

HYPER-LINKED COMMUNICATIONS

WebRTC enabled asynchronous collaboration

Friday 3rd June, 2016

Henrique Rocha

Instituto Superior Técnico

Universidade de Lisboa

henrique.rocha@tecnico.ulisboa.pt

Supervisors: Ricardo Pereira & Paulo Chainho



OVERVIEW

1. Introduction
2. Related Work
3. Architecture
4. Implementation
5. Evaluation
6. Conclusions
7. Future Work

INTRODUCTION

Written communication could never replace face to face communication.

“No computer in our lifetimes will ever rival a human voice’s capacity to conveying rich and complex social and emotional meaning”

— Geddes, Martin

Today, we can achieve more.

PROBLEM STATEMENT

An application that provides a collaborative environment and a way to remember our past communications would be a strong tool.

Real-time communication applications can make a difference on business, education and health sectors.

THESIS GOALS

Allow multi party conference calls.

Record and playback interactive video.

Create a collaborative environment

Use only standard technologies like JavaScript, WebRTC, HTML5 and CSS3.

RELATED WORK

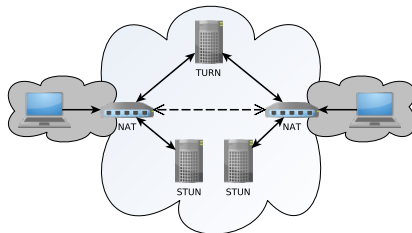
STATE OF THE ART OVERVIEW

1. Connection Establishment
2. Streaming audio and video
3. Overlay communications with content
4. Collaboration Environment

CONNECTION ESTABLISHMENT

- Network Address Translation

- STUN + TURN = ICE



WebRTC (Web Real-Time Communications)

- Access to camera, microphone and screen*
- Peer to Peer file and stream sharing
- Standardized protocols
- No plug-ins required



* requires installing a plug-in yet.

OVERLAY COMMUNICATIONS WITH CONTENT






- **Concepts:** HyperMedia & Detail on Demand
- **Implementations:** HyperCafe & HyperHitchcock



Table: Comparison between Operational Transformation libraries

Library	Server Freedom	Storage Freedom	Operations
ShareJS	✗	✗	text+objects
TogetherJS	✗	✓	text+objects
Goodow	✗	✗	text+objects
Etherpad Lite	✗	✗	extendable
OT.js	✓	✓	text

OVERVIEW

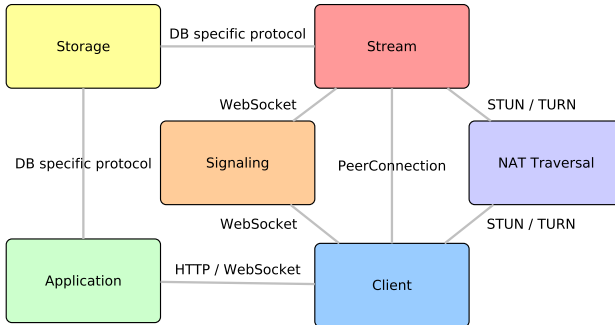
					
Name	Skype	Hangouts	Jitsi	Kurento	Our proposal
Technology	Proprietary	WebRTC ¹	WebRTC	WebRTC	WebRTC
Development	✗	✓ ²	✓	✓	✓
Audio/Video	✓	✓	✓	✓	✓
Text Messaging	✓	✓	✓	✗	✓
Collaborative Editor	✗	✓	✓	✗	✓
File sharing	✓	✓	✓	✗	✓
Recording & Playback	✗	✗	✗	✓	✓
Interactive Content	✗	✗	✗	✗	✓

¹ requires installing a plug-in on non chrome web browsers.

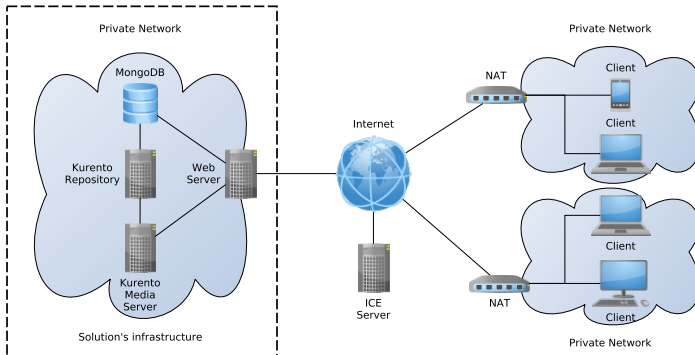
² allows the development of extensions.

