

CDNI Control Interface / Triggers - 2nd Edition (RFC 8007bis) IETF 121

Sanjay Mishra/Nir Sopher/Jay Robertson/Alan Arolovitch
Dublin, November 8th, 2024

RFC 8007bis v15: What changed from v14?

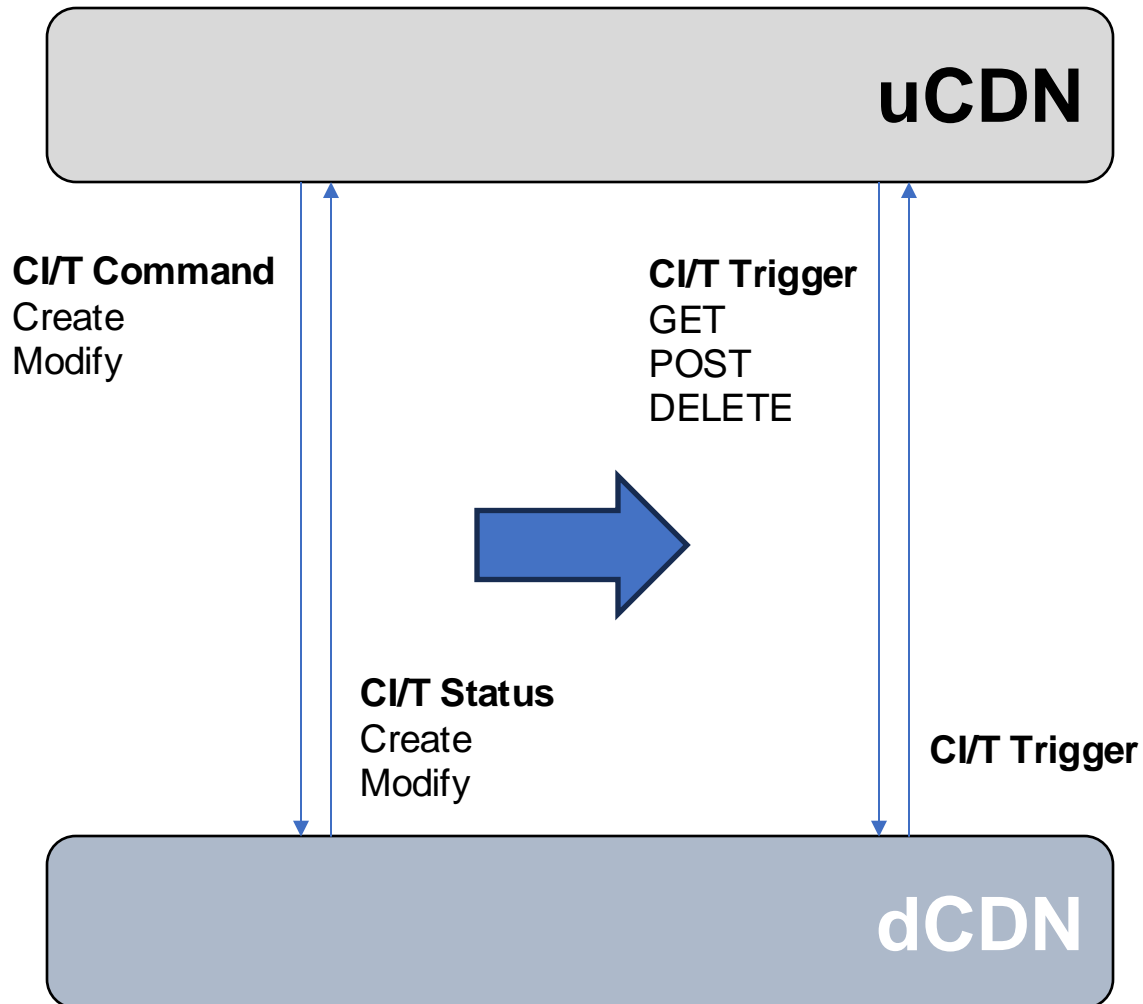
- **Published v15 on October 21st 2024: Initial RESTful draft**
 - <https://datatracker.ietf.org/doc/html/draft-ietf-cdni-ci-triggers-rfc8007bis-15>
- **What changed?**
 - REST: Unified trigger specs, actions, states, labels, errors into ci-trigger.v2
 - REST: Unified trigger objects, filtered collections into ci-trigger-collection.v2
 - Streamlining the document structure for better flow
 - Reformatting and syntactic clean-up of JSON examples for readability
 - Purge/invalidation and acquisition in progress
- **A clean-up v16 is already in progress**
- A diff from the v14 is available at:
 - <https://author-tools.ietf.org/iddiff?url1=draft-ietf-cdni-ci-triggers-rfc8007bis-14&url2=draft-ietf-cdni-ci-triggers-rfc8007bis-15>

