

Network Working Group
Internet-Draft
Updates: [8006](#), 8008 (if approved)
Intended status: Standards Track
Expires: January 3, 2022

G. Goldstein
W. Power
Lumen Technologies
G. Bichot
Broadpeak
A. Sioniz
Telefonica
July 2, 2021

CDNI Metadata Model Extensions
draft-goldstein-cdni-metadata-model-extensions-00

Abstract

Open Caching architecture is a use case of Content Delivery Networks Interconnection (CDNI) in which the commercial Content Delivery Network (CDN) is the upstream CDN (uCDN) and the ISP caching layer serves as the downstream CDN (dCDN). This document proposed extensions to the [[RFC8006](#)] Metadata Model by way of a set of GenericMetadata objects that address extend the original CDNI capabilities to meet the more general needs of the CDN and Open Caching industry.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 3, 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents

(<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

<u>1.</u>	<u>Introduction</u>	<u>3</u>
<u>1.1.</u>	<u>Terminology</u>	<u>4</u>
<u>1.2.</u>	<u>Requirements Language</u>	<u>4</u>
<u>2.</u>	<u>CDNI Additional GenericMetadata Objects</u>	<u>4</u>
<u>2.1.</u>	<u>AllowCompress</u>	<u>5</u>
<u>2.2.</u>	<u>CachePolicy</u>	<u>5</u>
<u>2.3.</u>	<u>ComputedCacheKey</u>	<u>5</u>
<u>2.4.</u>	<u>CrossoriginPolicy</u>	<u>5</u>
<u>2.5.</u>	<u>NegativeCacheKey</u>	<u>5</u>
<u>2.6.</u>	<u>CacheBypassPolicy</u>	<u>5</u>
<u>2.7.</u>	<u>OcnSelection</u>	<u>5</u>
<u>2.8.</u>	<u>PrivateFeatureList</u>	<u>6</u>
<u>2.9.</u>	<u>ProcessingStages</u>	<u>6</u>
<u>2.10.</u>	<u>RequestedCapacityLimits</u>	<u>6</u>
<u>2.11.</u>	<u>RequestRouting</u>	<u>7</u>
<u>2.12.</u>	<u>ServiceIDs</u>	<u>7</u>
<u>2.13.</u>	<u>SourceMetadataExtended</u>	<u>7</u>
<u>2.14.</u>	<u>StaleContentCachePolicy</u>	<u>7</u>
<u>2.15.</u>	<u>TrafficType</u>	<u>7</u>
<u>3.</u>	<u>CDNI Additional FCI Objects</u>	<u>7</u>
<u>3.1.</u>	<u>FCI.ProcessingStages</u>	<u>8</u>
<u>3.2.</u>	<u>FCI.SourceMetadataExtended</u>	<u>8</u>
<u>3.3.</u>	<u>FCI.RequestRouting</u>	<u>8</u>
<u>3.4.</u>	<u>FCI.PrivateFeatures</u>	<u>8</u>
<u>3.5.</u>	<u>FCI.OcnSelection</u>	<u>8</u>
<u>4.</u>	<u>Metadata Expression Language</u>	<u>8</u>
<u>5.</u>	<u>IANA Considerations</u>	<u>8</u>
<u>5.1.</u>	<u>CDNI Payload Types</u>	<u>9</u>
<u>6.</u>	<u>Security Considerations</u>	<u>9</u>
<u>7.</u>	<u>Acknowledgements</u>	<u>9</u>
<u>8.</u>	<u>References</u>	<u>9</u>
<u>8.1.</u>	<u>Normative References</u>	<u>9</u>
<u>8.2.</u>	<u>Informative References</u>	<u>10</u>
	<u>Authors' Addresses</u>	<u>10</u>

