EXPERIENCES FROM A FIELD TEST USING ICN FOR LIVE VIDEO STREAMING

Authors:

Adeel Malik Börje Ohlman (Ericsson)	Bengt Ahlgren Anders Lindgren (SICS)	Edith Ngai Lukas Klingsbo Magnus Lång (Uppsala University)

Presenter: Adeel Malik

Workshop on Multimedia Streaming in ICNs (MuSIC)
3rd July, 2015

UPPSALA UNIVERSITET

SWEDISH ICT
SICS

ERICSSON

INTRODUCTION

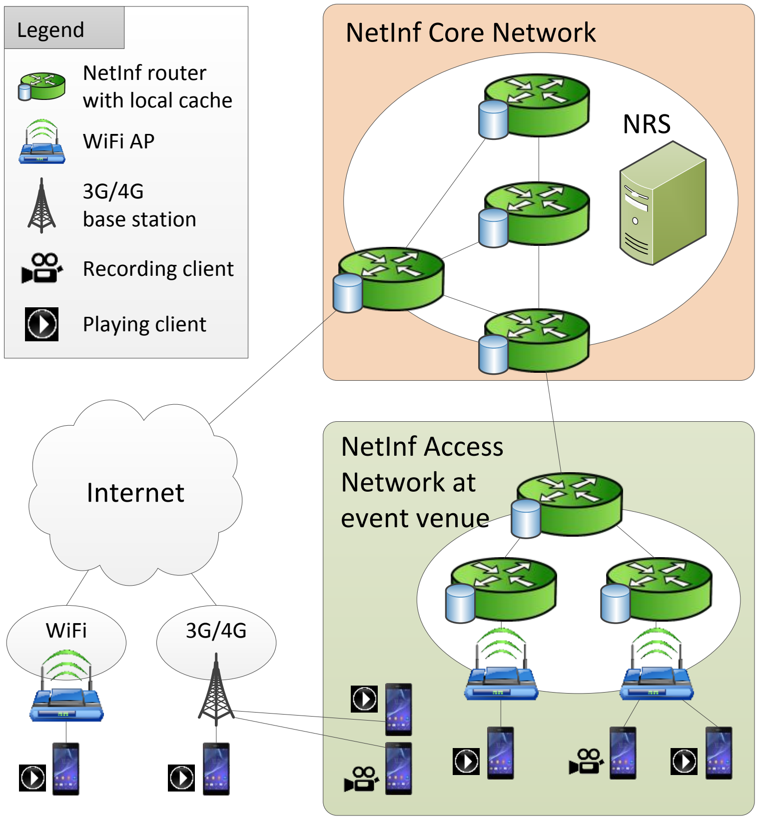
- A live video streaming system built on the NetInf ICN architecture
- System was field tested at the FIS Nordic Ski World Championship in February 2015 in Falun, Sweden
- System includes a set of NetInf routers + Mobile streaming application for video recording and live viewing

MOTIVATION

- System targets the use case at "events with large crowds" e.g. Sports events, concerts, festivals, fairs
  - ICN is very well suited
  - Current cloud-based services → One unicast data stream per client
  - No dependency on global ICN infrastructure
- ICN treats data objects as a first class citizen
  - Caching of content at the data level
  - Request aggregation at the data level
  - Reduced congestion and improved delivery speed

ARCHITECTURE

- System built on the NetInf ICN architecture → Overlay on existing Internet protocols
  - Connectivity to the global internet
  - Incremental deployment of the system possible
- Name Resolution Server (NRS)
  - Stores Name-Locator bindings
  - Stores metadata and provides search function
- NetInf Service Discovery
  - Local WiFi infrastructure → Multicast DNS (mDNS)
  - Internet → DNS resolution
- Caching
  - NDOs cached on-path
  - Primarily useful when playing “recorded” videos
- Request Aggregation
  - Subscription and GET requests aggregated
  - Place NetInf routers at network edges