ARCHITECTURE

MODULES

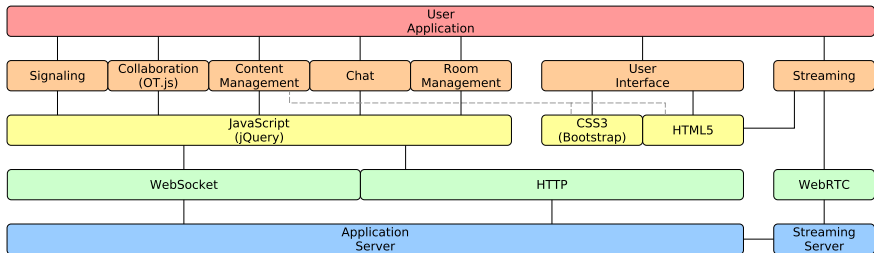


SYSTEM INFRASTRUCTURE



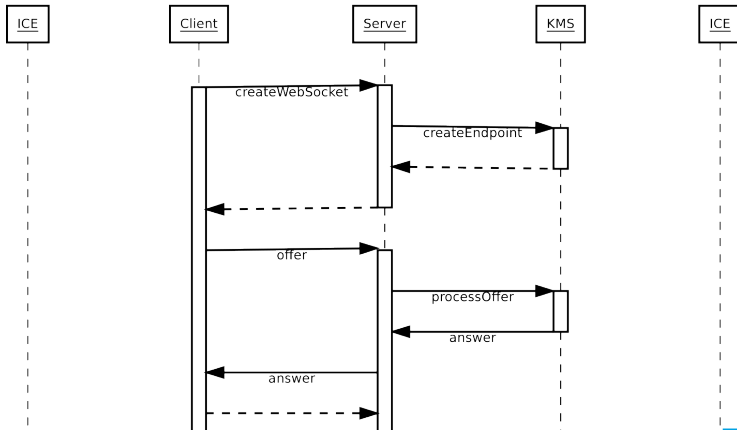
- **Signaling Server & Web Server:** Play Framework
- **Stream Server:** Kurento Media Server
- **Storage:** MongoDB & Kurento Repository
- **NAT Traversal:** Public STUN Servers

APPLICATION ARCHITECTURE

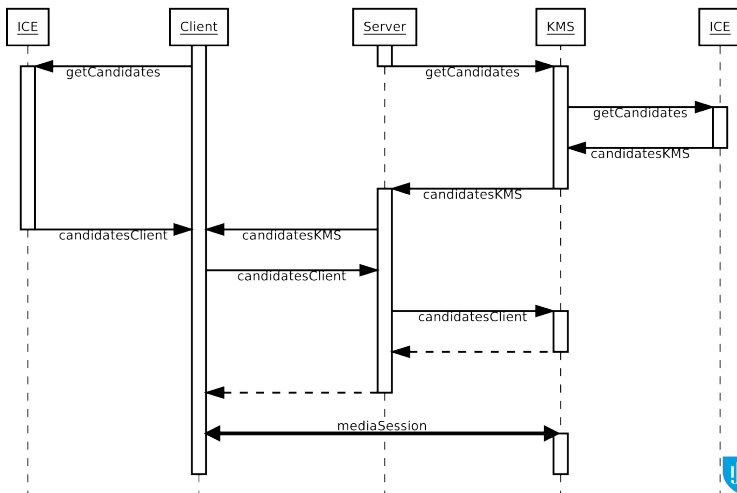


IMPLEMENTATION

SIGNALING PROTOCOL - PART 1

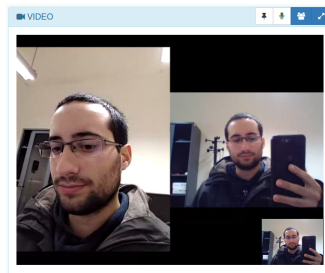
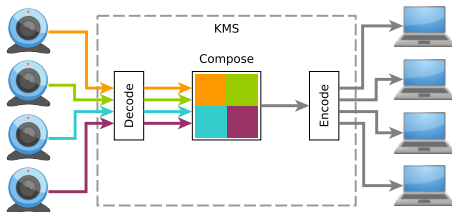


SIGNALING PROTOCOL - PART 2



STREAM OPERATIONS



- Server-side recording to database (Kurento Repository).
- Server-side stream composition.
- Speech detection.




