

# Hyper-linked Communications: WebRTC enabled asynchronous collaboration

Michael Shell  
School of Electrical and  
Computer Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332-0250

Email: <http://www.michaelsshell.org/contact.html>

Homer Simpson  
Twentieth Century Fox  
Springfield, USA  
Email: [homer@thesimpsons.com](mailto:homer@thesimpsons.com)

James Kirk  
and Montgomery Scott  
Starfleet Academy  
San Francisco, California 96678-2391  
Telephone: (800) 555-1212  
Fax: (888) 555-1212

## Abstract

*The Hyper-linked communications concept applies much of the hypermedia concepts, widely used on Web content. This paradigm allows to synchronize, structure and navigate communication content integrated into voice and video calls.*

*Voice and image together can express emotions like no other medium can. With hypermedia concepts, we can add more value to conference calls.*

*WebRTC technology allows real-time communications between web browsers without the need to install additional software. The nature of web browser applications already follows the hypermedia concept, which makes WebRTC the ideal technology to apply the hyper-linked communications concepts. The web browser platform provides an abstraction layer that makes it possible to create applications that run independently from the operating system. The native support for WebRTC in operating systems extends its usage to outside the web browser, allowing for the exploration of functionalities for which web browsers provide poor support, such as video recording and massive information storage.*

*Our goal was the development of an application targeted to the web platform, resorting to WebRTC, that leveraged the hyper-linked communications by providing a conference environment enriched with multiple media types, collaborative text editors, time*

*annotations, instant messaging and a mechanism to superimpose hyper-content to video.*

*In this document, we present the current State Of The Art in hyper-linked communications and related technologies, propose and implement an architecture for an hyper-linked communication application based on WebRTC. This work was evaluated by users, who reported that they liked to use it and thought it to be extremely innovative.*

## 1. Introduction

### 1.1. Background

Since the early days of Human History, we tried to communicate over far locations, from smoke signals to letters delivered by messengers. Real-time communications were limited or even nonexistent. Despite all the efforts made to improve communications, written communication could never replace face to face communication. With the advent of the telephone network, communications have taken a very important step for us to feel more connected with whom we communicate. Still, only the human voice was not enough, and the invention of cameras and consequent video digitalization were a huge step for real-time communications.

In the past, handwritten documents were limited to a writer per page at a time. Writing a book collab-

oratively was a difficult task due to synchronization between writers. Today, we can achieve more. It is possible to write a document collaboratively, correct spelling mistakes without wasting physical resources, restructure text at any moment, add a video to a newspaper article and more. Although much of what was said seems banal nowadays, none of this was possible before the computer's invention.

As Martin Geddes states[?], "No computer in our lifetimes will ever rival a human voice's capacity to conveying rich and complex social and emotional meaning". Although nothing replaces the physical contact with a person while we communicate, we are at a time when we can do more than just visual and verbal communication. Hypermedia can be added to video and voice in order to extend its value. The concept of structured voice and video synchronized with hypermedia is called hypervoice[?].

As communications technologies appeared, we adapted the way we communicate. The purpose of this project is not the replacement of the current video and audio communications, but to enrich them with hypermedia content and make them a more natural and easy to learn process.

With the advent of WebRTC and its successive integration with web browsers, it became possible to develop video conference web applications without plugins, this presents a range of possibilities on what can be implemented using already existing web technologies.

Furthermore, real-time communication applications can make a significant difference on business, education and health sectors by providing tools for developing teaching and learning online, teamworking and socializing web applications.

## 1.2. Proposed Solution

For multiple reasons, we often need to repeat or postpone some of our tasks, some people tend to forget what they hear or see.

Therefore, a real-time system is a huge source of information that requires much attention from its users. In this context, an application that provides a way to remember our past communications would be a strong tool for not only to catch what we lost but also to enhance our knowledge.

Our goal in this project is to develop an application targeted to the web platform, resorting to Web Real-Time Communication (WebRTC), that leverages the hyper-linked communications by providing a video conference environment enriched with interactive and non-interactive discrete media types such as images,

subtitles, forms and all types of content that can be added using HyperText Markup Language (HTML)<sup>5</sup>, Cascading Style Sheets (CSS)<sup>3</sup> and *JavaScript* including continuous media types such as video, music and animations.

One of the key features of this project is the ability to navigate in time in order to reproduce the conversation again or introduce hyper-content to it such as time annotations, interactive lists of topics and subtitles. In this context we also provide a simpler method for creating and synchronizing hyper-content using *QR codes*.

In addition to this conference environment, which provides different functionalities than traditional conference environments such as *Skype* and *Google Hangouts*, we also enable a collaborative text editor and a chat that supports sending time hyper-links and files to conference participants.

Furthermore, another relevant feature is the possibility to compose multiple video streams into a single one, which enables adding more users to conference rooms without impacting on clients performance. Users can change to individual streams on demand or automatically to the talking users.

## 1.3. Thesis Contribution

Making it clear, this project aims to complement current audio, text and video communications in order to create rich and collaborative interfaces with the ability to add more content on a future time (e.g. creating time annotations for improving content search) in order to increase its value. It is also important to highlight another goal of this project which is the ability to navigate in time by rewinding communications, fast-forward and jump to certain points.

A web application with an easy to learn user interface was developed to accomplish solving our problem. Our application, unnamed yet, is targeted at web browsers that are compatible with only standard technologies like JavaScript, WebRTC, HTML5 and CSS3. Any additional plug-in was avoided, *JavaScript* libraries were preferred as they can be downloaded on the fly.

We have presented an architecture that can meet our goals, implemented the respective prototype and tested it with real users and performance benchmarks.

According to Martin Geddes, the quality of the interaction worsens as the number of users increase[?]. In our testing phases we will quantify and qualify the impact of increasing users on the interface and performance of our prototype.

All the problems faced during the development and limitations were reported on the thesis so that a future project better than ours can be easily and better developed.

## 2. Outline

This rest of this document is structured as follows:

- **Chapter ??** describes the previous work in the field.
- **Chapter ??** describes the system requirements and the architecture for an Web Application that fulfills the goals of this thesis.
- **Chapter ??** describes the implementation of our Web Application and the technologies chosen.
- **Chapter ??** presents the evaluation tests performed and the corresponding results.
- **Chapter ??** summarizes the work developed and proposes future work.

mds  
September 17, 2014

## **2.1. Subsection Heading Here**

Subsection text here.

**2.1.1. Subsubsection Heading Here.** Subsubsection text here.

## **3. Conclusion**

The conclusion goes here.

## **Acknowledgments**

The authors would like to thank...