DECADE Architecture

draft-ietf-decade-arch-00

Richard Alimi, Hongqiang Liu,
Dirk Kutscher, Akbar Rahman,
Richard Yang

DECADE Architecture Objectives (Reminder)

- Architecture for accessing in-network storage
  - Entities and protocol (interfaces)
  - For enhancing P2P and similar distribution systems
  - To be implementable with existing IETF protocols

- Main architecture elements
  - Naming of resources
  - Resource management and authorization
  - Actual data transmission / distribution to servers

Changes since previous version

- Document became a WG draft

- Added more details on:
  - Data sequencing and naming
  - DECADE protocols (DRP and SDT)
  - Server-to-server protocols
  - Evaluation of candidate existing protocols
  - Started by evaluating HTTP

Open Issues (1)

- Does the WG agree on the approach for “Data Sequencing and Naming” (section 4.4)?
  - DECADE derives object names from hashes of the data object content (and the data objects are immutable)
  - Application names will be independent of the DECADE name (and mapping between names will be maintained by the application)

Open Issues (2)

- Does the WG agree on the approach for "DECADE Protocols" (section 5)?
  - DRP provides configuration of access control and resource sharing policies on DECADE servers
  - SDT provides data access interface and is used to read/write objects from a server

Open Issues (3): DRP and SDT

- DRP and SDT conceptually separate
  - Could actually be realized by one protocol
- **How to delegate authorization to application clients?**
  - Client must indicate authorization and resource sharing policy to a DECADE server
- **Approach 1: SDT carries DRP tokens inline**
  - DRP defines format of a token which is carried in an extension field of SDT
  - Tokens generated (by a DECADE client or other trusted entity), which may be used to access its storage at a DECADE server
  - Tokens may be used itself, or given to other DECADE clients
- **Approach 2: Separate protocol**
  - DECADE client indicates policy to DECADE server via an independent protocol (e.g., separate messages)
  - Client periodically sends updates (e.g., update to resource policy) to the server
- Observations:
  - Approach 1 appears simpler in specification and implementation
    - Smaller delay for allowing a peer to first read/write from DECADE storage
  - Approach 2 may provide more room for extensibility (though it isn't clear if we need it)

Open Issues (4): Querying Server Status

- DECADE status may include more than available in SDT (e.g., resource usage by self and authorized peers)
  - This is an exercise in how DRP may be designed to evolve and be extended

- If token-based approach:
  - Option 1: Extend SDT
    - Approach responsible WG's with a proposal
  - Option 2: Allow SDT responses to include DRP metadata
    - Returned metadata includes additional DECADE status
    - May piggyback on data responses too (in addition to SDT status responses)

- If DRP is separate protocol:
  - Option 1: Extend SDT
  - Option 2: DRP defines its own status message
    - Contains status only for DRP-layer status

- Observations
  - There appear to be alternatives that still let DECADE evolve independently in either case

Next Steps for Design Team

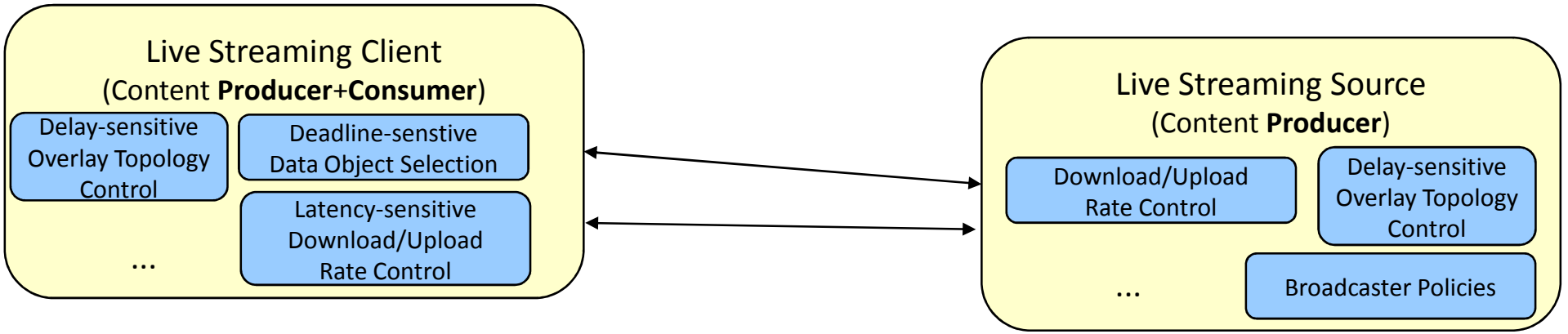
- Address open issues as described
- Concretize naming
- Complete conceptual descriptions of DRP and SDT
- Complete HTTP example implementation