Summary of Changes in v15:

Alan to present on the changes made in v15:

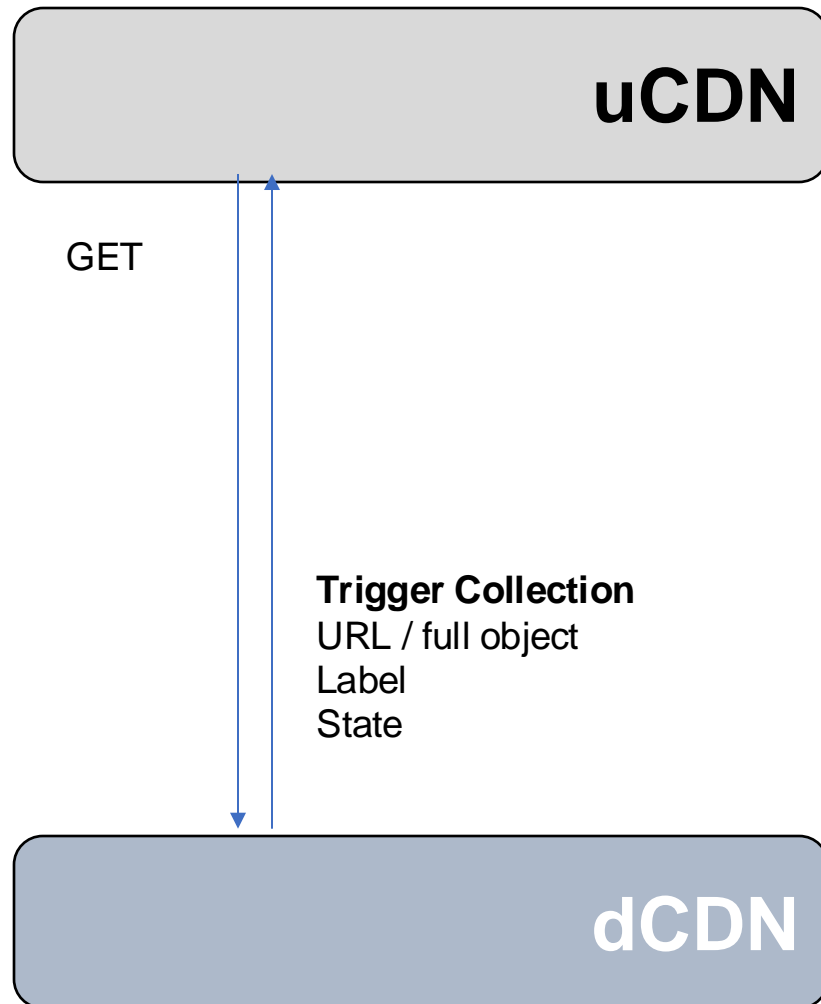
- REST: Trigger Object
- REST: Trigger Collection Object
- Streamlining the document structure for better flow
- Reformatting and syntactic clean-up of JSON
- Purge/invalidation and acquisition in progress

REST: Trigger Object



- uCDN and dCDN exchange representations of one C/I/T Trigger object (ci-trigger.v2)
- dCDN can support C/I/Tv1 and C/I/Tv2 in one interface
- HTTP methods convey operations
- Merge trigger spec, trigger action, extensions, state, labels, errors, CDN path into the Trigger object
- Trigger state instead of status
- Document language cleanup

REST: Trigger Collection Object



- One object created and managed by dCDN that represents the plurality of trigger objects, updated with trigger resource changes
- Filtered Collections became multiple representations of Trigger Collection object
- Triggers encoded as resources URLs or as full objects
- Filtered by trigger state
- Filtered by trigger label

Streamlined Document Structure

- Introduction
- Architecture
- Interface
- Object Properties and Encoding
- Footprint and Capabilities
- Examples
- IANA: MIME types and object type registries
- Security
- References