HYPER-CONTENT

- Create & Search content
- Scheduler
- QR codes
- Security concerns

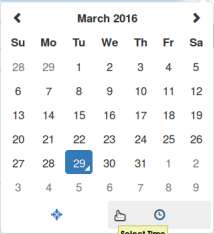
CONTENT EDITOR





Begin: (timeline )
29/03/2016 04:56:40 

End: (timeline )
29/03/2016 04:56:41

Content:
Hello World!

Is Caption: ☒



- Playback recordings
- Create time hyper-links & annotations
- Search annotations & content

- Instant text messaging
 - WebSockets
- File sharing
 - HTTP file upload
 - stored in the database
- Collaborative text editor (OT.js)
 - retain
 - insert
 - delete

HyperChat

Search

FRIENDSHIP REQUESTS

- Cookie Monster
- Kermit the Frog
- Dora Korolczyk
- Gongalo Rocha

TIMELINE

Annotations

Time annotation	
Recordings	
2 June 00:26	2 June 00:27

VIDEO

Time annotation

A simple subtitle

CONTENT EDITOR

Begin: [Timeline]

02:06/2016 00:23:47

Duration:

01:00:10

Content:

A simple subtitle

Is Caption: ✓

Recent Chat History

hello!

Henrique Rocha |
2016-06-01T23:25:43.173Z

Enter Message

GROUP MEMBERS

Add member +

- Henrique Rocha
- Cookie Monster
- Kermit the Frog

Collaborative Editor

```

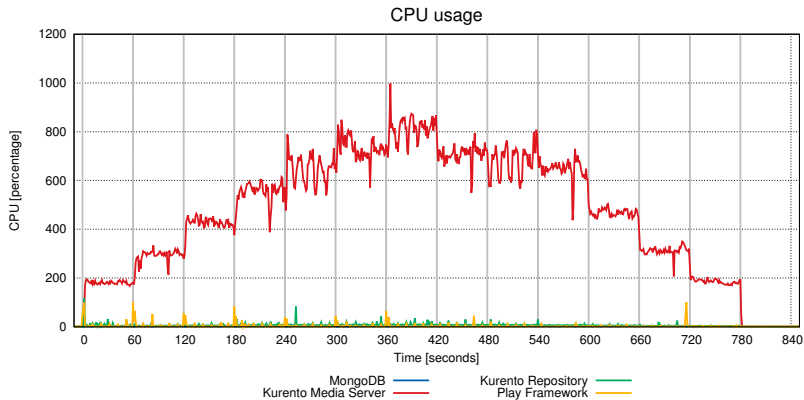
1 this is our
2 collaborative
3 editor
    
```