BACKUP

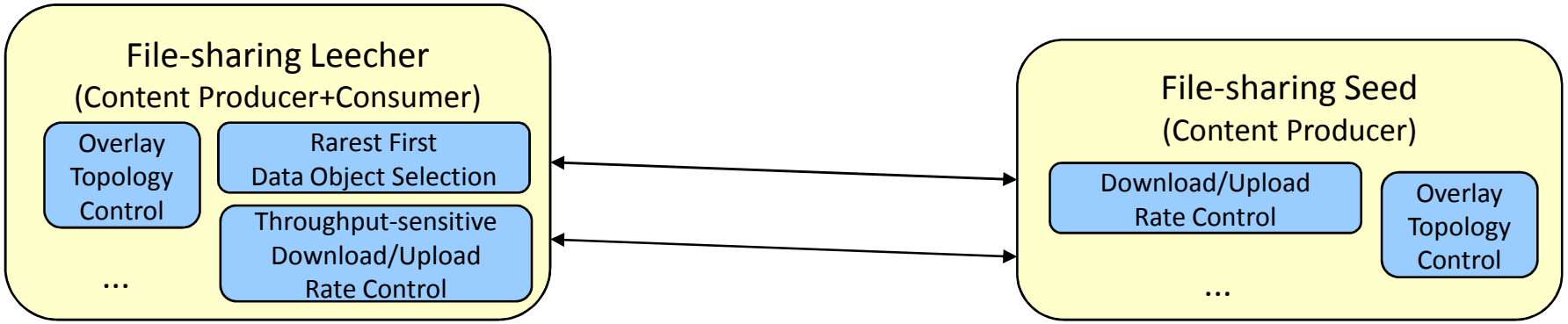
Draft Purpose

- DECADE intended to improve network efficiency of P2P apps
  - ... by introducing storage into the network
- Draft presents one possible architecture for DECADE
  - Satisfies requirements from draft-ietf-decade-reqs
  - Principles drawn from observations from draft-ietf-decade-survey
- Focus on major design issues
  - Explicitly define/describe major design principles
  - Define remaining issues to be resolved
- Feedback much appreciated

