SCALABLE REAL-TIME COLLABORATIVE COMMUNICATION OVER NDN USING SERVICE EDGE ROUTERS

Syed Obaid Amin, Haitao Zhang*, Asit Chakraborti, Aytac Azgin, Ravishankar Ravindran and GQ Wang
Huawei Research Center Santa Clara, *UCLA
{obaid.amin, asit.chakraborti, aytac.azgin, ravi.ravindran, gq.wang}@huawei.com, *haitao.zhang@gmail.com
Introduction

• Recently we have converted our Audio/Video conferencing application to NDN from CCNx-0.8.2

• Reasons:
  – Scalability, after 12 participants (each consumes and produces the AV stream), the CCNx forwarder didn’t seem to scale well.
  – To experience NFD performance
Quick Revision

[A]: Conference Agent
- CA discovers the CP to join the conference. CP returns the namespace to CA to join an active conference session
- Conference session namespace is pushed to CP and also shared with CA
- Namespace contains the time-stamp, sequence identifying the content
- CA maintains a local digest tree of namespaces for recovery

[B]: Conference Proxy
- CP maintains a local digest tree of namespaces for recovery
- Digest updates from remote conference participants are pushed to CA
- CP handles multiple conference sessions simultaneously using conf-Id
- CP pushes the digest updates to SC and receives updates from SC to be pushed to CA

[C]: Sync Controller
- SC relays the namespaces among distributed CP instances
- Maintains a digest tree of conference state updates received from remote CP
- Updates from one CP are pushed to the remote CP based on the conference state and the Interest shown by specific CP

Application Architecture

GUI based version

- Used for actual demo
- Live feed is captured from Camera and Microphone

Headless version

- Used for emulation
- No GUI
- Emulated generation of AV packets
- Goal: testing scalability
Emulation Details using Headless Version

- 3 Service Edge Routers
- 5 host server each running 9 containers
- Video traffic model was derived from our earlier prototype
- Audio content was modeled after G.729 codec generating CBR traffic of 30Kbps
Emulation Results

- Two sets of experiments:
  1. All participants are producing and consuming
  2. One consumer rest of the participants act as producer
- Caching affords better performance even for real-time applications
- The 90th percentile is <150ms and <250ms for Audio and Video

### Percentage of usable contents at the consumer side

<table>
<thead>
<tr>
<th>#Participants</th>
<th>Quality_AUDIO (%)</th>
<th>Quality_VIDEO (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>99.93</td>
<td>99.91</td>
</tr>
<tr>
<td>6</td>
<td>99.98</td>
<td>100.00</td>
</tr>
<tr>
<td>9</td>
<td>100.00</td>
<td>99.97</td>
</tr>
<tr>
<td>12</td>
<td>99.76</td>
<td>99.81</td>
</tr>
<tr>
<td>15</td>
<td>97.69</td>
<td>97.46</td>
</tr>
</tbody>
</table>

### Audio and video latency performance

- Audio and video latency performance with all consumer-producer nodes
- Audio and video latency performance with single consumer and multiple producer nodes
The Demo Topology

UE: User Entity
CA: Client Agent
NFD: NDN Forwarding Daemon
Thanks!
Backup slides
Motivation

• ICN Deployment
  – Caching and aggregation at the Edge
  – Names for service/content/device enable context aware networking
  – Potential for new business models for network operators

• Service from the Edge
  – Service-centric Compute, Storage and Bandwidth scaling using virtualization
  – Tailor services to locality and user context (mobility, social parameters)
  – Minimize latency and jitter
  – Avoid backbone bottlenecks

• NFV/SDN programmability
  – Enables compute and network virtualization
  – Allows realization of new network architectures like ICN
  – 5G Network Slicing using the same technologies
AV Conferencing Survey

- Many existing solutions support only multi-party audio conference and 2-party video conference
- P2P systems:
  - High control signaling overhead, complex client design
- Client/Server:
  - Centralized processing, complex design, expensive, limited scalability
- IP Multicast:
  - Due to lack of extensive IP multicast deployments, very few IP multicast based conferencing solutions are available

<table>
<thead>
<tr>
<th>Service</th>
<th>Max. frame rate (frames/second)</th>
<th>Max. # of simultaneous video participants</th>
<th>S/C or P2P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eedo WebClass</td>
<td></td>
<td>6</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>IOMeeting</td>
<td>30</td>
<td>10</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>EarthLink</td>
<td>30</td>
<td>24</td>
<td>S/C</td>
</tr>
<tr>
<td>VideoLive</td>
<td>30</td>
<td>6</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>Himeeting</td>
<td>17</td>
<td>20</td>
<td>S/C</td>
</tr>
<tr>
<td>VidSoft</td>
<td>30</td>
<td>10</td>
<td>S/C</td>
</tr>
<tr>
<td>MegaMeeting</td>
<td>30</td>
<td>16</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>Smartmeeting</td>
<td>15</td>
<td>4</td>
<td>S/C</td>
</tr>
<tr>
<td>Webconference</td>
<td>15</td>
<td>10</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>Mebeam</td>
<td></td>
<td>16</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>Confest</td>
<td>30</td>
<td>15</td>
<td>S/C</td>
</tr>
<tr>
<td>CloudMeeting</td>
<td>30</td>
<td>6</td>
<td>S/C</td>
</tr>
<tr>
<td>Linktivity WebDemo</td>
<td>30</td>
<td>6</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>WebEx</td>
<td>30</td>
<td>6</td>
<td>web-based S/C</td>
</tr>
<tr>
<td>Nefsis</td>
<td>30</td>
<td>10</td>
<td>S/C</td>
</tr>
<tr>
<td>Lava-Lava</td>
<td>15</td>
<td>5</td>
<td>decentralized P2P</td>
</tr>
<tr>
<td>Qnext</td>
<td></td>
<td>4</td>
<td>centralized P2P</td>
</tr>
<tr>
<td>Vsee</td>
<td>30</td>
<td>8</td>
<td>decentralized P2P</td>
</tr>
</tbody>
</table>

Comparison of different conferencing models

Centralized, IP based: Bandwidth within the network is $O(N^2)$ where $N$ is the number of clients.

Decentralized, ICN based: Bandwidth consumed in the network is $O(N \times R)$ where $N$ is the number of clients, and $R$ is the number of Edge Routers.
Simplified System Design

Controller

VSER (Proxy + NFD)

VSER (Proxy + NFD)

UE
Client Agent
AV App

UE
Client Agent
AV App

UE
Client Agent
AV App
Why NDN-based video conference

• NDN has built-in cache/multicast support
  – Reduce network traffic

• NDN has built-in mobility support
  – No concept of end-to-end connection
  – Do not need to tear down old connections and set up new connections again

• NDN has built-in security support
  – Provide a way to verify the identities of data publishers
  – Provide a way to secure the video data directly (do not rely on third parties)