On HTTP Live Streaming in Large Enterprises

ROYAL INSTITUTE OF TECHNOLOGY

PEERIALISM

Roberto Roverso
Sameh El-Ansary
Mikael Högvist

ABSTRACT

In this work, we present a distributed caching solution which addresses the problem of efficient delivery of HTTP live streams in large private networks.

With our system, we have conducted tests on a number of pilot deployments. The largest of them, with 3000 concurrent viewers, consistently showed that our system saves more than 90% of traffic towards the source of the stream while providing the same quality of user experience of a CDN.

Another result is that our solution was able to reduce the load on the bottlenecks in the network by an average of 91.6%.

REFERENCES

1. S. Akhshabi, A. C. Begen, and C. Dovrolis. An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP. In Proc. of ACM MMSys, 2011.
2. V. Raychoudhury, J. Cao, and W. Wu. Top k-leader election in wireless ad hoc networks. In Proc. of IEEE ICCCN, 2008.
3. R. Reale, R. Roverso, S. El-Ansary, and S. Haridi. DTL: Dynamic Transport Library for Peer-To-Peer Applications. In Proc. of ICDCN, 2012.
4. R. Roverso, S. El-Ansary, and S. Haridi. Smoothcache: Http-live streaming goes peer-to-peer. In Proc. of IFIP NETWORKING, 2012.

CONTACT INFORMATION

www.peerialism.com

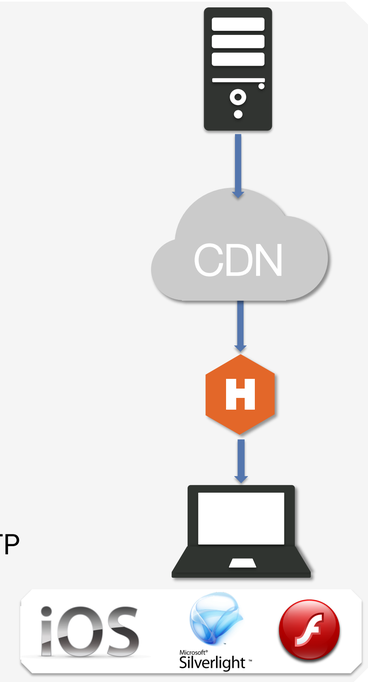
HTTP Live Streaming

HTTP-Live is the de-facto industry standard for distributing live video streams over the Internet

- Based on a **pull model** that utilizes HTTP as transport protocol
- Content available **with multiple video and audio qualities** (bitrates)
- Player implements complex heuristics to choose which quality to render

Advantages

- Routers and firewalls are more permissive to HTTP
- HTTP caching is straight-forward
- **Cheaper CDN cost**



Distributed Caching with Hive

The **Hive Client** is a local **HTTP proxy**

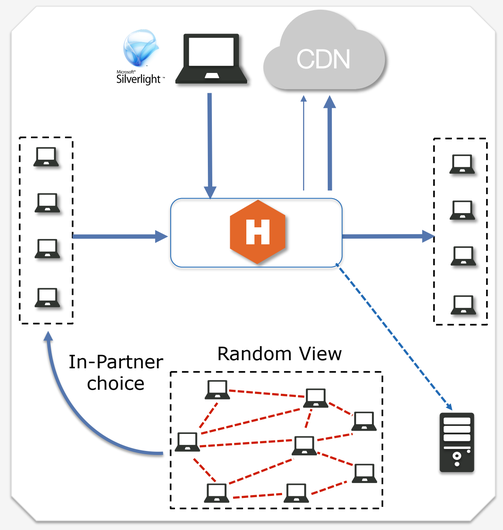
Goals

- **Off-load bottlenecks** between network segments
- Same **quality of user experience** as a CDN

Functions as **distributed cache**

- Cache hit: retrieve from other peer
- Cache miss: retrieve from CDN

Peer-to-Peer operations completely **transparent** to the player and Stream's Source



CDN vs. Hive

Advantages of a software-based solution

- No expensive hardware at each network segment, i.e. caching servers/proxies
- No need for full knowledge of network structure
- Avoid hardware administration/maintenance/upgrade

Same **quality of user experience** (QoE)

- Same experience delay
- Same or **better delivered throughput** and quality (bitrate)

Hive Building Blocks

UDP-based transport library

Security and reliability.

- Authentication towards a central authority
- All traffic is **encrypted through SSL**

DTL dynamic congestion control protocol [3]

- **Runtime priority configuration** of transfers
- Priority levels range from lower-than-best-effort, designed to **yield to TCP**, up to four times more aggressive than TCP

State-of-the art **NAT Traversal**

- NATCracker traversal scheme
- Estimated **80% two-way connectivity**

Locality-aware **peer sampling**

- Gossip-based
- **Used for membership**
- Sampling of peers characteristics: network segment, upload bandwidth, request point, current requested bitrate, connectivity information

Locality-aware **K-leader election**

- Tracker and Gossip-based
- 2nd-Layer discovery (UDP Broadcast)

