parc®

A Xerox Company

ICN CACHE CONTROL

Marc Mosko

Palo Alto Research Center

ATIS eCON Presentation (August 11, 2016)

CACHE CONTROL

- In NDN/CCNx, there is a “Content Store”
  - The ContentStore is a standardized cache that operates close to the data plane (forwarder).
  - Allows opportunistic caching of any Data/Content Object, or pre-populated (via unstandardized out-of-band protocol).
  - This presentation covers the cache control directives available to the Content Store.
  - In both NDN and CCNx, the cache control directives only apply to the ContentStore serving an object, not to the “fast path” data plane.

NDN

- Freshness
  - A Data object may contain an optional “FreshnessSeconds” field inside the signature envelope.
    - If missing, Data is always “fresh”.
  - It is a relative time. It is like HTTP “max-age.”
    - A Data object in a ContentStore is considered fresh if it was received less than FreshnessSeconds seconds ago. Otherwise it is stale.
    - A stale Data object is still cachable.
    - Unlike HTTP, a stale Data can be served without origin validation.
  - An Interest may carry a “MustBeFresh” flag
    - Only a “fresh” content store entry can satisfy it.
    - Akin to HTTP client directive “max-stale” (though not the same)

CCNX

- ExpiryTime
  - – Inside a ContentObject's signature envelope, it is an absoluate time after which a ContentStore cannot serve the ContentObject.
  - – Not the same as HTTP "Expires" -- no sense of fresh/stale.
    - One could say before ExpiryTime, it is "alive" and after it is "dead".
- RecommendedCacheTime (RCT)
  - – In the unsigned headers of a ContentObject (i.e. mutable). It is the absolute time after which the producer believes there is little value to continue caching an object.
  - – It is not "alive/dead" or "fresh/stale". It is a recommendation.
  - – It is not an error to continue to cache and serve an object beyond the RCT.
  - – ExpiryTime takes precedence.

RECOMMENDED CACHE TIME

- Expected usage
  - Because the RCT is outside the signature envelope, a producer can create a content object with a long expiry time (e.g. a week) but short RCT (e.g. 5 minutes).
  - This allows the producer to send the same signed object over and over (without re-signing) but still get control over how long it should stay in cache.
  - Using the relative time FreshnessSeconds gives a similar ability, to set a short Freshness (e.g. 5 minutes) and be able to re-issue the same signed object over and over again.

SUMMARY

- NDN
  - There is no mechanism to require a cache stop serving an object.
  - It is permitted to keep serving a “stale” object.
  - A consumer can ask for a fresh object (but that does not remove the stale one).
- CCNx
  - There is a hard deadline ExpiryTime to stop a cache from serving an object. Requires re-signing to change.
  - There is a soft deadline RecommendedCacheTime to remove from cache, not mandatory to remove.

NDN VS CCNX APPROACH

- In our opinion, using relative times that cannot be decremented is the wrong choice for an ICN.
  - In HTTP, relative times are used end-to-end or over cache hierarchies (trees).
  - If the object does not form a cycle in the cache hierarchy, relative is ok.
- In ICN, an object can form a cycle over intermediate nodes.
  - It is easy to construct examples where a Data object with a FreshnessSeconds keeps cycling around the network forever.
- In a Data object response, there is no way to tell Fresh from Stale.
  - If an intermediate node forwards one interest with MustBeFresh and one without (they cannot be aggregated), the intermediate node cannot tell if the response is fresh or stale.
- CCNx requires loosely synchronized time for cache directives.