PROJECT SPECIFICATION OVERVIEW

2nd September, 2014
BREAKDOWN

› Recording/Playing clients
› Naming, Header NDO, Manifest NDOs
› NetInf core network, Name Resolution Server (NRS), NetInf routers
› NetInf DHCP
› Event browser
› NetInf transport
› Storage eviction
› Video/Audio encoding & chunking
› Code considerations
NETWORK ARCHITECTURE

NetInf core network

LEGEND
- NetInf router with local storage
- WiFi AP
- GW router
- 3G/LTE BS
- Recording client
- Playing client

Global IP network

NetInf router instances

NetInf DHCP

NRS

Stadium & Ski tracks

Local access network

NetInf router with local storage

WiFi AP

GW router

3G/LTE BS

Recording client

Playing client

Local access network

3G/LTE core network

NetInf router

NetInf DHCP

Global IP network

NetInf core network

Stadium & Ski tracks

Local access network

NetInf router with local storage

WiFi AP

GW router

3G/LTE BS

Recording client

Playing client
RECORDING/PLAYING CLIENTS

› Recording client
  – Break recorded video into chunks
  – Publish chunks locally or onto a NetInf node in the NetInf core network
  – For each published chunk, add name-location binding in the Name Resolution Server (NRS)

› Playing client
  – Fetch a live or previously recorded stream
  – Provide an Event Browser interface (for users to access videos)
  – Ability to rewind the video during live streaming
RECORDING/PLAYING CLIENTS

› Use NetInf transport for publishing and fetching data
   – Always-on background process

› Use MPEG-DASH or HLS streaming standards
   – For segmenting, re-assembling and streaming
   – Must be adapted to run over NetInf transport

› Organization
   – Recording and Playing clients can be integrated or exist as separate apps
   – Event browser can be integrated with the Playing client or exist as a separate app
NAMING, HEADER NDO, MANIFEST NDOs

› Naming
  - Hashing content for video chunk NDOs
  - SHA-256 can be used for hashing

› Header NDO
  - Entire video represented by one NDO
  - Contains references to manifest NDOs
  - Contain video meta-data e.g. title, publisher, timestamp, geo-location etc.

› Manifest NDOs
  - Each manifest NDO corresponds to a specific representation of video/audio stream (i.e. bit rate, resolution etc)
  - Manifest NDO contains list of published audio/video NDOs

› Key issues
  - Naming of NDOs with dynamic content e.g. Manifest NDOs
  - Referencing video/audio chunk NDOs in Manifest files
  - Number of video/audio chunk NDOs listed in Manifest files during live streaming
  - Retrieving previously generated chunks when rewinding
MPEG-DASH MEDIA PRESENTATION DESCRIPTION (MPD) DATA MODEL

MPD describes accessible Segments and corresponding timing

- Media Presentation
  - Period, start=0s
  - Period, start=100s
  - Period, start=295s

- Adaptation Set 1
  - video

- Adaptation Set 2
  - audio

- Representation 1
  - bandwidth=500kbit/s
  - width 640, height 480

- Representation 2
  - bandwidth=250kbit/s
  - width 640, height 480

- Segment Info
  - Initialization Segment
    - http://www.e.com/ahs-5.3gp
  - Media Segment 1
    - start=0s
    - http://www.e.com/ahs-5-1.3gs
  - Media Segment 2
    - start=10s
    - http://www.e.com/ahs-5-2.3gs
  - Media Segment 3
    - start=20s
    - http://www.e.com/ahs-5-3.3gs
  - Media Segment 20
    - start=190s
    - http://www.e.com/ahs-5-20.3gs

- Splicing of arbitrary content
- Selection of Components
- Select/Switch of Bandwidth
NetInf Core Network
- Contains NetInf routers, central Name Resolution Server (NRS), NetInf DHCP

Name Resolution Server (NRS)
- Lookup directory for locating NDOs
- Central NRS in NetInf Core Network
  - Acts as last-resort NRS
  - Must store location bindings for all "published" data
- Local NRS in NetInf routers
  - Stores bindings for local NDOs (published or cached)

NetInf routers
- Caching, local NRS, Request Aggregation, Publish-Subscribe
- Local storage (permanent storage and temporary storage)
NETINF DHCP

› Facilitate auto-configuration of NetInf nodes (clients, routers)

› Auto-configuration:-
  – Determine next-hop NetInf router for recording/playing clients
  – Load distribution between NetInf routers?
EVENT BROWSER

› Displays a map of the arena

› Video thumbnails plotted on the map based on geo-location of the recorded video

› Information can be retrieved from the NRS
  – NRS contains Header NDOs with associated meta-data (title, publisher, timestamp, geo-location etc)
  – Mechanism → Poll or Publish-Subscribe

› Video access:-
  – Live snapshot
  – Time sliders for historical videos
NETINF TRANSPORT

- UDP convergence layer
  (DHCP REQ, DHCP RESP, SUBSCRIBE, NOTIFICATION)

- HTTP convergence layer
  (GET, GET-RESP, PUBLISH, PUBLISH-RESP)
STORAGE EVICTION

› Storage in NetInf nodes
  - Temporary
    › Store cached data
    › Simple eviction algorithm → Evict NDO with least hits in recent time window
  - Permanent
    › Store published data
    › Sophisticated eviction algorithm → Delete video in its entirety, no stray video/audio chunks in the network
VIDEO/AUDIO ENCODING & CHUNKING

Recorder
Capture images
Video/Audio Encoder
Encoded byte stream
Segmenter
Split encoded stream to form video chunks
NetInf Wrapper
NDOs sent over NetInf transport
MPEG-DASH / HLS

Player

Video/Audio Decoder
Encoded byte stream
Merger
Video chunks
NetInf Unwrapper
Merge back all chunks to retrieve the encoded stream

Recorder

Video/Audio Encoder

Segmenter

NetInf Wrapper

Player

Video/Audio Decoder

Merger

NetInf Unwrapper
CODE CONSIDERATIONS

› Available implementations
  – NetInf router (Erlang)
  – Android recording/playing application (Java)
  – NetInf library (Java)
  – Event Browser (Java)

› Optimizations
  – Minimize streaming delay
  – NetInf transport → Always-on background process
  – Reliability and performance
  – Android application should be user-friendly, easily downloadable, independent of Android version