Save

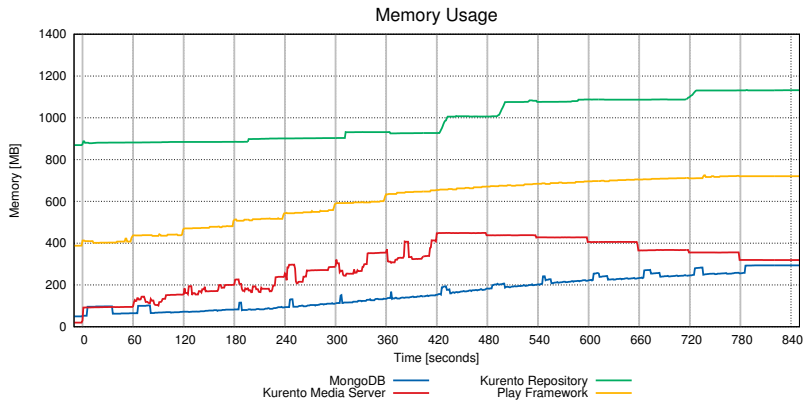
Delete QRCode Preview Save

EVALUATION

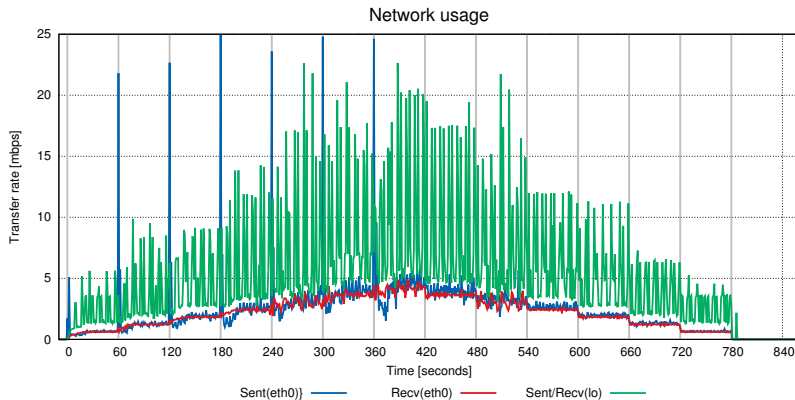
PERFORMANCE TESTS AT SERVER - CPU USAGE



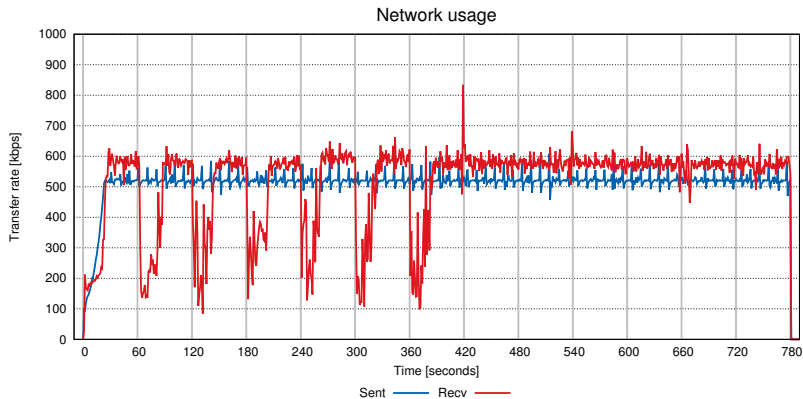
PERFORMANCE TESTS AT SERVER - MEMORY USAGE



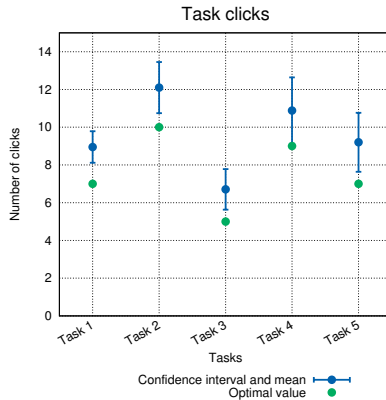
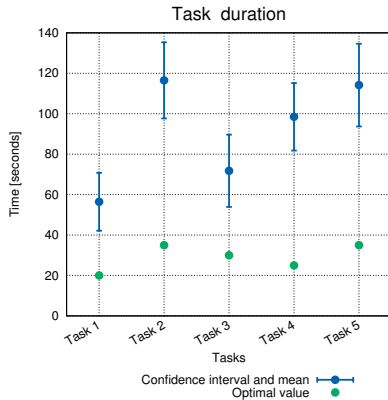
PERFORMANCE TESTS AT SERVER - NETWORK USAGE



PERFORMANCE TESTS AT CLIENT - NETWORK USAGE

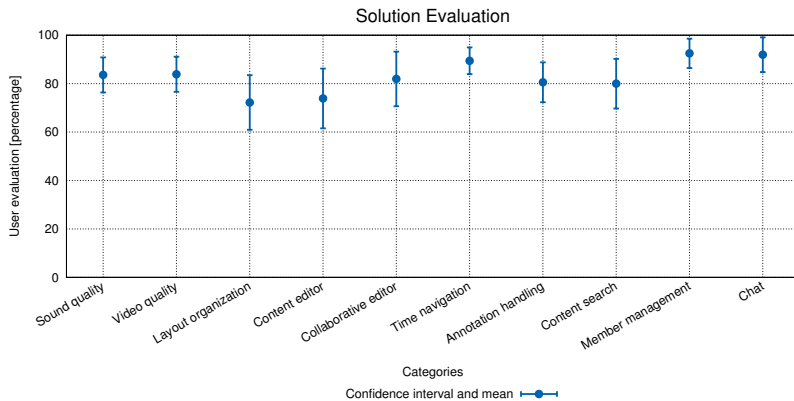


USABILITY TESTS



- Difficulty per task.
- Errors per task.