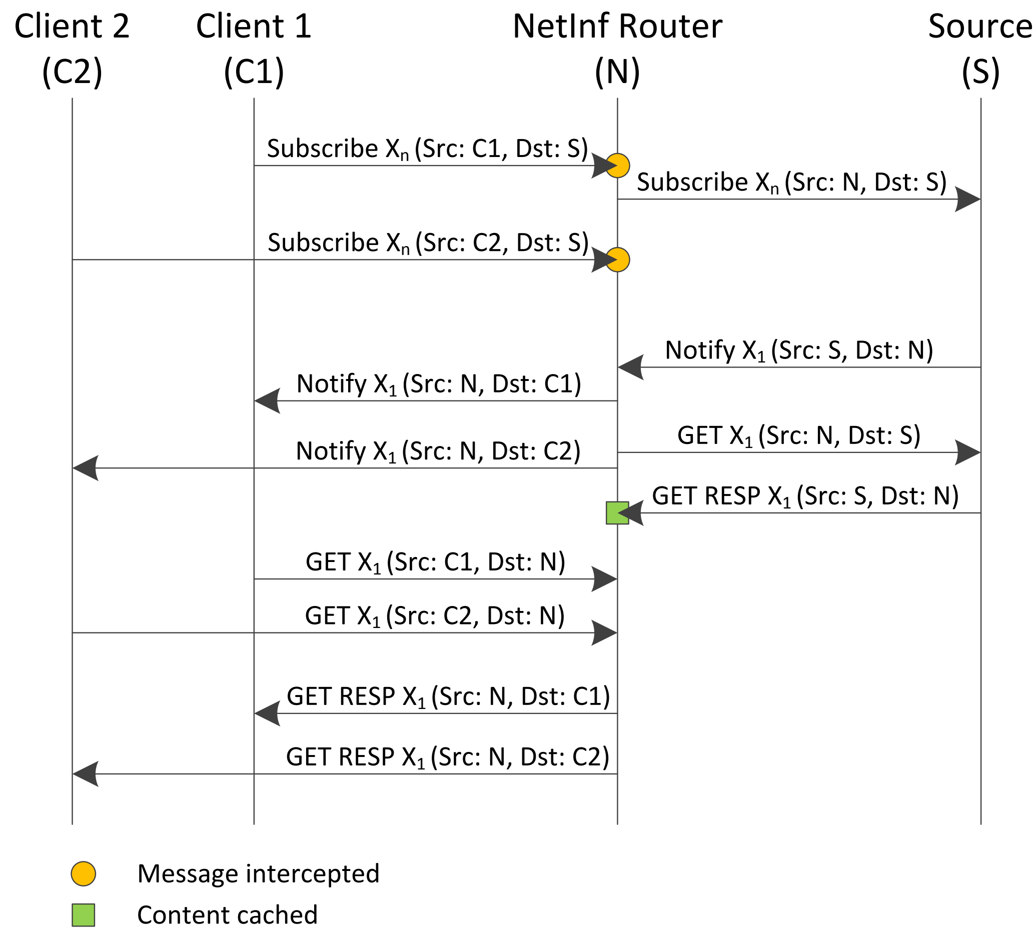


VIDEO STREAM REPRESENTATION

- In ICN content is abstracted in the form of Named Data Objects (NDOs)
  - A video can be organized into several video chunks/NDOs
- Entire video stream is represented by a single Header NDO
  - The Header NDO contains metadata for each video
- Video chunk NDOs are linked to the Header NDO through a field in their metadata

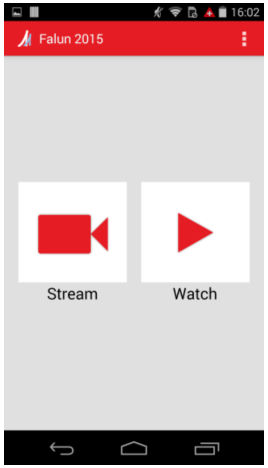
SUBSCRIBE-NOTIFY PROTOCOL AND CONTENT RETRIEVAL

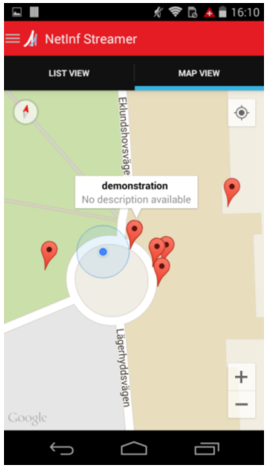
- Hop-by-hop subscription relationships
- Subscription requests are aggregated
- Hierarchical point-to-multipoint tree
- Subscribe-Notify messages → Netinf UDP convergence layer
- NetInf GET and GET RESP messages → NetInf HTTP convergence layer