Hive Overlay

Random mesh

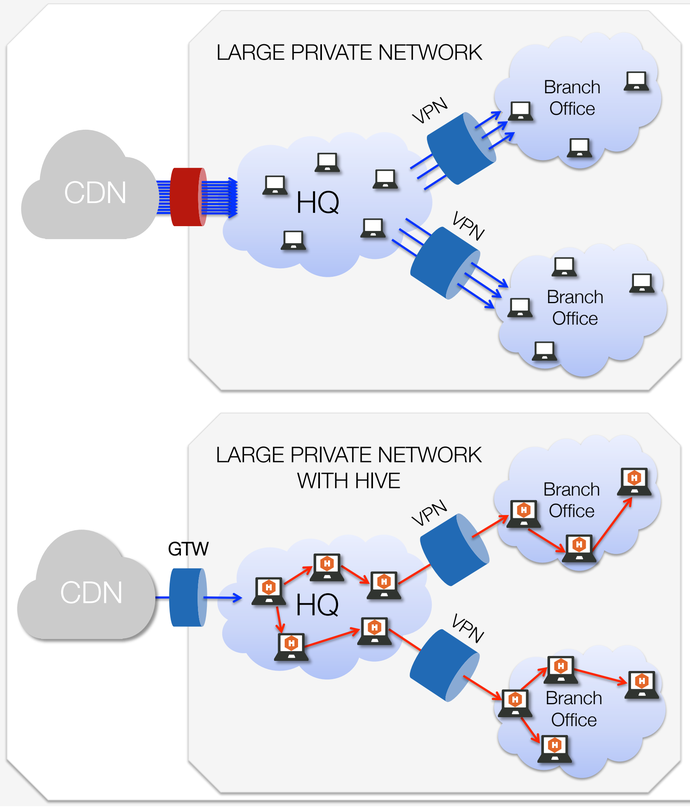
- Build **fixed-size neighborhood** randomly with input from a tracker or other peers (peer-exchange)
- Collect information about locality, stream characteristics (peer sampling) and reliability

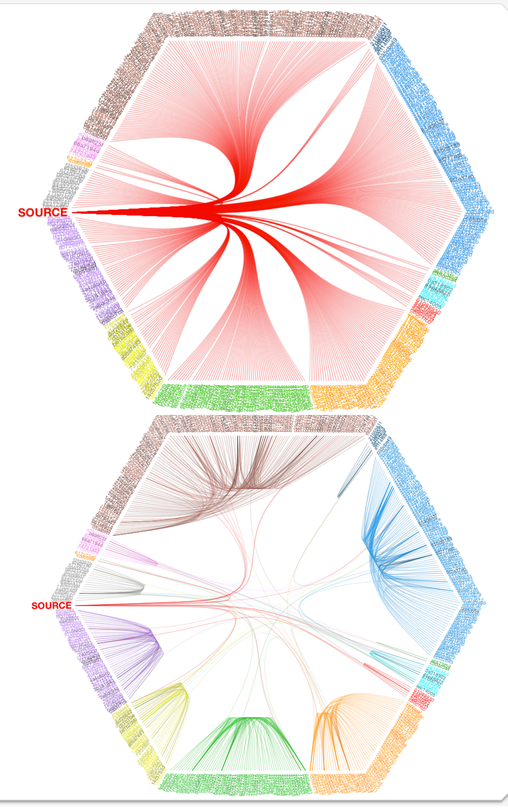
Delivery mesh

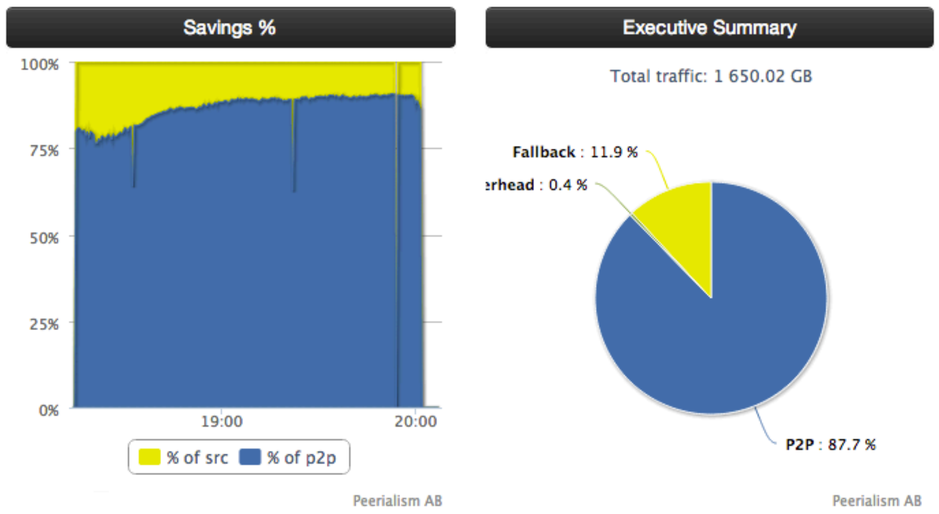
- Build fixed size partnership neighborhood
- Chose preferably peers inside same network segment (office, branch, HQ)
- Identify **leaders for each network segment** (K-leader election [2])
- Leader connect to one another to form a **overlay hierarchy which follows network structure**
- Delivery mesh is replicated for each bitrate

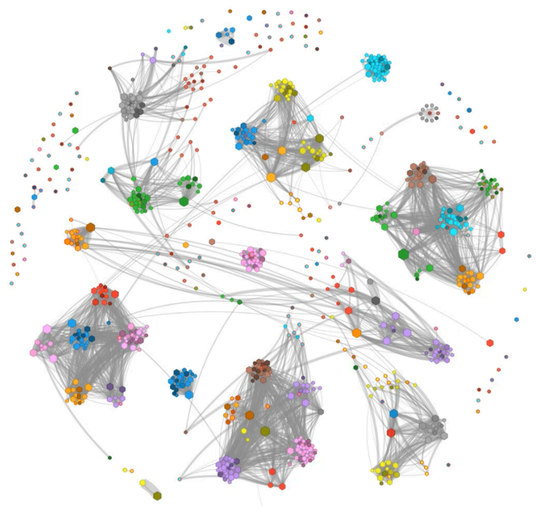
Delivery

- Leaders **pre-fetch data ahead of other peers** and inject in network segment
- Non-leader peers swarm inside network segment









Results from a stream with more than 2000 concurrent clients and the overlay map