OVERALL EVALUATION



CONCLUSIONS

CONCLUSIONS

- New usage scenarios for communication and collaboration applications.
- Enrich communications using hypermedia concepts. Record, playback and collaboration features.
- Prototype implementation and testing.

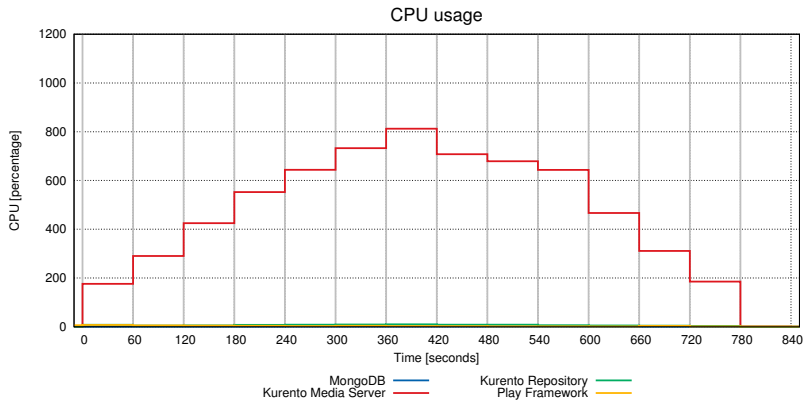
FUTURE WORK

FUTURE WORK

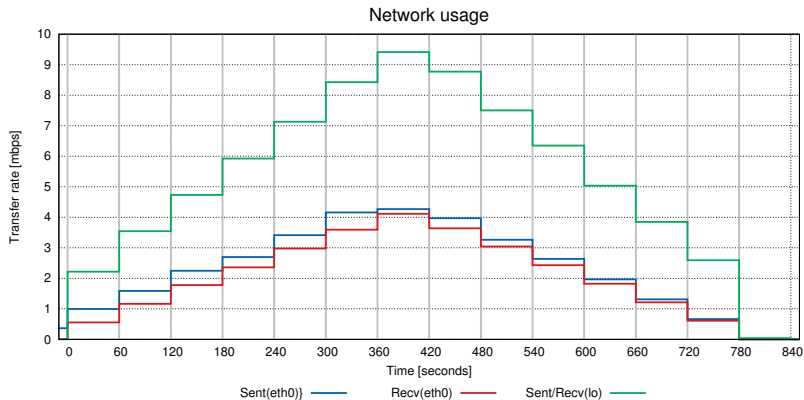
- Implement fast-forward playback.
- Improve solution's security.
- Scale our solution to multiple servers.

Questions?

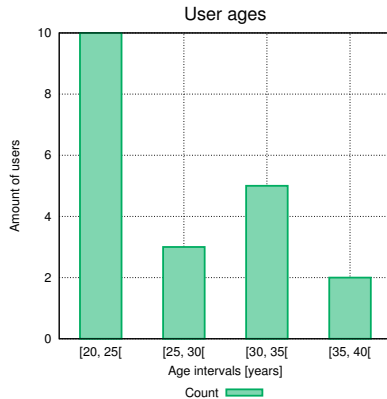
PERFORMANCE TESTS - CPU (AVERAGE)