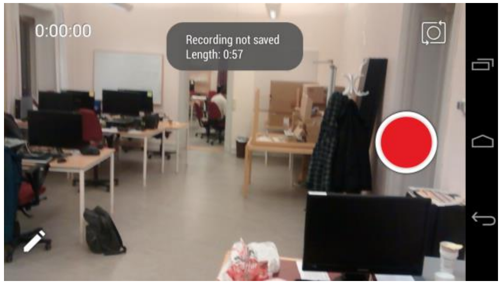


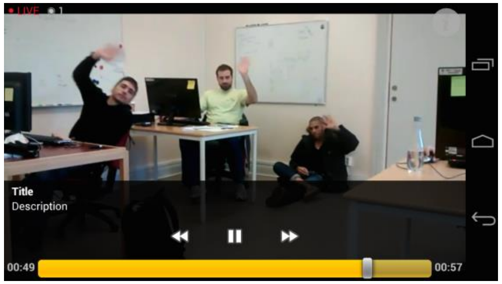
ANDROID APPLICATION

- Recording and viewing clients
- Event Browser
  - Video stream selection
- Video encoding and chunking
  - H.264 encoding
  - MP4 packaging
  - Data rate = 1 Mbps
  - 1 chunk = 2 secs
  - Playout buffer = 10 secs



