Boyan Tabakov

Techno-Economic Feasibility of Web Real-Time Communications

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 07.01.2014

Thesis supervisor:

Prof. Heikki Hämmäinen

Thesis advisor:

M.Sc. (Tech.) Tapio Levä

Aalto University
School of Electrical Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Boyan Tabakov

Title: Techno-Economic Feasibility of Web Real-Time Communications

Date: 07.01.2014 Language: English Number of pages:8+71

Department of Communications and Networking

Professorship: Communications Ecosystem Code: S-38

Supervisor: Prof. Heikki Hämmäinen

Advisor: M.Sc. (Tech.) Tapio Levä

WebRTC is an ongoing effort to build an open framework for real-time audio and video communication capabilities that turn Web browsers, and other clients supporting it, into a platform for person-to-person communication. Previously, real-time communication (RTC) has been achievable in the Web browser only by installing third party software. WebRTC brings native support for RTC to the Web browsers and exposes it freely to web developers via standardized JavaScript API. This brings RTC as a feature to the Web, which can foster further innovation.

This thesis studies the techno-economic feasibility of WebRTC with the help of a framework for feasibility analysis of Internet protocols, developed by Levä and Suomi (2013). To provide input for the framework, we conduct an interview study, as well as research of available Web resources. Further, we explore what market opportunities may arise, provided that WebRTC is successfully adopted. To do that, we use Value Network Configurations as a tool for studying and visualizing the possible relationships between market players and the roles they assume in the ecosystem.

We find that WebRTC is a feasible technology in its basic, but highly relevant use case of one-to-one browser-to-browser communication. While we discover a number of unresolved challenges, we do not see any insurmountable obstacles that would prevent WebRTC adoption. WebRTC opens up opportunities for companies that would use it directly to deliver an RTC service, but also creates space for WebRTC PaaS providers in the market. Additionally, WebRTC interconnecting with legacy systems, such as PSTN or PLMN, opens up opportunity for telecom operators to explore creating new ways of communication for their customers.

Keywords: WebRTC, RTCWEB, Web, real-time communications, Internet evolution, Value Network Configurations, VoIP, techno-economic feasibility

Preface

This Master's Thesis has been written as partial fulfillment for the Master of Science Degree at Aalto University, School of Electrical Engineering. The work has been carried out in the Department of Communications and Networking.

I wish to express my gratitude to all the people that have supported me in my work. First of all, I would like to thank Professor Heikki Hämmäinen for providing me the opportunity to write this thesis under his guidance and for piloting and developing the excellent Master's Programme in Communications Ecosystem, in which I completed my studies. I would like to thank Tapio Levä for his invaluable advice and support throughout the entire research process. I am also extremely grateful to all the interviewees who spent some of their valuable time in conversation with me and contributed greatly to the completion of this research.

I would like to thank my employer, Futurice, for sponsoring my work and for being truly the best place to work in all of Europe. Special thanks to Mats Malmstén, Sirkka-Liisa Rotinen and Risto Sarvas, who showed me that Futurice does care for its employees.

I would also like to thank Alexander Zahariev for the encouragement and the pleasant conversations we had many times over lunch while I was working on this thesis.

My very special thanks belong to my closest and dearest Ksenia, who has always given me strength and supported me through all my work. Thank you!

Finally, I would like to thank all my family and friends. Especially my parents, Milena and Vesselin, for their selfless support throughout my life and the best upbringing that one could hope for.

Helsinki, 16.12.2013

Boyan Tabakov

Contents

Abbreviations

API	Application Programming Interface
B2B	Business to Business
B2C	Business to Consumer
DTLS	Datagram Transport Layer Security
IaaS	Infrastructure as a Service
ICE	Interactive Connectivity Establishment
IdP	Identity Provider
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ISP	Internet Service Provider
JSEP	JavaScript Session Establishment Protocol
MPEG	Moving Picture Experts Group
NAT	Network Address Translation
OTT	Over-The-Top Content or Service
PaaS	Platform as a Service
PLMN	Public Land Mobile Network
PSTN	Public Switched Telephone Network
P2P	Peer-to-peer
RCS	Rich Communication Services
RTC	Real-Time Communications
RTMFP	Secure Real-Time Media Flow Protocol
RTP	Real-time Transport Protocol
SCTP	Stream Control Transmission Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SRTP	Secure Real-time Transport Protocol
STUN	Session Traversal Utilities for NAT
TCP	Transmission Control Protocol
TURN	Traversal Using Relays around NAT

UC	Unified Communications
UCS	Unified Communications Solutions
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VNC	Value Network Configuration
VoIP	Voice over IP
WebRTC	Web Real-Time Communications
WG	IETF Working Group
W3C	World Wide Web Consortium
XMPP	Extensible Messaging and Presence Protocol

Chapter 1

Introduction

The World Wide Web (WWW, the Web) is the most widely known and recognized system that is accessed over the Internet. Furthermore, for the majority of Internet users, the word "Internet" is equivalent to the Web (D'Esposito and Gardner, 1999). For those, the Internet *is* what you access via a Web browser. The two are further interlinked as development of the features and services that the Web provides may have effect across other parts of the Internet ecosystem, for instance other systems, service providers, vendors, corporate and consumer users. For that reason, the evolution of the Web is a key component in the evolution of the Internet itself.

