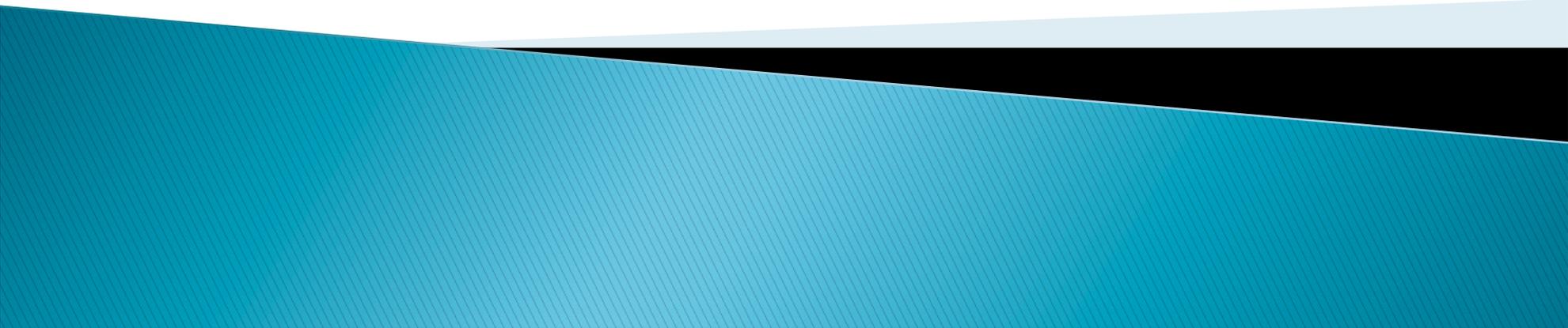# CDNI Framework

draft-davie-cdni-framework-01.txt

A. Falk, B. Davie, L. Peterson et al.

Recap: Objective of Framework

- Address the question "how do all the pieces fit together?"
  - Think of a frame providing the structure to which the other pieces get attached
  - Describes how all the CDNI components and additional out-of-scope components (e.g. inter CDN acquisition, request interface) combine to deliver full CDNI solution
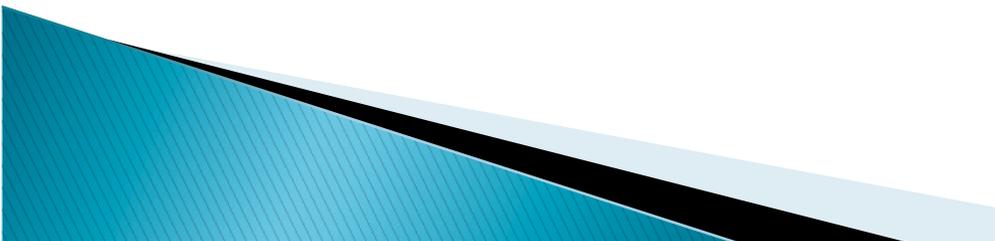
Objective of Framework (2)

- Illustrate key design tradeoffs
  - E.g. HTTP– versus DNS–based redirection
  - E.g. Recursive versus iterative request routing
- Define the functionality of each interface
- Leave details of interface specifications (Request Routing, Control, Metadata, Logging) to other documents

Changes from –00

- New text on the functionality of the Four Interfaces
  - Realized that Framework is likely best place to document this
- Terminology
  - Defined "synchronous" and "asynchronous" operations, and use this in examples
  - Incorporated recursive & iterative request routing
- Security Considerations Expanded
  - Added reference to underlying mechanisms of other protocols (e.g. HTTPS)
 - DRM discussed (considered out of scope)

Changes from –00 (2)

- Request Routing Interface
  - Note that it has 2 parts:
    - the asynchronous advertisement of dCDN capabilities, footprint etc.
    - Synchronous redirection of user requests to a particular dCDN
  - Discuss RRI options (BGP, ALTO, etc.) for async part
- Metadata Interface
  - Discussion added on how MI can be used to control distribution of individual content items or sets of content
    - E.g. geoblocking, availability window
  - Open issue: is removal of content so similar to limiting its availability that removal/purge is a MI function?
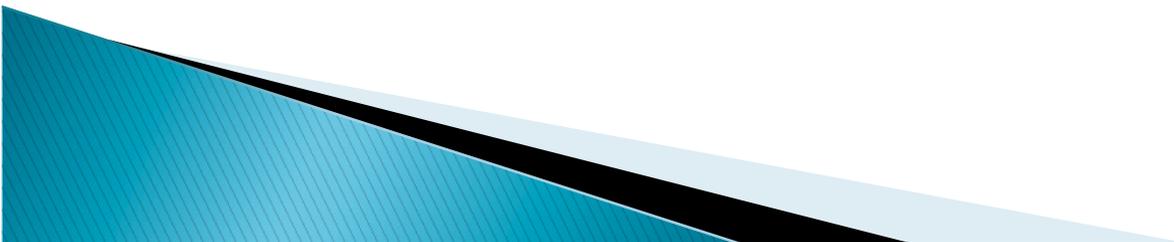
Changes from -00 (3)