JSON fixes

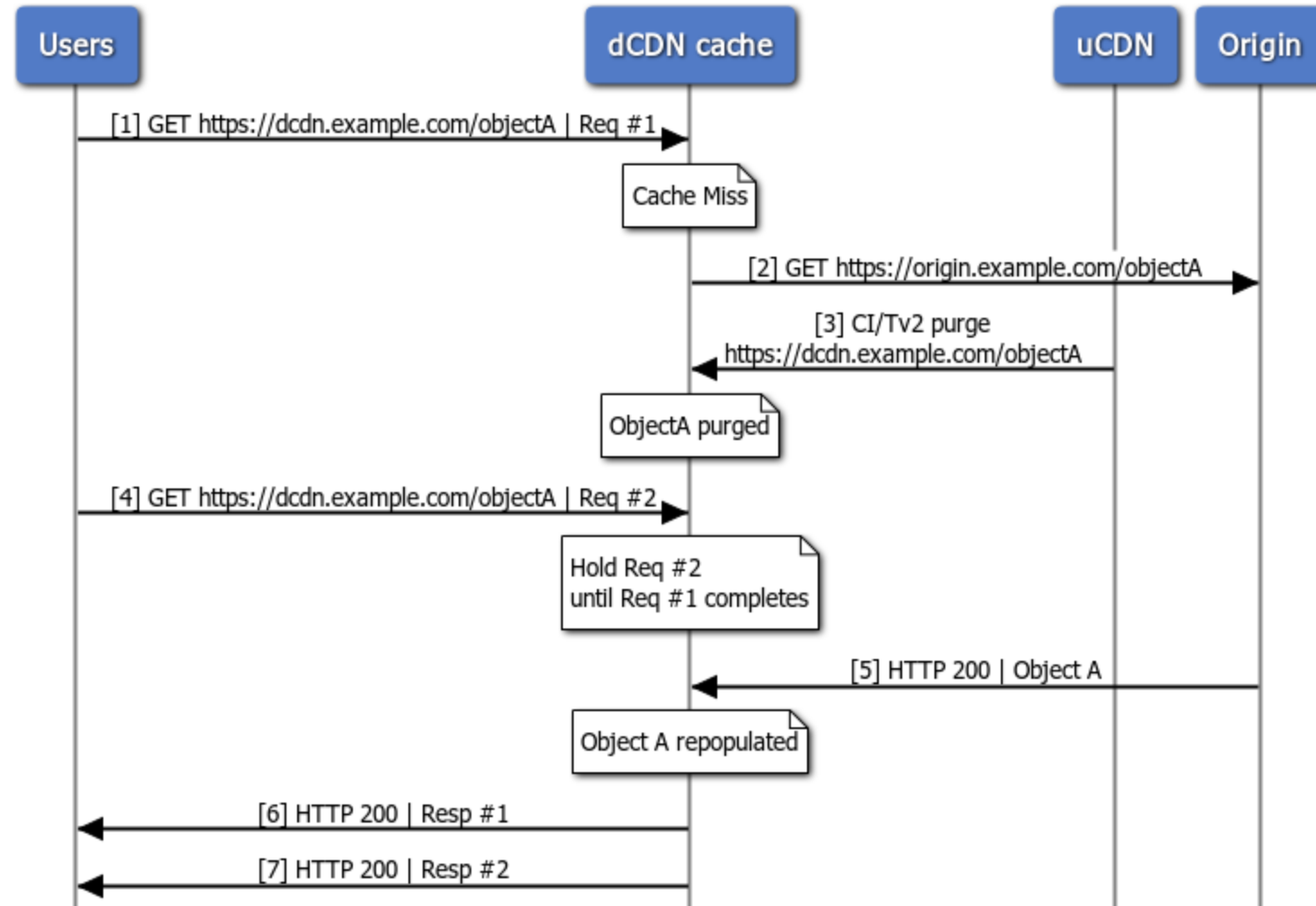
- All JSON snippets and examples pass JSON validation and have a common format
- ObjectList fix in “content-objectlist” (key-value pair as spec value)

```
{
  "action": "preposition",
  "specs": [{
    "trigger-subject": "content",
    "generic-trigger-spec-type": "content-objectlist",
    "generic-trigger-spec-value": {
      "objects": [{
        "href": "https://www.example.com/hls/title/index.m3u8",
        "type": "hls"
      }]
    }
  ]},
  "cdn-path": [ "AS64496:0" ]
}
```