1. Introduction

The Streaming Video Alliance [[SVA](#)] is a global association that works to solve streaming video challenges in an effort to improve end-user experience and adoption. The Open Caching Working Group [[OCWG](#)] of the Streaming Video Alliance [[SVA](#)] is focused on the delegation of video delivery requests from commercial CDNs to a caching layer at the ISP's network. Open Caching architecture is a specific use case of CDNI where the commercial CDN is the upstream CDN (uCDN) and the ISP caching layer is the downstream CDN (dCDN). The interchange of content delivery configuration metadata between the various entities in the delivery ecosystem is essential for efficient interoperability. The need for an industry-standard API and metadata model becomes increasingly important as content and service providers automate more of their operations, and as technologies, such as open caching, require coordination of content delivery configurations. In order to achieve this, the Open Caching Configuration Interface Specification [[OC-CI](#)] defines an interface contemplating a set of use cases.

The following capabilities extend the [[RFC8006](#)] Metadata Model:

- o Enhanced Source definitions, with load balancing, failover, and extended authorization methods
- o A rich set of Cache Control Policies and computed cache keys
- o Rules for Dynamic CORS Headers
- o Traffic Types
- o ServiceID Metadata
- o Processing Stage Rules, enabling metadata to be applied conditionally at various stages in the CDN request/response pipeline.
- o Request URI Rewrites
- o HTTP Header Modifications
- o HTTP Status Modifications
- o Synthetic HTTP Responses
- o An Expression Language for matching rules and synthesis of dynamic values

- o Open Caching Configuration Metadata
- o Private Features

For consistency with other CDNI documents this document follows the CDNI convention of uCDN (upstream CDN) and dCDN (downstream CDN) to represent the commercial CDN and ISP caching layer respectively.

This document defines and registers CDNI GenericMetadata objects (as defined in [section 4 of \[RFC8006\]](#)), registers additional CDNI Payload Types ([section 7.1 of \[RFC8006\]](#)), and adds capability objects (extending [section 5 in \[RFC8008\]](#))

1.1. Terminology

The following terms are used throughout this document:

- o CDN - Content Delivery Network

Additionally, this document reuses the terminology defined in [\[RFC6707\]](#), [\[RFC7336\]](#), [\[RFC8006\]](#), [\[RFC8007\]](#), [\[RFC8008\]](#), and [\[RFC8804\]](#). Specifically, we use the following CDNI acronyms:

- o uCDN, dCDN - Upstream CDN and Downstream CDN respectively (see [\[RFC7336\]](#))

1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14 \[RFC2119\]](#) [\[RFC8174\]](#) when, and only when, they appear in all capitals, as shown here.

2. CDNI Additional GenericMetadata Objects

[Section 4 of \[RFC8006\]](#) defines a set of GenericMetadata Object Types. Below we define additional GenericMetadata Objects.

Note: In the following sections, the term "mandatory-to-specify" is used to convey which properties MUST be included when serializing a given capability object. When mandatory-to-specify is defined as "Yes" for an individual property, it means that if the object containing that property is included in a message, then the mandatory-to-specify property MUST also be included.

2.1. AllowCompress

AllowCompress is a new GenericMetadata object that allows the dCDN to compress content before sending to the client.

2.2. CachePolicy

CachePolicy is a new GenericMetadata object that allows for the uCDN to specify internal caching policies for the dCDN and external caching policies advertised to clients of the dCDN.

2.3. ComputedCacheKey

ComputedCacheKey is a new GenericMetadata object that allows for the specification of a cache key using the metadata expression language. Typical use cases would involve the construction of a cache key from one or more elements of the HTTP request. In cases where both the ComputedCacheKey and the Cache object are applied, the ComputedCacheKey will take precedence.

2.4. CrossoriginPolicy

CrossoriginPolicy allows for the specification of dynamically generated CORS headers.

2.5. NegativeCacheKey

NegativeCachePolicy is a new GenericMetadata object that allows for the specification of caching policies based on error response codes received from the origin.

2.6. CacheBypassPolicy

CacheBypassPolicy is a new GenericMetadata object that allows a client request to be set as non cacheable. It is expected that this feature will be used to allow clients to bypass cache when testing the uCDN fill path. Note, CacheBypassPolicy only applies to the current request. In addition, any content previously cached (by client requests that do not set CacheBypassPolicy) is not evicted.