Initially the Web, as well as the Web browser — the interface the Web is accessed with — were plain text. Then, one of the first major milestones in the evolution of the Web was the introduction of the Mosaic Web browser, which had a graphical user interface and allowed graphics, together with text to become commonplace in the documents on the Web (Vetter et al., 1994). Later, development in the modern Web browsers and supporting technologies brought multimedia to the Web. Video and audio content, still images and animations, mixed together in interactive Web sites, became a norm.

However, the rich media content is mostly just that — static *content* that is pre-produced and published, then delivered via the Web to its target recipients. The Web, on the other hand, has become increasingly a platform for communication, driven by the rise of the social networks, a venue where many express themselves and where many share with their friends, family or the general public various bits and pieces of their lives. Regardless, whenever real-time communication is required, without the help of additional software the Web can generally offer just text-based instant messaging.

Web Real-Time Communications, or WebRTC, is an ongoing effort to eliminate this limitation of the Web, driven by several major browser vendors (Google, Mozilla, Microsoft, Opera) and other well-known companies

(Cisco, Ericsson, etc.). WebRTC is an open framework for *real-time audio and video communication* capabilities that turn Web browsers, and other clients supporting it, into a universally accessible platform for person-to-person communication (Jennings et al., 2013; Loreto and Romano, 2012). While real-time voice and video is not new to the Internet, so far it has been achievable in the Web browser only by installing third party software, such as Adobe Flash or Skype plug-ins, neither of which are — even though widespread — as ubiquitous as the Web browser itself. WebRTC brings native support for RTC to the Web browsers and exposes it freely to web developers via standardized JavaScript API (Jennings et al., 2013).

WebRTC is arguably a key enabler for the next stage of Web — and Internet — evolution. As a common open platform, it would allow any Web site or Web-based service or application to easily add voice or video communication functionality. This prospect may seem most obvious in the context of the big social networks, such as Facebook, Twitter, VK and LinkedIn, but may also open up unexpected opportunities as it leaves the way open for innovation.

1.1 Research question

Industry experts often opine that WebRTC has the potential of being a disruptive innovation (see, for instance, Bubley, 2013d; Kelly, 2013; Levent-Levi, 2012b). However, given the recent introduction of the technology, very few academic publications have been done on WebRTC. For example, Loreto and Romano (2012) and Jennings et al. (2013) provide overview of the technology and the current state of development, Amirante et al. (2013) look into integration between WebRTC and SIP-based systems and A. Johnston et al. (2013) explore issues, specific to use of WebRTC in enterprises.

Therefore, this thesis aims at expanding research on the topic with a more holistic and economic view of WebRTC by evaluating the technology's techno-economic feasibility. Furthermore, we analyze market opportunities that may arise as a result of its adoption and discuss possible strategic options for some of the key stakeholders in the market. The questions that we will answer are as follows:

- Is WebRTC techno-economically feasible?
- What market opportunities will likely occur as a result of WebRTC adoption for the relevant stakeholders in the ecosystem?

In pursuit of answering these questions, we set the following objectives for this research:

- identify the most likely use cases of WebRTC;
- identify technical and economic challenges that the technology must solve, in order to be feasible;
- identify the stakeholders and Value Network Configurations (VNC) that would form around them;
- analyse the VNCs for market opportunities and explore strategic options for the relevant stakeholders;

1.2 Research scope

The use of real-time communication over the Internet can be split into two main markets — consumer (or business to consumer, B2C) and corporate (or business to business, B2B). The consumer market includes services such as Microsoft's Skype, Apple's FaceTime, Google's Hangouts and Viber. The corporate market includes, for example, enterprise Voice over IP solutions and videoconferencing equipment and services, where companies like Cisco, Polycom, Radvision and Vidyo are well known. In order to build a holistic view of WebRTC, the thesis covers aspects of both the consumer and corporate markets.

Following the Internet protocol feasibility analysis framework (Levä and Suomi, 2013), the thesis will present some technical details regarding the implementation of WebRTC. However, this is not the main focus of the work and will mainly be used to provide a basis for the rest of the thesis. Furthermore, as an exhaustive study of all use cases of WebRTC would be prohibitively lengthy, we scope our feasibility analysis largely to the basic use case of one-to-one, browser-to-browser communication on desktop computers within a single communications service.