Purge/invalidation and Acquisition in progress

v14

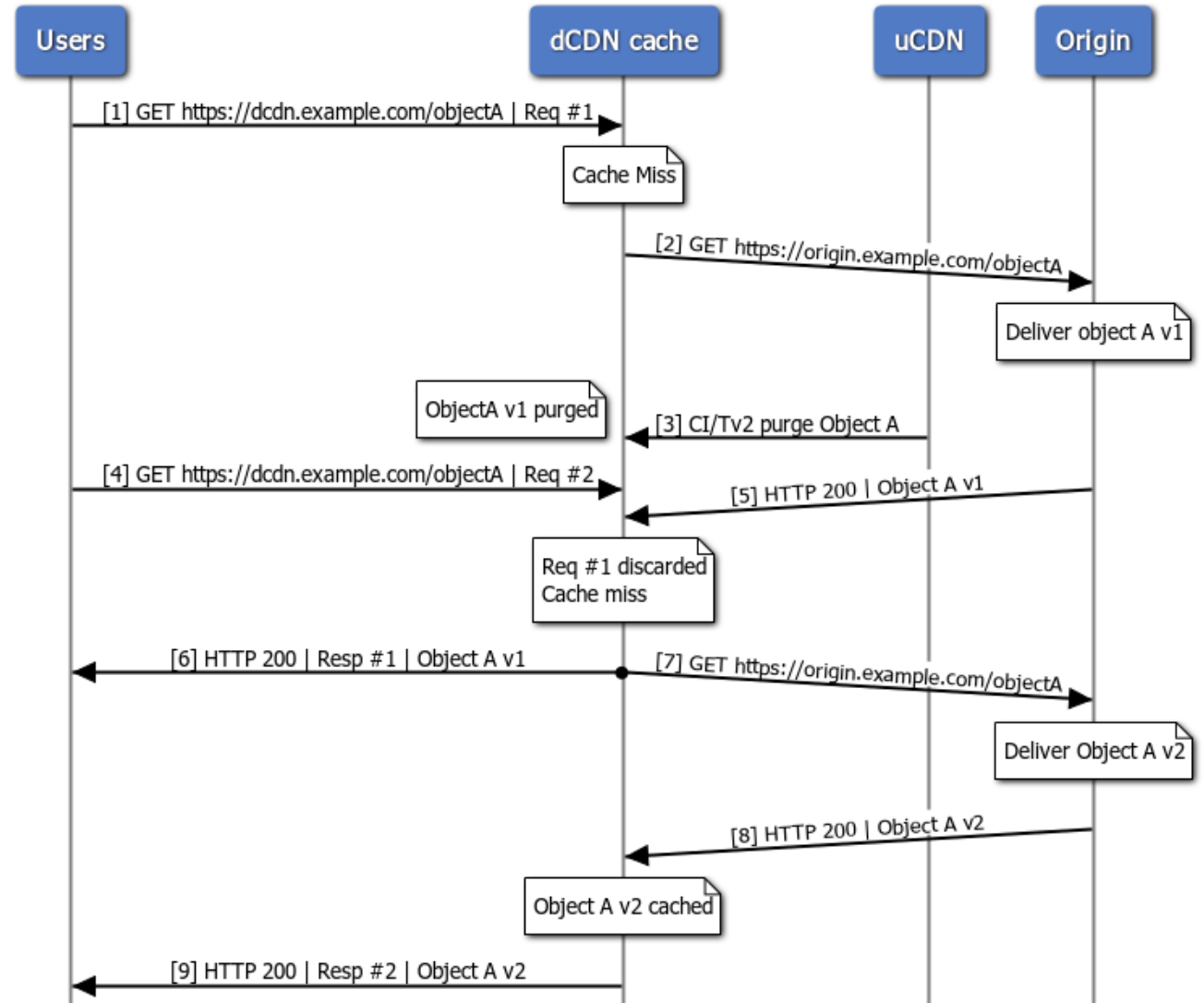
Because the CI/T Command timing is under dCDN's control, dCDN implementation can choose whether to apply CI/T "invalidate" and "purge" commands to content acquisition that has already started when the command is received.



Purge/invalidation and Acquisition in progress

v15

dCDN implementation SHOULD apply "invalidate" and "purge" triggers to content acquisition that is in progress when the trigger becomes active, to avoid placing purged or invalidated content into cache upon completion of the content acquisition



RFC 8007bis v16: Under development

- **To be completed and published after IETF 121**
- **What is changing?**
 - Address issues found in internal review of v15 (spelling, grammatical, etc.)
 - Incorporate review feedback from Brett Mertens: trigger spec constraints chapter and numerous miscellaneous fixes
 - If-Modified-Since / Last-Modified conditional requests and examples
 - Improved format of URLs in examples for clarity/consistency (use of UUIDv4 in examples, dCDN and uCDN domain separation)
 - Strict requirement for dCDN to match trigger with uCDN configuration
 - **Review “legacy” areas around CDN cascading, interconnected CDNs**
 - **Extended examples**
 - **Complete refactoring of filtered collections**
 - **Incorporate comments from subsequent reviews**

What's outstanding?

1. Finish and publish v16 after IETF 121
2. Need more reviews beyond the RFC authors!
3. Incorporate feedback from reviews

Next steps?

- **More reviews**
- Finish v16 draft
- Working Group Last Call at IETF 122