Towards Pervasive and Mobile Gaming with Distributed Cloud Infrastructure

Teemu Kämäräinen, Matti Siekkinen, Yu Xiao, Antti Ylä-Jääski
Introduction
Background

In Mobile Cloud Gaming the game is rendered on the cloud and streamed to a mobile thin client such as a mobile phone.

Latency is still the main issue in Cloud Gaming.

Mobility adds other issues: screen size, controls, power consumption …
Motivation from previous work

Different games have different latency requirements. [1]

Current distant cloud infrastructure cannot meet the requirements of all game types. [2]

Cloud gaming enables the use of (almost) any device as the user equipment. The user should have the freedom to choose and switch from using one device to another on-the-fly.


Proposed solutions

Different games have different latency requirements. [1]

Choose server location based on the requirements of the specific game.

Current distant cloud infrastructure cannot meet the requirements of all game types. [2]

Use a more distributed architecture to host cloud gaming servers.

Cloud gaming enables the use of (almost) an ....

Design a prototype which enables the user to switch between the mobile phone and for example a public display.
System Design
System Design: Key Points

Distributed cloud computing infrastructure
- Public (distant) cloud, Enterprise cloud, private cloud

Central Resource Provisioning server
- Makes the decision on which physical machine to instantiate the server
- Criteria: requirements of the particular game, availability, network latency etc.

User has the freedom to choose between several different ways to display and control the game
- Directly on a smartphone with the touch screen or a gamepad
- Using the smartphone as an input device while showing video on a public display or TV
Distributed Cloud Infrastructure

Distant Cloud (e.g. Amazon EC2)

Resource Provisioning Server

Internet

Public display

Data center at operator premises

Local network

Smart TV

Mobile device

Gamepad

Private cloud

VM

VM

VM

VM

VM
Implementation
Implementation

Clients use the open-source cloud gaming platform GamingAnywhere

- Added the support for external controls
- Modified the Android client to include “control only” mode when the video display is directed to an external screen ie. a public display

Prototype uses VM synthesis techniques developed by Ha et al. [3] to launch VMs

A proof-of-concept HTML5 client also implemented.

Workflow of initializing the game

1. Service request - identities - game request
2. - 4. Inform state changes
3. VM ready message
4. Video stream
5. Control flow

Game Hosting Cloud

Public display

Game client / Web browser

Distributed File System

Provisioning service
Switching from the phone to a browser in a public display

1. Initiate connection with smart phone to the cloud server
2. Open URL in the public display or Smart TV (or a display connected to a computer)
3. Scan the display specific QR code given by the web site
   → The video is streamed to the public display
   → User can continue to control the game with the mobile phone
Switching from the phone to a browser in a public display (1/4)
Switching from the phone to a browser in a public display (2/4)
Switching from the phone to a browser in a public display (3/4)
Switching from the phone to a browser in a public display (4/4)
Evaluation
## Metrics

<table>
<thead>
<tr>
<th>Game type</th>
<th>Delay range (ms) (MOS 4 to 3)</th>
<th>Threshold chosen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast-paced</td>
<td>40-80</td>
<td>60</td>
</tr>
<tr>
<td>Medium-paced</td>
<td>80-180</td>
<td>130</td>
</tr>
<tr>
<td>Slow-paced</td>
<td>80-300</td>
<td>190</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network type</td>
<td>Wi-Fi / LTE / 3G</td>
</tr>
<tr>
<td>Server location</td>
<td>Local (Local network / ISP) / Remote cloud (Amazon EC2)</td>
</tr>
<tr>
<td>Control</td>
<td>Touch screen / External gamepad</td>
</tr>
<tr>
<td>Display</td>
<td>Smartphone / External display</td>
</tr>
</tbody>
</table>
### Response delay

<table>
<thead>
<tr>
<th>Network delay (ND)</th>
<th>Playout delay (OD)</th>
<th>Processing delay (PD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wi-Fi Local</td>
<td>LTE - ISP</td>
<td>LTE - EC2</td>
</tr>
<tr>
<td>54</td>
<td>56</td>
<td>108</td>
</tr>
<tr>
<td>130</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Response delay (ms)

- **Fast-paced games**: 108 ms
- **Medium-paced games**: 56 ms
- **Slow-paced games**: 54 ms

This diagram illustrates the response delay for different types of networks and game speeds.
Power measurement setup
Power consumption

Average Power Consumption (mW)

- Control & Video (LTE): 2971 mW
- Control only (LTE): 1921 mW
- Control & Video (Wi-Fi): 2017 mW
- Control only (Wi-Fi): 1212 mW
Power consumption

The use of Wi-Fi cuts the power consumption by a third

Using the mobile phone only for controls also yields very significant energy savings
- 35% for Wi-Fi and 40% for LTE

Touch screen seems to use more power than an externally connected gamepad (USB)
Conclusion & Future Work
Conclusion

Proximity of the cloud gaming server should be influenced by the latency tolerance of the particular game

- User’s network connection also affects the need to instantiate the games closer

Cloud gaming enables the flexible use of user equipment and public displays to control and view the game

A short enough response delay is achievable even for the most demanding games using the distributed deployment model
Future Work

A user study needed to better quantify the quality of experience for different delays and control methods.

Further analysis of the delay variations in LTE networks. For example the effect of signal quality and cross-traffic.

A wide-scale analysis of the optimal cloud server topology for example for the use case: Europe.
Preliminary Results

- Network delay (ms) vs. RSRP (dBm)
  - Median
  - 25% percentile
  - 75% percentile

- Time (s) vs. Network delay (ms)
  - Slow-paced games
  - Medium-paced games
  - Fast-paced games

- Network delay under different traffic conditions:
  - No cross-traffic
  - 1 user max. UL
  - 2 users max. UL
  - 3 users max. UL
Thank you.
Questions?
teemu.kamarainen@aalto.fi