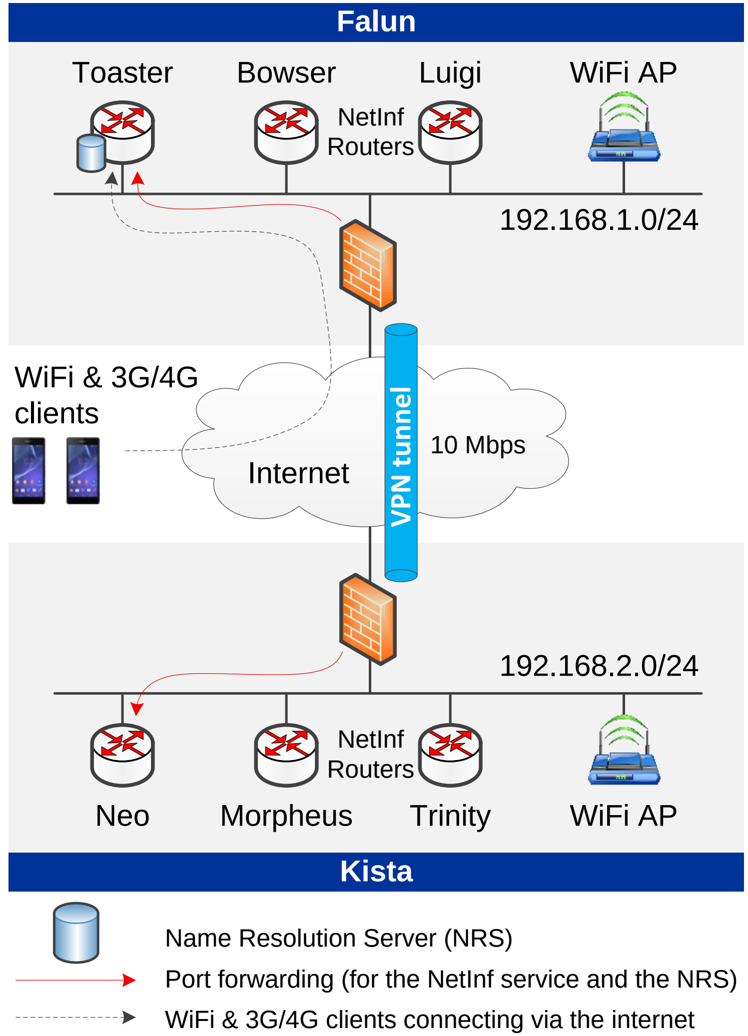






NETWORK SETUP FOR THE FIELD TEST

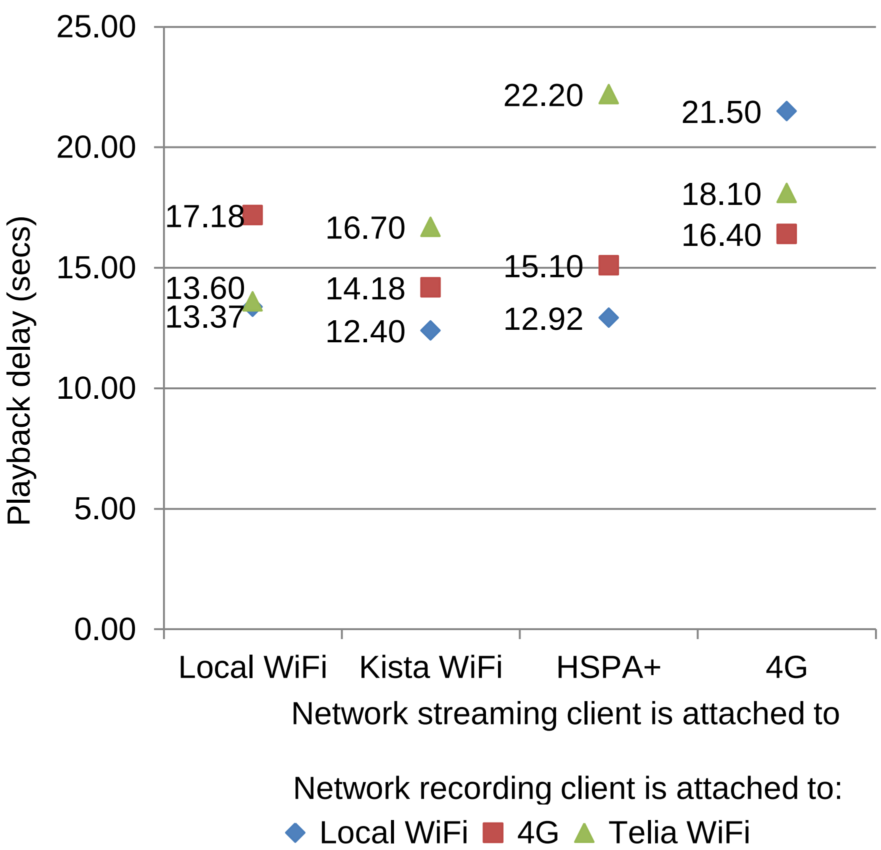
- Subnet separation for the two sites
  - For proper functioning of mDNS
- Traffic aggregation over the VPN tunnel
- Why was a VPN tunnel needed?
  - Only one public IP address in Falun
- NRS hosted at Toaster
- Software for NetInf routers written in Erlang; streaming mobile app is Android-based