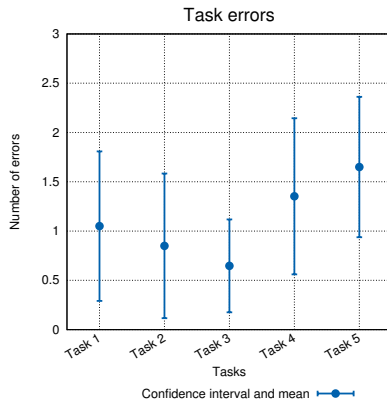
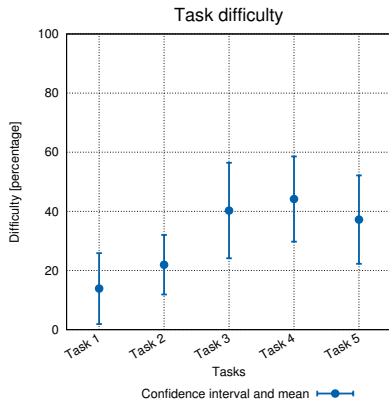
PERFORMANCE TESTS - NETWORK USAGE (AVERAGE)











USER INTERFACE TESTS



FIVE TASKS



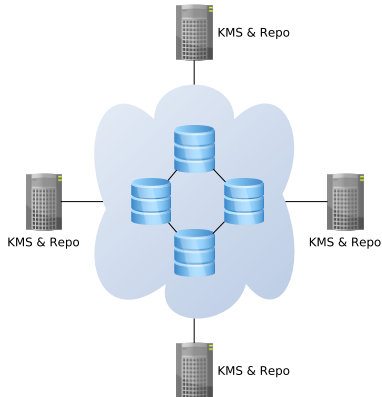
WEB BROWSER SUPPORT

								
	Canary	Chrome	Opera	Nightly	Firefox	Bowser	Edge	Safari
PeerConnection API	Green	Green	Green	Green	Green	Green	Yellow	Red
getUserMedia	Green	Green	Green	Green	Green	Green	Green	Red
dataChannels	Green	Green	Green	Green	Green	Green	Red	Red
TURN support	Green	Green	Green	Green	Green	Green	Green	Red
Echo cancellation	Green	Green	Green	Green	Green	Green	Green	Red
MediaStream API	Green	Green	Green	Green	Green	Green	Green	Red
mediaConstraints	Yellow	Yellow	Yellow	Green	Green	Yellow	Yellow	Red
Multiple Streams	Yellow	Yellow	Yellow	Green	Green	Green	Red	Red
Simulcast	Yellow	Yellow	Red	Yellow	Red	Red	Yellow	Red
Screen Sharing	Yellow	Yellow	Red	Yellow	Yellow	Red	Yellow	Red
Stream re-broadcasting	Green	Yellow	Yellow	Green	Green	Red	Red	Red
getStats API	Yellow	Yellow	Yellow	Green	Green	Red	Green	Red
ORTC API	Red	Red	Red	Red	Red	Red	Green	Red
H.264 video	Yellow	Red	Red	Green	Green	Green	Yellow	Red
VP8 video	Green	Green	Green	Green	Green	Green	Red	Red
Solid interoperability	Green	Green	Green	Green	Green	Green	Yellow	Red
srcObject in media element	Green	Yellow	Yellow	Green	Green	Red	Green	Red
Promise based getUserMedia	Yellow	Yellow	Yellow	Green	Green	Green	Green	Red
Promise based PeerConnection API	Yellow	Yellow	Yellow	Green	Green	Green	Yellow	Red
WebAudio Integration	Green	Yellow	Yellow	Green	Green	Red	Yellow	Red
MediaRecorder Integration	Green	Green	Red	Green	Green	Red	Red	Red
Canvas Integration	Red	Red	Red	Green	Green	Red	Red	Red
Test support	Green	Green	Red	Green	Green	Red	Green	Red

- Subset of actions implemented and triggered by messages.
- Defined actions on demand, requires detecting malicious code.
 - EarlyBird - machine learning techniques.

SCALABILITY

- Kurento Media Server and Kurento Repository in the same machine.
- Fast data transfer on localhost.
- Database bottleneck



- Implement fast forward on Kurento Media Server.
- Convert videos in real time with a new playback velocity using *ffmpeg*.
- Convert videos after recording with predefined playback velocities.

- MongoDB - Always receiving data, caches inserted data on RAM. Checkpoints data to disk every 60 seconds or when journal data exceeds 2GB.
- Play Framework - JVM uses memory recycling techniques instead of releasing to operating system.
- Kurento Repository - Caching videos.
- Kurento Media Server - Releases some memory and also recycles some of it.