1.3 Methods

Levä and Suomi (2013) construct a comprehensive framework for techno-economic feasibility analysis of Internet protocols. The framework looks at the respective protocol in the context of its use cases, technical architecture and deployment options, or in other words in the context of a full technological solution or service that implements the protocol. The thesis will rely on this framework to answer the first question, presented in Section 1.1. Additionally, we will conduct semi-structured interviews and desk research on various Web resources to support the research with real-world information. The main goal is to bring insight from various parts of the industry, as well as the academia.

The analysis part of the thesis will incorporate Value Network Configurations (Casey et al., 2010), building on the results from our interview and desk research study, in order to answer the the second research question.

1.4 Structure

The main part of the thesis continues as follows. In Section 2 the theoretical frameworks that this thesis relies on will be presented. Next, in Section 3 we will provide the technical background required to understand the technology behind WebRTC, as well as its place in the Internet ecosystem. Section 4 explains our research process, how the interviews are conducted and what are the key questions asked. Then, in Section 5 the results from our interview study and desk research are presented, answering to the first research question. The following Section 6 presents the rest of our contribution and analysis — studying the second research question. Finally, Section 7 provides a discussion of the findings in the thesis and gives a brief outlook into possible future research.

Chapter 2

Theoretical frameworks

This chapter presents the main theoretical frameworks that facilitate the analysis in this thesis.

2.1 Internet protocol feasibility analysis framework

Levä and Suomi (2013) have developed a framework that allows for a comprehensive study of Internet protocols, especially aimed at identifying potential deployment and techno-economic challenges early on in the protocol development and standardization process. The framework draws from earlier research in various protocol case studies, such as Multipath TCP (Levä, Warma, et al., 2010) and Host Identity Protocol (Levä, Komu, et al., 2013).

According to Levä and Suomi (2013), the framework is the first attempt to provide a systematic process for studying Internet protocols during their development from both technical and economic perspective. The authors recognize that in order to result in a complete and relevant analysis of the protocol, all major stakeholders should be taken into account in respect to the protocol's impact on them and the incentives they may have for adoption and deployment of the protocol.

The framework defines an iterative process of identifying deployment challenges that comprises of six analysis steps: 1) use case analysis; 2) technical architecture analysis; 3) value network analysis; 4) deployment environment analysis; 5) feasibility analysis; and 6) solution analysis (Levä and Suomi, 2013). Each step presents a set of questions, which once answered provide input for the next step. Ultimately, the solutions analysis in step 6 provides suggestions on how the challenges, identified in the earlier steps could be addressed (Levä and Suomi, 2013). Figure 2.1 illustrates the framework and its steps.

Next, we describe briefly each step of the framework.

Use case analysis

In this step, the aim is to describe the purpose and functionalities of the protocol and the different use cases that are enabled as a result, as well as what the expected benefits of implementing or using the protocol are.

Technical architecture analysis

The step looks at the technical architecture of each use case and the corresponding deployment actions that must take place in order to realize the architecture. Clearly, multiple technical architectures may be possible to consider at the same time. The framework does not impose restrictions on that, but Levä and Suomi (2013) advise that focusing on a single architecture might be better approach, because otherwise a thorough analysis in the following steps might prove to be challenging.

Value network analysis

This step lists the stakeholders and maps the deployment actions, determined in the previous step, to the corresponding stakeholder. Also, here the technical roles that arise from the deployment actions are identified. Using these, the value network of the provided service is built.

Deployment environment analysis

In this step, the environment in which the protocol will be deployed is described. The external factors that may affect the deployment are presented and possible substitutes are listed. The environment analysis typically includes also political, economic and social aspects and possible future evolution. The power of the different stakeholders to affect the deployment of the protocol is also examined.

Feasibility analysis

In this step, evaluation is done on the incentives for the relevant stakeholders and comparison is performed on costs against benefits. Furthermore, possible network effects and their likely impact on adoption are studied and attention is paid to discovering what the deployment challenges for the protocol are.