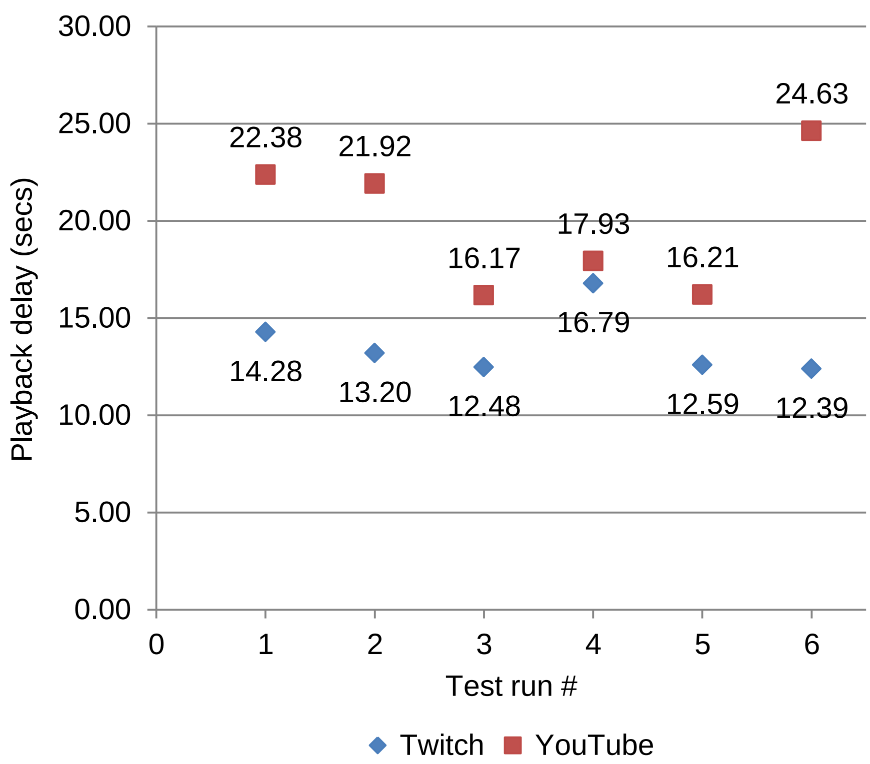


TESTS AND MEASUREMENTS

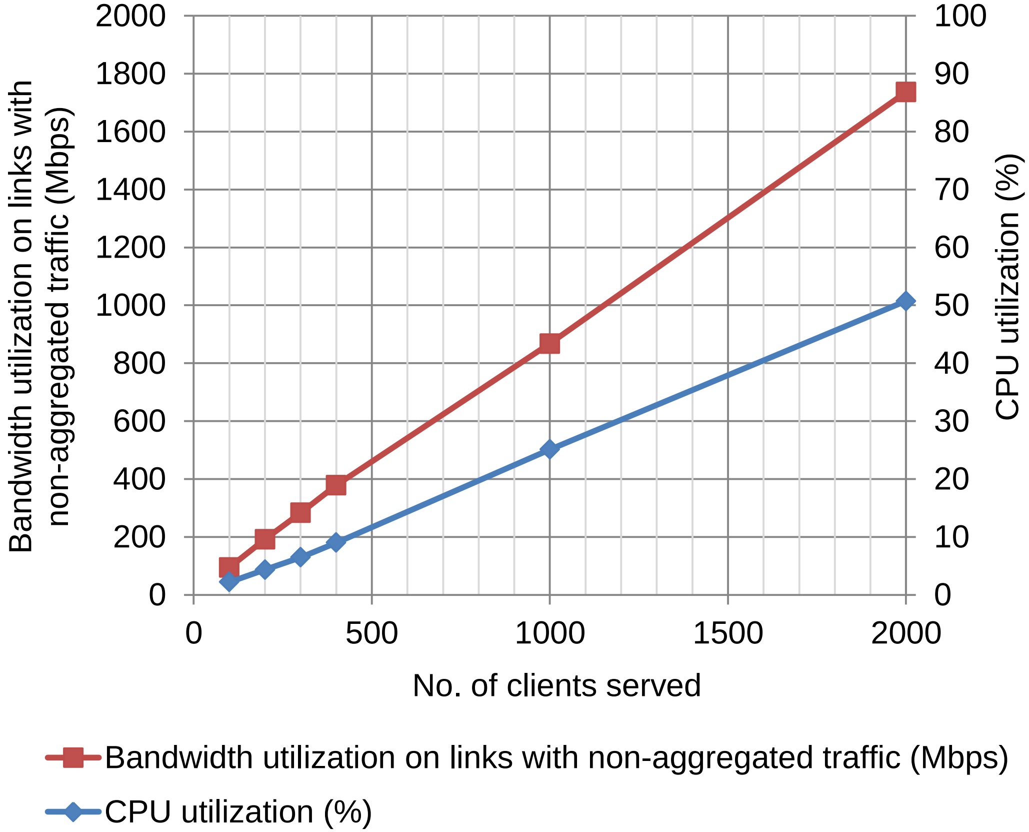
- End-to-end live streaming delay
  - Comparison with Twitch and YouTube
- System scalability with a large number of clients
  - Huge number of emulated clients used to measure the aggregation efficiency of the NetInf router across the 10 Mbps link
- System robustness → Qualitative field tests performed with 20 Android mobile devices
  - Several recording clients publishing at the same time
  - Several playing clients streaming at the same time

