

Towards designing a Network Operator-assisted Information Centric Delivery System

Dionysos Team

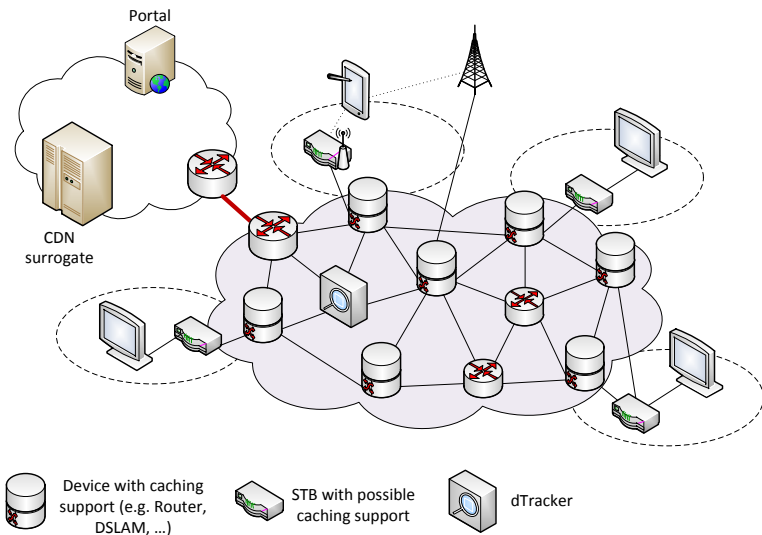
INRIA Rennes, France

Paris, March 2012

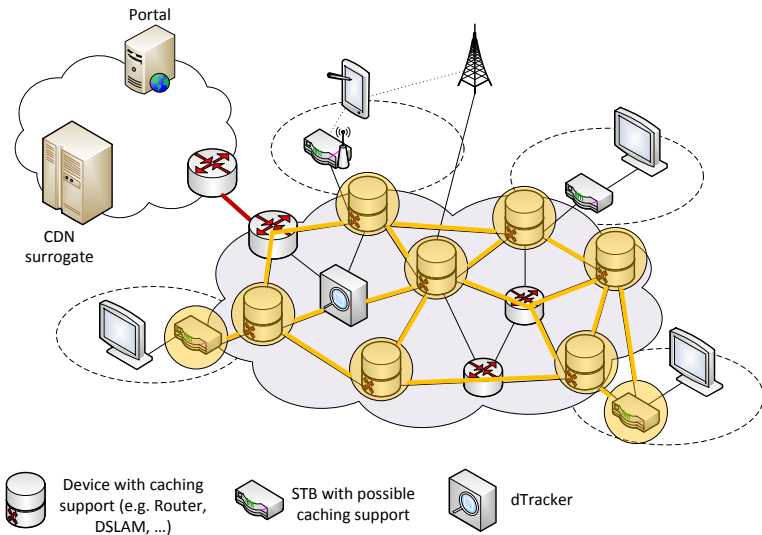
CCN vs. ICN

- CCN directly routes and delivers named pieces of content at the packet level of the network
 - Novel networking paradigm
 - Automatic and application-neutral caching
- CCN is a specific ICN architecture
- Our view:
 - Keep the current network paradigm while naming pieces of content
 - Just like in P2P
 - CCN in the intra-domain

Considered Architecture: Distributed CDN



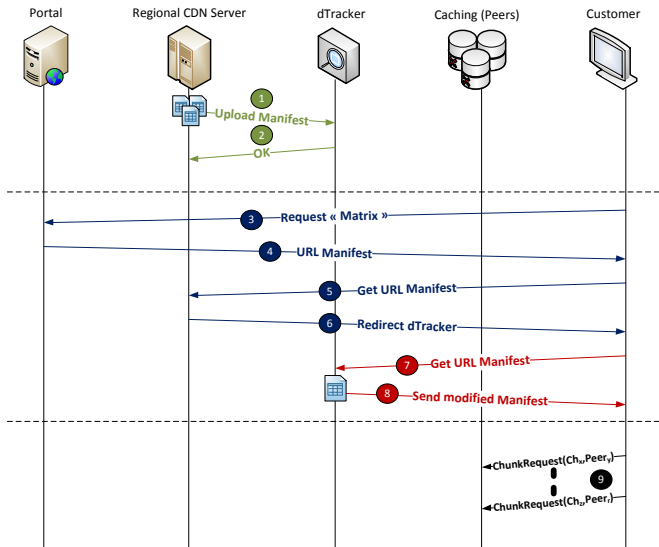
P2P-based caching



P2P vs. P2P-based caching

	P2P	P2P-based caching
Organisation	Self-organised	Managed
Mode	Decentralised	Decentralized/Centralised
Bandwidth	Limited (upload)	Large
Cache size	Moderate	Limited
Type of slice	Fixed size (e.g. 256KB)	GoP-based
Transport	TCP	Adaptive HTTP/TCP
Pieces' selection	Random, Rarest, ...	Playout windowd, ...
Quality	1	N

How to realize an ICN?