- **More commentary after examples**
  - Exploring the design space, not prescriptive
  - Pros/cons of recursive vs. iterative, and of http- vs DNS-based redirection
- **Clarifications in response to list comments**
  - Multiple uCDNs permitted for a single CSP
  - uCDN has the option NOT to redirect to any dCDN
  - Clarify that examples and deployment scenarios are not exhaustive, merely illustrative
  - Clarify that RRI needs to be configured/ bootstrapped
- **Typos**

Open Issues

- Where do "triggers" (like purge, pre-position) live – control or metadata or other?
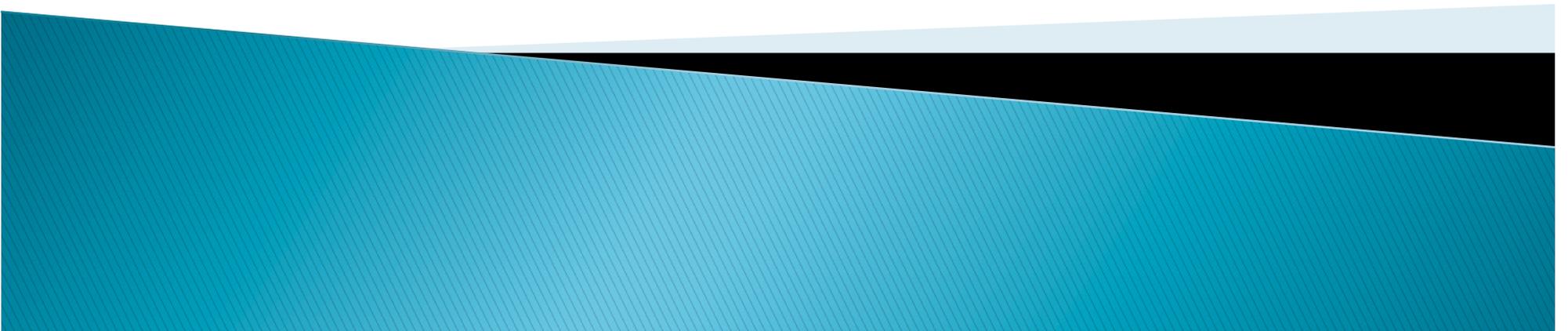- May need to refine terminology regarding "push"

# Next steps

- Discuss and resolve open issues
- WG draft?

Backup

Message Flow examples

▶ Series of examples to illustrate:
  ◦ The "big picture" of how operations proceed to distribute content, metadata and control information
  ◦ Different request routing styles, including recursive and iterative
  ◦ Various ways the Request Routing, Metadata and Control interfaces may be used
  ◦ How certain agreements/conventions between providers may assist interoperation
    • E.g. naming conventions for acquisition nodes
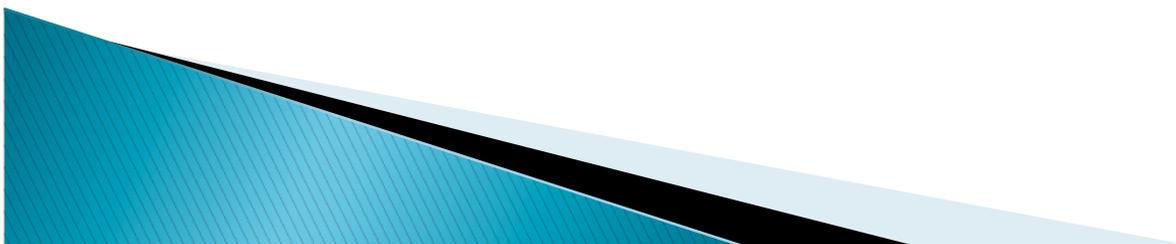
Message Flow Details

▸ Read the draft :-)

▸ It's been noted that we have a lot of detail on domain names, DNS operation, and HTTP redirection

▸ The names illustrate the type of things that will either need to be configured or exchanged in protocols to be defined

▸ The use of DNS- and HTTP-based redirection needs to get documented somewhere

　◦ The request interface is "out of scope" for CDNI in the sense that no new mechanisms are to be defined

Deployment Scenarios

- Illustration of how various deployments may be supported
  - Mesh of CDNs
  - CSP uses CDNI to interact with CDN(s)
  - CDN Exchange
- Not exhaustive
- Show some examples of useful subsetting of the CDNI interfaces

Trust Model and Security Issues

- Identify trust & security issues that are unique to interconnected CDNs
- Key issue: (non-)transitivity of trust
  - CSP trusts uCDN who trusts dCDN but CSP doesn't trust dCDN
  - "Trust but verify" covers some cases
    - E.g. 3rd party monitoring of end-end performance
    - But, if problems are found, may be harder to pinpoint the culprit in a chain of CDNs
  - Detailed interface specs should tackle this
- Single CDN access control methods must also work in CDNI (e.g. URL signing)
- Avoiding open proxy behavior