System-Wide Capabilities"

Capability Name	Type	Definition
cdmi_data_dedupe	JSON String	If present and “true”, this capability indicates that the cloud storage system supports data deduplication.

10) In Clause 12.1.5, add a new row at end of "Table 104 – Capabilities for Containers"

Capability Name	Type	Definition
cdmi_create_dataobject_dedupe	JSON String	If present and “true”, this capability indicates that the cloud storage system shall support the ability of the container to add a new data object that is deduplicated.