2.7. OcnSelection

A new GenericMetadata object that allows the uCDN to indicate to the dCDN a preference in terms of OCN selection.

2.8. PrivateFeatureList

A GenericMetadata configuration object as a base generic object that permits the control of private features.

2.9. ProcessingStages

A ProcessingStages object is a type of GenericMetadata that describes the matching rules, metadata, and transformations to be applied at specific stages in the request processing pipeline.

It is typical in CDN configurations to define matching rules and metadata that are to be applied at specific stages in the request processing pipeline. For example, it may be required to append a host header prior to forwarding a request to an origin, or modify the response returned from an origin prior to storing in the cache. The following four processing stages are defined:

- o clientRequest - Rules run on the client request prior to further processing.
- o originRequest - Rules run prior to making a request to the origin.
- o originResponse - Rules run after a response is received from the origin and before being placed in the cache.
- o clientResponse - Rules run prior to sending the response to the client. If the response is from the cache, rules are applied to the response retrieved from the cache prior to sending to the client.

Each of the four processing stages is represented by an array of StageRules objects, with each StageRules object defining match criteria along with metadata that should be applied if the match applies to true. It should be noted that all of the StageRules objects in the array are evaluated and processed in order. A possible future extension to this processing model could allow for an if-else structure, where processing for a stage is halted upon matching of a StageRule match expression.

2.10. RequestedCapacityLimits

MI.RequestedCapacityLimits is a new GenericMetadata object that allows the uCDN to communicate to the dCDN a desired change in the advertised traffic delegation limits for a given host and footprint.

2.11. RequestRouting

MI.RequestRouting is a new GenericMetadata object that allows the uCDN to force the dCDN request routing mode(s) to be applied when working in iterative redirection mode. The list of redirection modes supported by the dCDN is advertised through the FCI.RedirectionMode object. The list of request routing modes supported by the dCDN is advertised through the FCI.RequestRoutingMode object.

2.12. ServiceIDs

MI.ServiceIDs is a new GenericMetadata object that allows for the specification of two tiers of CDN-specific identifiers and names. The interpretation of these identifiers is implementation specific.

2.13. SourceMetadataExtended

SourceMetadataExtended is an alternative to the CDNI standard SourceMetadata object, which adds a property to specify load balancing across multiple sources, as well as a SourceExtended sub-object with additional attributes to the CDNI standard Source object. While both SourceMetadataExtended and SourceMetadata can be provided for backward compatibility, a dCDN that advertises capability for SourceMetadataExtended will ignore SourceMetadata if both are provided for a given host or path match.

2.14. StaleContentCachePolicy

StaleContentCachePolicy is a new GenericMetadata object that allows the uCDN to specify the policy to use by a dCDN when responding with stale content.

2.15. TrafficType

TrafficType metadata defines a set of descriptors that characterize either the type or usage of the traffic, enabling CDNs and OCNs to apply any internal configuration rules without exposing an unnecessary amount of internal details.

3. CDNI Additonal FCI Objects

[Section 5 of \[RFC8008\]](#) describes the FCI Capability Advertisement Object, which includes a CDNI Capability Object as well as the capability object type (a CDNI Payload Type). The section also defines the Capability Objects per such type. Below we define additional Capability Objects.

Note: In the following sections, the term "mandatory-to-specify" is used to convey which properties MUST be included when serializing a given capability object. When mandatory-to-specify is defined as "Yes" for an individual property, it means that if the object containing that property is included in an FCI message, then the mandatory-to-specify property MUST also be included.

3.1. FCI.ProcessingStages

This object is used to signal the set of features that are supported in relation with the ProcessingStages configuration object. Those optional features depend on the CDNI-MEL language support.

3.2. FCI.SourceMetadataExtended

This object is used to signal the supported features related to the SourceMetadataExtended configuration object.