RESULTS – PLAYBACK DELAY MEASUREMENTS





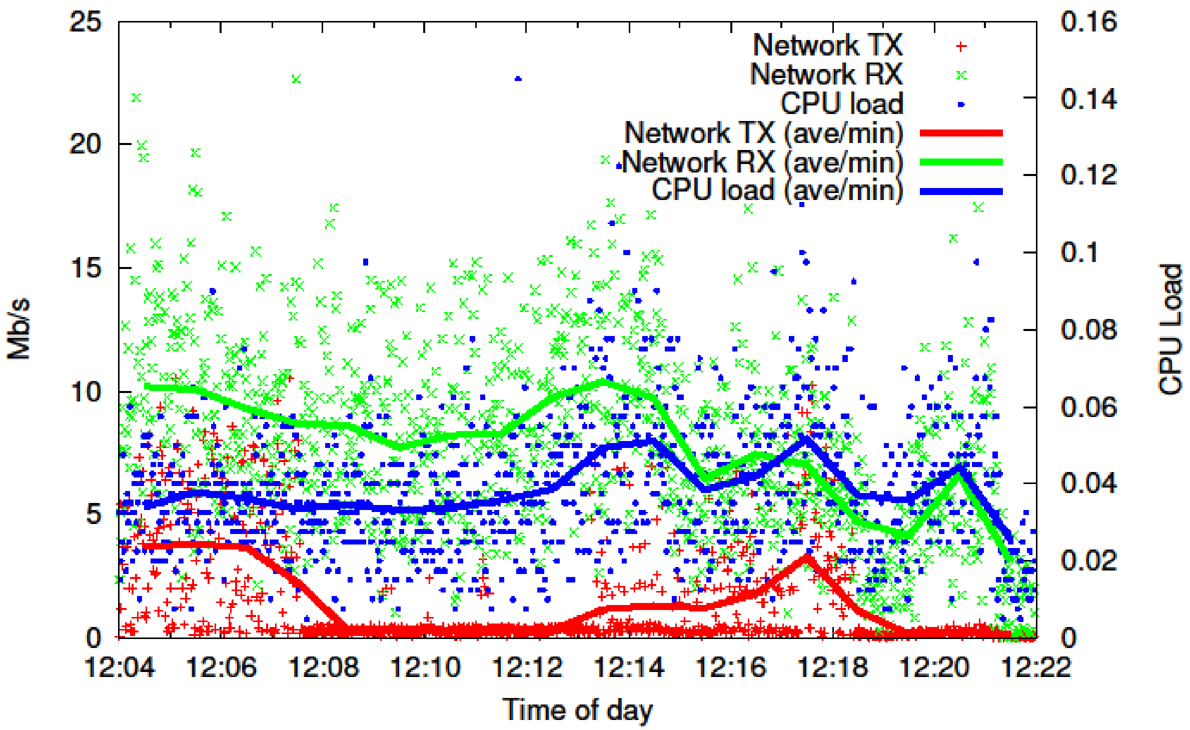
RESULTS – AGGREGATION EFFICIENCY OF THE NETINF ROUTER



Bandwidth on link with aggregated traffic = 10 Mbps
Total bandwidth on links with non-aggregated traffic = 2 Gbps

RESULTS – QUALITATIVE FIELD TEST I

- Network and CPU load with many publishers
- 10 recording clients simultaneously published live streams
- Average NetInf publish rate = 2 per second
- Average CPU load = 4%
- Average Network RX load = 7.8 Mbps



EXPERIENCES FROM THE FIELD TEST AND CONCLUSIONS

- System has been very stable → NetInf routers have been running for months
- When publishing/viewing did not work it was in most cases due to bad connectivity (usually WiFi problems)
  – User would benefit from having more information directly in the app about the current connectivity status
- ICNs and global flash crowds
  – Current CDNs do not work well when the demand for the services is unknown; for true flash crowds there are no solutions that scale to global audiences
  – We have performed limited field tests; we still think that the scalability properties of ICN and NetInf look very promising
  – We can aggregate more than 2000 clients on one router; with a three-level hierarchy we can stream to a global audience of 8 billion users from one Android phone without having to do any pre-configuration in the network

Questions and Discussion