

CDNI FCI using ALTO

-04 updates and discussion

Jan Seedorf

Richard Yang

Kevin Ma

Jon Peterson

Shawn Lin

ALTO Interim Meeting

Dec 11, 2018

Overview of Updates

- Changes since IETF 102:
 - Revise the wording (**The whole document**)
 - Simplify some descriptions
 - Fix some typos and grammar issues
 - Re-organize the error handling process to make it clear (**Sec 5.6**)
 - The changes are already proposed during IETF 102
 - Add some examples for better illustration (**Sec 3.6**)
 - Complete the security considerations (**Sec 8**)

Current Status of Remaining Issues

- Two issues reported by WG.
- Issue 1: The motivation of filtering on both capabilities and footprints is missing.
 - The proposal of those two kinds of filtering is just to make the protocol complete. But currently, authors have not figured out a strong motivating example / use case.
 - Hope both ALTO and CDNI WGs to share opinions.
- Issue 2: The current naming for the service data format is not convincing.
 - After some internal discussions, authors have several proposals but no final decision.
 - Will go through the proposals during the meeting.

Unconvincing Naming and Data Format

- This issue comes from both CDNI FCI Map and Filtered CDNI FCI Map:
 - The response data format of current CDNI FCI Map service is not an object-map.
 - The semantics of the current Filtered CDNI FCI Map service is not the map filtering.
 - Fundamentally, the two unconvincing specifications are related.

```
object {  
  CDNIFCIMapData cdni-fci-map;  
} InfoResourceCDNIFCIMap : ResponseEntityBase;
```

```
object {  
  BaseAdvertisementObject capabilities<1..*>;  
} CDNIFCIMapData
```

Let's review the notations defined in RFC7285:

```
object { Type1 name1; Type2 name2<1..*>; [Type3 name3;]  
  } Type4;
```

```
object-map { Type1 -> Type2; } Type3;
```

A Map service in the ALTO framework should map a variable in some type to another variable in some other type.

Information Base v.s. Map

Why is hard to expose CDNI FCI using ALTO Map service?

The key challenge: the data structure of the CDNI FCI advertisement is not a map, but **an Information Base without the Primary Key**.

CDNI Footprints & Capabilities Information Base (FCIB)

capability-type	capability-value	footprints
FCI.DeliveryProtocol	http/1.1	<Footprint objects>
FCI.DeliveryProtocol	http/1.1, https/1.1	<Footprint objects>
FCI.DeliveryProtocol	https/1.1	<Footprint objects>

In an FCIB, any single attribute cannot identify a unique entry, i.e., there is no primary key.

An intuitive idea: introduce the primary key.

Two Solutions to Introduce a Primary Key

- Solution 1: compose <capability-type, capability-value> as a candidate key.
 - **Key insight:** footprints with the same <capability-type, capability-value> can be merged into a single FCIB entry.
 - Similar to the current Filtered CDNI FCI Map design but different.
- Design details:
 - Provide a property map to define the CDNI capability ids.
 - Provide another property map to map each capability id to a sequence of footprint objects.
 - Filter capabilities by id.
- Benefits:
 - No need to introduce any new map service.
 - Can define the inherit rule of CDNI capabilities to make the map compact.

```
"cdnicap:dp:http1.1": {  
  "capability-type": "FCI.DeliveryProtocol",  
  "capability-value": ["http/1.1"]}
```

can inherit all
footprints owned by

```
"cdnicap:dp:http1.1:https1.1": {  
  "capability-type": "FCI.DeliveryProtocol",  
  "capability-value": ["http/1.1", "https/1.1"]}
```

Two Solutions to Introduce a Primary Key

- Solution 2: the FCIB can be re-organized as <footprint-type, footprint-value, capabilities>. The <footprint-type, footprint-value> is a candidate key.
 - **Key insight:** The composition of <footprint-type, footprint-value> can identify a unique footprint entity.
 - footprint-type -> entity-domain
 - footprint-value -> domain-specific-entity-address
 - capabilities -> property
 - Filter footprints by entity addresses.
 - Refer to the current FCI Property Map design.

Open Question: RDB Query v.s. Map Filtering

Limitation of the Map Filtering: Cannot filter FCIB entries by **both** capabilities and footprints **in a single request**. (But do we have any strong motivation to do this?)

Example: Filtering all information like {capability-type == "X" and footprints in Y} is not available.

Another potential solution: Consider the CDNI FCIB as a RDB Table, we can allow the dCDN to use the RDB Query Language (SQL) to pull the compact information from uCDN. (But it is more complex and may break the ALTO framework.)

Which one is better?

Next Steps

- Fix all the remaining issues
- Check the latest SSE and Unified Property documents to see if any updates are required
- Call for reviews

Backup Slides