MOLStream: A Modular Rapid Development and Evaluation Framework for Live P2P Streaming

Prof. Roy Friedman, Alexander Libov, Prof. Ymir Vigfusson

Technion

HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Introduction

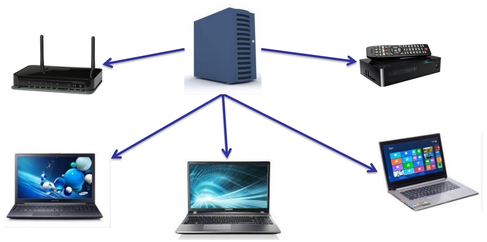
Streaming of video from the Internet has became more and more popular

Over 6 billion hours of video are watched each month on YouTube

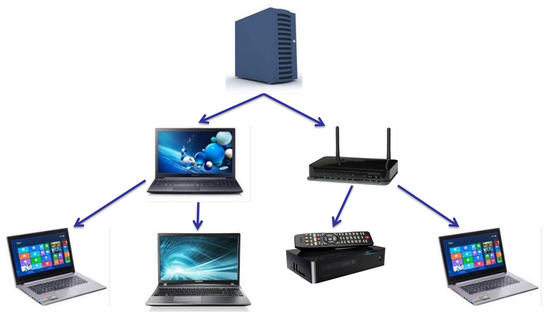
20% of all videos are cat videos

Centralized vs. P2P

Streaming can be done using a *centralized solution*



A *peer-to-peer (P2P)* solution can save money to the service provider and ISPs while keeping clients happy



Research questions

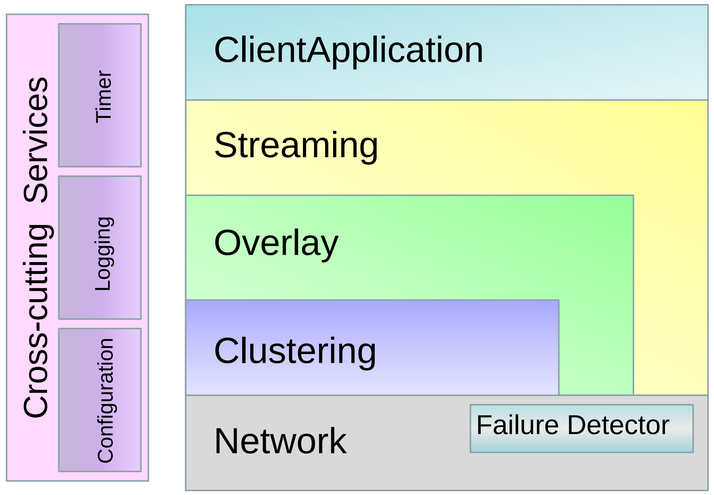
What are the building blocks of a P2P live streaming system?
Can every system be decomposed into these blocks?

Can we mix and match building blocks from different live streaming systems?

Overview

MOLStream allows P2P streaming protocols to be decomposed into basic blocks, each associated with a standard functional specification

MOLStream enables specific implementations of these building blocks to be combined in order to devise, refine and evaluate new P2P live streaming protocols

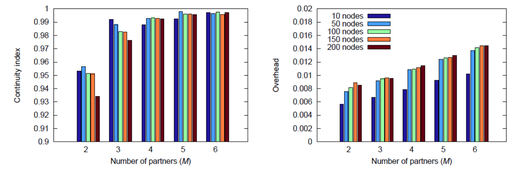


MOLStream Library

Module	Type	LOC
Coolstreaming	Streaming algorithm	274
mTreeBone	Streaming algorithm	35
TreePush	Streaming algorithm	46
Prime	Streaming algorithm	305
Bootstrap: Random group	Overlay protocol	92
Bootstrap: Random node	Overlay protocol	78
Coolstreaming	Overlay protocol (gossip-based)	169
Araneola	Overlay protocol (gossip-based)	196
General gossip-based overlay	Overlay protocol (gossip-based)	68
SCAMP	Overlay protocol (gossip-based)	66
Prime overlay	Overlay protocol (group-based)	56
TreeBone overlay	Overlay protocol (tree-based)	276

We have implemented several popular P2P live streaming protocols in MOLStream

Comparison



(a) MOLStream Coolstreaming Playback continuity index.

(b) MOLStream Coolstreaming overhead.

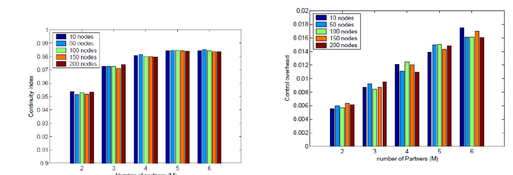


Fig. 8. Continuity index as a function of the number of partners.

(c) Original Coolstreaming continuity index graph (from [24]).

Fig. 7. Control overhead as a function of the number of partners for different overlay sizes. (Control overhead= Control traffic volume/Video traffic volume at each node).

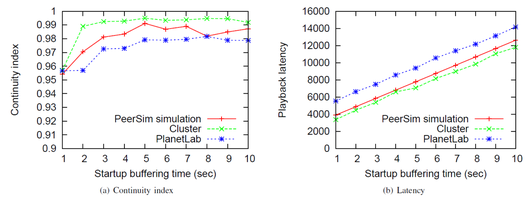
(d) Original Coolstreaming overhead graph (from [24])

Figure 2. **Case study of CoolStreaming.** The continuity index and the overhead of a MOLStream Coolstreaming implementation protocol in the MOLStream compared to the original results of the Coolstreaming protocol.

We implemented Coolstreaming in the MOLStream framework.

We compare to the results from the original Coolstreaming paper.

Results



(a) Continuity index

(b) Latency

Figure 5. Comparison of Coolstreaming using PeerSim simulator, Cluster and Planetlab testbed

The same logic can run on a single machine as a simulation or on a real network.

Table II. Continuity Index, Playback Latency and Control overhead measured for different overlays with the Coolstreaming streaming component

Overlay	Continuity Index	Playback Latency (ms)	Control Overhead
AraneolaOverlayL03	0.997	9893	0.008
BSCAMPOverlay1	0.996	9771	0.008
CoolStreamingOverlayM4	0.97	9239	0.01
PrimeOverlay	0.925	8624	0.006

We compare between different overlays and the Coolstreaming Streaming module