Towards an ICN delivery system

- The modified manifest (i.e. playlist) give a reference to the stored chunk.
- If the chunk is not stored within the intra-domain it should be downloaded directly from the regional CDN
- Two possible approaches:
 - Centralized (Structured P2P): Omniscient dTracker
 - Decentralized (Non structured): Selected

Centralized ICN

- ViPeer approach
- The dTracker is in charge of populating the caches
 - Depending on chunks' popularity ...
- Huge computation requirements (require approaches like MAP Reduce, ...)
- Incompatible with caching strategies
 - Performance problem in some cases
- CDN behavior inside the intra-domain

Decentralized ICN

- The dTracker gives the optimal peers (i.e. ALTO like) to the customer
 - The requested chunks are virtually shared between peers.
- The peers send requests to their direct neighbors
 - if the requested chunk is not present locally.
- The peers decide locally to keep the chunk
 - Fully compatible with the caching strategy
- Distributed, scalable ... efficient?

Detailed architecture I

dTracker's functionality:

- Optimal peers selection
 - P4P-like approach (i.e. geographical location),
 - Possible full control of the topology by the ISP
- Semi-structured
 - Selection criterions: links' bandwidth, location, ...
- Peers don't necessarily contain the requested chunks
- Chunks are virtually shared between the selected peer
- dTracker signal to the selected peers the chunks to be requested
 - Possible proactive chunks' downloading
- Modification of the manifest
 - Direct interaction with the final users (i.e. clients)
 - Seamless to the type of clients (i.e. Dash support is the only requirement)

Detailed architecture II

Peers' functionality :

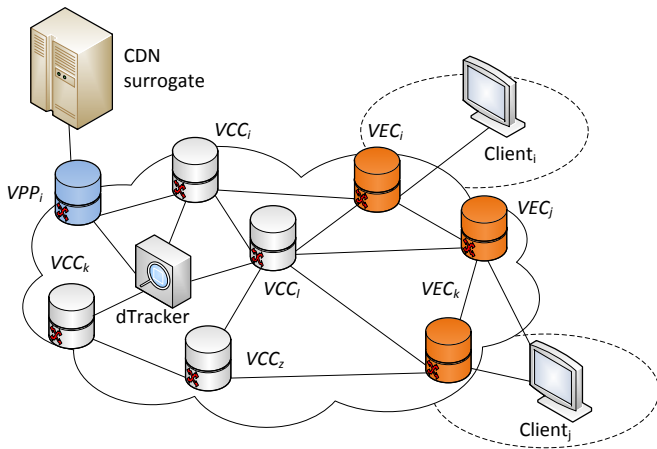
- Semi-structured
 - List of the neighboring peers controlled by the dTacker
- Fully distributed architecture
 - Non structured (Gnutella-like behavior)
 - Multi-hop (topology-aware), Automatic replication, ...
- Chunks requests
 - From the other peers if the chunks are present within the intra-domain
 - From the regional CDN if the chunks are not yet in the intra-domain

Detailed architecture III

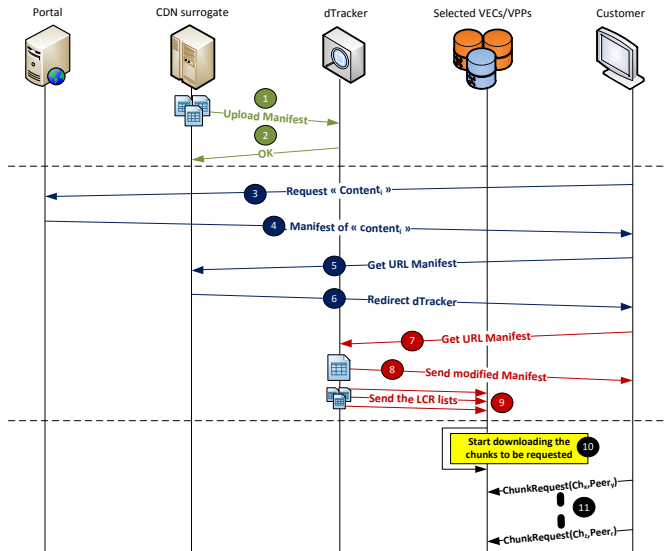
Clients' functionality :

- DASH-compliant

Network elements



Proactive download of the chunks



Content Centric Networks (CCN) I

- what?
 - Overlay solution (Gnutella-like)
 - in the intra-domain (possible interdomain communication using VCCs)
 - for adaptive video streaming (DASH)
 - using QoE indicators
- why?
 - Current technological evolution, new usages, digital switchover
 - Our previous work on P2P and on CDNs, for video delivery
 - Our previous work on congestion control
 - Keeping the current network architecture
 - Exploiting our results on QoE (e.g., adding indicators to the chunks' names)

Content Centric Networks (CCN) II

- challenges:
 - Replication and caching strategies (and possibly a combination of these strategies)
 - Routing and congestion control (and possibly a combination of these strategies)
 - Congestion avoidance (in particular, optimizing signaling overhead)
 - QoE-aware mechanisms usage
 - Cache selection algorithms
 - Economical models

Future investigations I

- Problem: cache replacement and replication
- Knapsack 0/1 problem
 - Each chunk is present in the cache (value 1) or not (value 0)
 - The presence of a chunk in the cache is associated with a gain (to be defined regarding the policy of the deployment)
 - Maximize the gain without exceeding the network cache capacity

Future investigations II

- If there are entries for an interest in the FIB table, the interest is sent to all faces:
 - this is not optimal.
- Ant Colony Optimization (ACO) solutions could be an interesting method to classify the entries regarding a certain criteria (maximize bandwidth, reduce delays... maximize quality).
 - Sent ants to find the optimum entry to use in order to satisfy the selected criteria.