Slide Title

- Abstract
- MI.CrossoriginPolicy
- MI.AllowCompress
- MI.ClientConnectionControl
Abstract

- This draft defines MI.GenericMetadata objects related to controlling edge access to resources via content delivery networks (CDNs) and Open Caching systems.
- Configuring Cross-Origin Resource Sharing (CORS) access rules and the dynamic generation of CORS headers is a key feature of typical configurations, as are the ability to define response body compression rules and client connection timeouts.
- This objects cover functionalities that represent day-by-day operations in configurations between Content Providers and CDNs like:
  - configuring Cross-Origin Resource Sharing (CORS) access rules and the dynamic generation of CORS headers
  - response body compression rules in the edge
  - client connection timeouts
Changes from version 00

- This draft no longer updates RFC 8006. It conforms a new group of Generic Metadata objects.
- MI.TrafficType MI.GenericMetadata object has been removed from this draft. This object will be included in a future draft.
- Many style and typographic errors have been corrected.
• When delivering video content is typical that Videoservice platforms use different domain realms for video portal that for video delivery (more event if using a third-party CDN)
• In such cases CORS(*) mechanism implemented in browsers and devices are in use, making necessary to generate Access-Control-Allow-* headers in the responses according to a predefined logic to authorise those Cross-domain requests
• Without any other mechanism, a cache system or Surrogate acting on behalf of the Content Provider Source should validate those requests based on the Origin header received in the request
• It is a typical use case in current CDNs to generate Synthetic responses in the edge servers, based on logic that permit to implement the policy a Content Provider want to use for CORS responses
• The proposed object is design to permit configuring those policies to a downstream CDN to generate those headers and synthetic responses without needing to ask the CP source.

* https://www.w3.org/TR/2020/SPSD-cors-20200602/
• Simple CORS configuration
  – Only controlling Access-Control-Allow-Origin header in responses
  – Permit the usage of wildcard in the response

• Advanced CORS configuration
  – Managing all the rest of Access-Control-Allow-* headers in responses
  – Generation of synthetic responses for preflight requests (CORS)

```json
{  
  "generic-metadata-type": "MI.CrossoriginPolicy",
  "generic-metadata-value": {
    "allow-origin": {
      "allow-list": [
        {  
          "pattern": "www.example.com"
        }
      ],
      "wildcard-return": true
    }
  }
}

{  
  "generic-metadata-type": "MI.CrossoriginPolicy",
  "generic-metadata-value": {
    "allow-origin": {
      "allow-list": [
        {  
          "pattern": "*:sourcepage.example.com"
        },
        "wildcard-return": false
      ],
      "allow-methods": [  
        "GET", "POST"
      ],
      "allow-credentials": true,
      "allow-headers": [  
        "X-PINGOTHER", "Content-Type"
      ],
      "expose-headers": [  
        "X-User", "Authorization"
      ],
      "max-age": 3600
    }
  }
}
```
MI.AllowCompress is a new GenericMetadata object that allows the dCDN to compress content before sending to the client.

Thus, the uCDN source can send only one copy of one object, being cached in the edge, while the edge server handles the different options based on the Accept-Encoding header the UA is using in the request.

It is expected this object is used along with other proposed MI objects as MI.HeaderTransform to restrict this functionality to specific object types as DASH manifests or HLS playlists, as using compression in the video segments brings no advantage.

```json
{
    "match": {
        "expression": "req.h.uri *= '*.m3u8'"
    },
    "stage-metadata": {
        "generic-metadata": [
            {
                "generic-metadata-type": "MI.AllowCompress",
                "generic-metadata-value": {
                    "allow-compress": "true"
                }
            }
        ]
    }
}
```
• MI.ClientConnectionControl is a new GenericMetadata object that specifies how a dCDN manages its connections to clients/players.
• Configuration metadata is required to define how connections against a client are maintained by a dCDN. Since the clients are typically owned/operated by a uCDN, giving this control to the uCDN allows it to accommodate device specific constraints and performance optimizations. A dCDN can also benefit from this configuration metadata to meet its security and resource consumption requirements.

```json
{
  "generic-metadata-type": "MI.ClientConnectionControl",
  "generic-metadata-value": {
    "connection-keep-alive-time-ms": 3
  }
}
```
Next steps

• Please, review the draft and provide any comments to the mailing list or the authors
• The authors ask for adoption of this draft in the CDN-I WG
Thank you