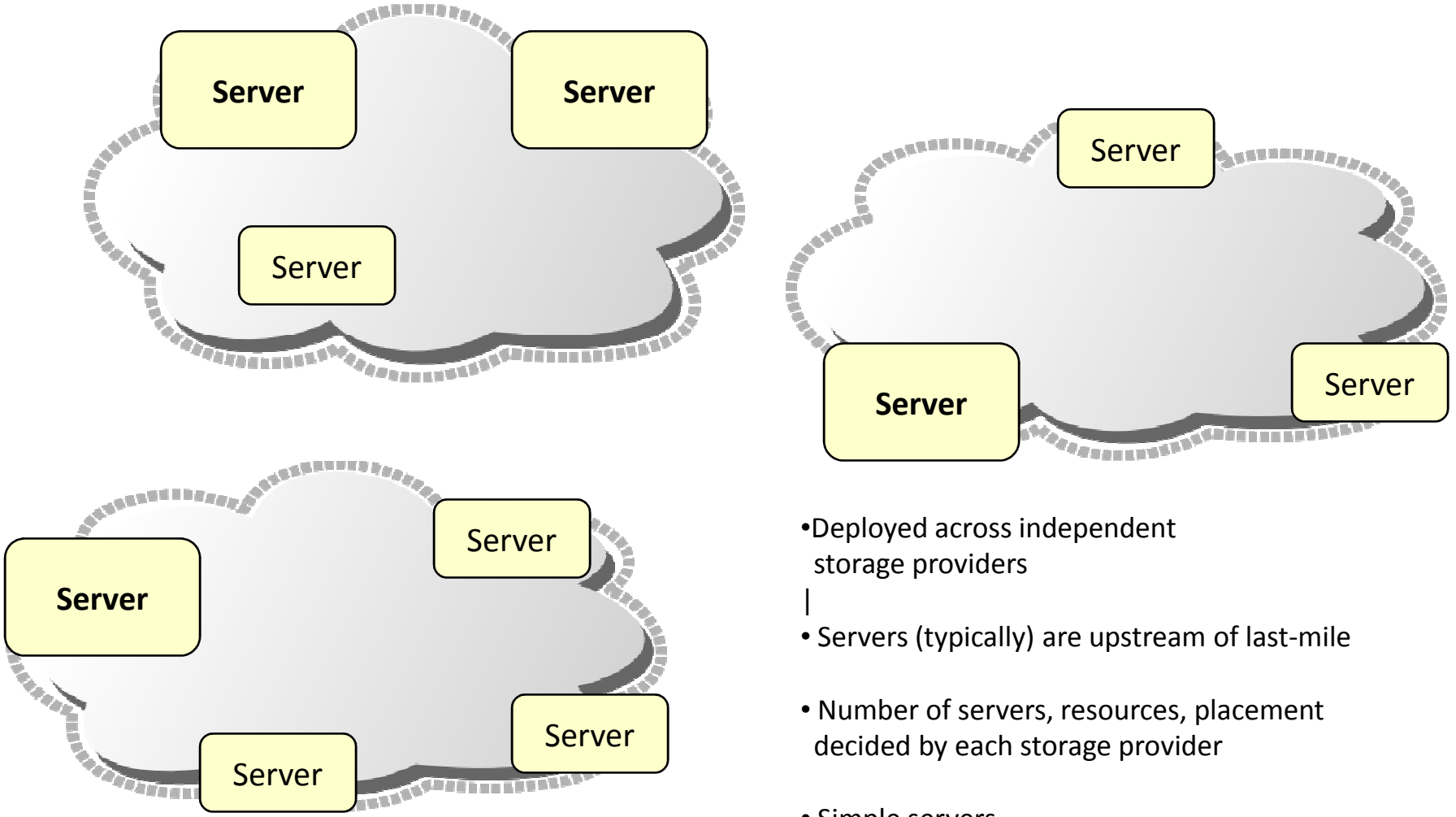
Architectural Entities: Applications



Typical Characteristics:
- Divide content into smaller objects for distribution
- Multiple sources for content

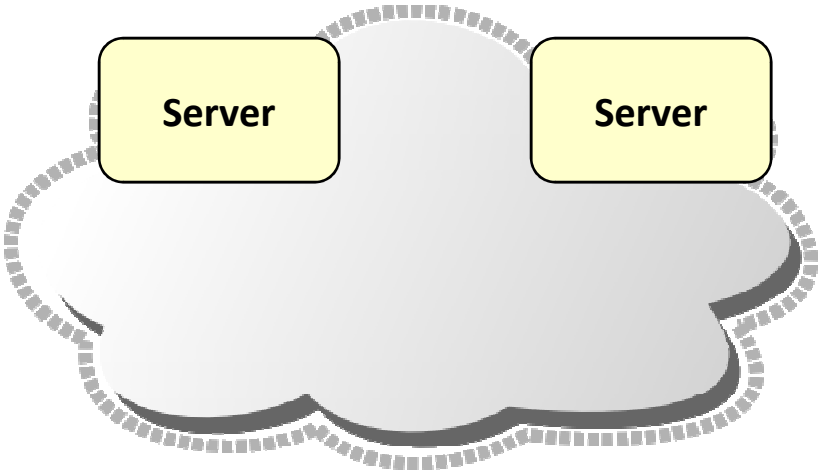


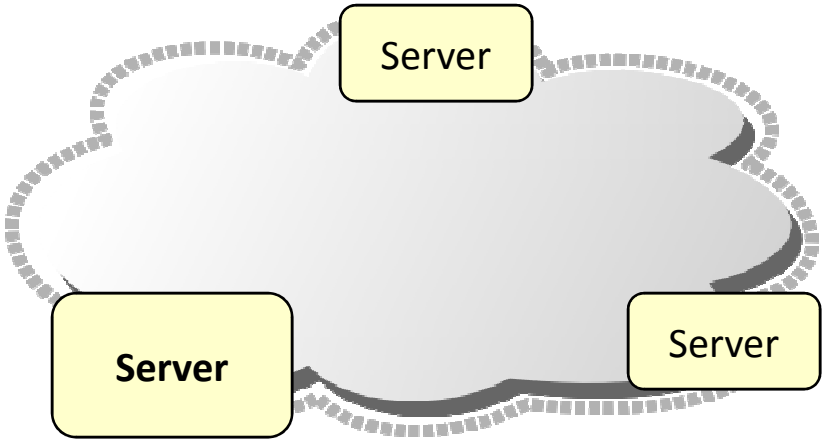
Architectural Entities: DECADE Servers



- Deployed across independent storage providers
- Servers (typically) are upstream of last-mile
- Number of servers, resources, placement decided by each storage provider
- Simple servers
- no distributed coordination (within DECADE)

Architectural Entities





Live Streaming Client
(DECADE Producer+Consumer)

Live Streaming Source
(DECADE Producer)

File-sharing Leecher
(DECADE Producer+Consumer)

File-sharing Seed
(DECADE Producer)