3.3. FCI.RequestRouting

This object is used by the dCDN to signal/announce the supported request routing modes. This can be optionally used by the uCDN to further select a subset of those modes when operating one of the iterative delegation modes.

3.4. FCI.PrivateFeatures

This object is used by the dCDN to signal/announce the list of supported private features.

3.5. FCI.OcnSelection

This object is used by the dCDN to signal/announce the supported OCN types and/or their transport arrangement and/or medium supported by OCNs.

4. Metadata Expression Language

The CDNI Metadata Expression Language provides a syntax with a rich set of variables, operators, and built-in functions to facilitate use cases within the extended CDNI metadata model. Details TBD

5. IANA Considerations

5.1. CDNI Payload Types

TBD. Two new auth-types will be introduced for use with SourceMetadataExtended: AWSv4Auth and HeaderAuth

6. Security Considerations

This specification is in accordance with the CDNI Request Routing: Footprint and Capabilities Semantics. As such, it is subject to the security and privacy considerations as defined in [Section 8 of \[RFC8006\]](#) and in [Section 7 of \[RFC8008\]](#) respectively.

MORE - TBD

7. Acknowledgements

The authors would like to express their gratitude to the members of the Streaming Video Alliance Open Caching Working Group for their guidance / contribution / reviews ...)

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8006] Niven-Jenkins, B., Murray, R., Caulfield, M., and K. Ma, "Content Delivery Network Interconnection (CDNI) Metadata", [RFC 8006](#), DOI 10.17487/RFC8006, December 2016, <<https://www.rfc-editor.org/info/rfc8006>>.
- [RFC8007] Murray, R. and B. Niven-Jenkins, "Content Delivery Network Interconnection (CDNI) Control Interface / Triggers", [RFC 8007](#), DOI 10.17487/RFC8007, December 2016, <<https://www.rfc-editor.org/info/rfc8007>>.
- [RFC8008] Seedorf, J., Peterson, J., Previdi, S., van Brandenburg, R., and K. Ma, "Content Delivery Network Interconnection (CDNI) Request Routing: Footprint and Capabilities Semantics", [RFC 8008](#), DOI 10.17487/RFC8008, December 2016, <<https://www.rfc-editor.org/info/rfc8008>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8804] Finkelman, O. and S. Mishra, "Content Delivery Network Interconnection (CDNI) Request Routing Extensions", [RFC 8804](#), DOI 10.17487/RFC8804, September 2020, <<https://www.rfc-editor.org/info/rfc8804>>.

8.2. Informative References

- [OC-CI] Goldstein, G., Ed., Power, W., Bichot, G., and A. Sioniz, "Open Caching - Configuration Interface Functional Specification (Parts 1,2,3)", Version 0.1, July 2021.
- [OCWG] "Open Caching Home Page", <<https://www.streamingvideoalliance.org/technical-groups/open-caching/>>.
- [RFC6707] Niven-Jenkins, B., Le Faucheur, F., and N. Bitar, "Content Distribution Network Interconnection (CDNI) Problem Statement", [RFC 6707](#), DOI 10.17487/RFC6707, September 2012, <<https://www.rfc-editor.org/info/rfc6707>>.
- [RFC7336] Peterson, L., Davie, B., and R. van Brandenburg, Ed., "Framework for Content Distribution Network Interconnection (CDNI)", [RFC 7336](#), DOI 10.17487/RFC7336, August 2014, <<https://www.rfc-editor.org/info/rfc7336>>.
- [SVA] "Streaming Video Alliance Home Page", <<https://www.streamingvideoalliance.org>>.

Authors' Addresses

Glenn Goldstein
Lumen Technologies
US

Email:
glennng1215@gmail.com

Will Power
Lumen Technologies
US

Email:
wrpower@gmail.com

Guillaume Bichot
Broadpeak
France

Email:

guillaume.bichot@gmail.com

Alfonzo Siloniz
Telefonica
Spain

Email:

alfonsosiloniz@gmail.com