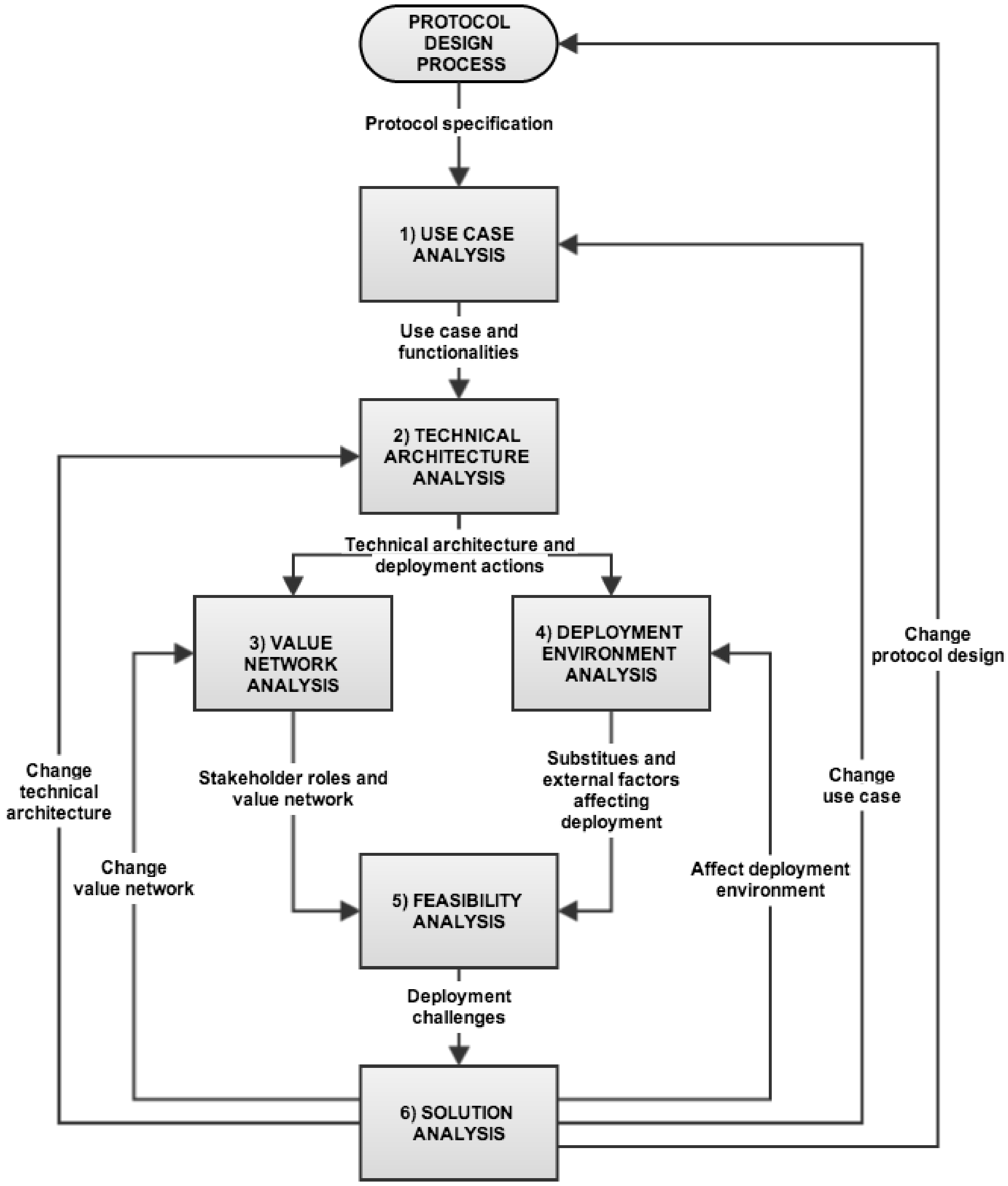


Figure 2.1: Framework for studying the techno-economic feasibility of Internet protocols (Levä and Suomi, 2013).

Solution analysis

In this step, potential solutions are discussed, that may help overcome the challenges listed in the previous step. Typically, solutions can be split in five groups: 1) changing protocol design; 2) changing use case; 3) changing technical architecture; 4) changing value network; and 5) affecting deployment environment.

2.2 Value Network Configurations

Value Network Configurations (VNCs) is a tool for performing value network analysis introduced by Casey et al. (2010). It is a visual way of representing the relationship between the stakeholders (actors) in an industry by mapping together the value network and technical architecture of a value creation activity.

Building the VNC begins with identifying the roles that arise around the technical components of the value creation activity and their interfaces (network protocols, API interfaces, etc.). Casey et al. (2010) define a role as a *"set of activities and technical components, the responsibility of which is not divided between separate actors"*. These roles are then assigned to the actors and the emerging business interfaces between the actors, such as contracts and monetary exchanges, are described (Casey et al., 2010). Figure 2.2 demonstrates the notation used to describe VNCs.

VNCs are a suitable tool for performing the value network analysis in step 3 of the framework, described in Section 2.1 (Levä and Suomi, 2013). The visual representation of the VNC allows for easy mapping of deployment actions to actors and comparison of the different VNCs that can form from the same underlying technical architecture (Levä and Suomi, 2013).

The explicit separation of roles and actors helps deliberate over multiple plausible VNCs. Alternative VNCs can naturally be constructed from the same set of roles and actors by considering different role mappings or altering

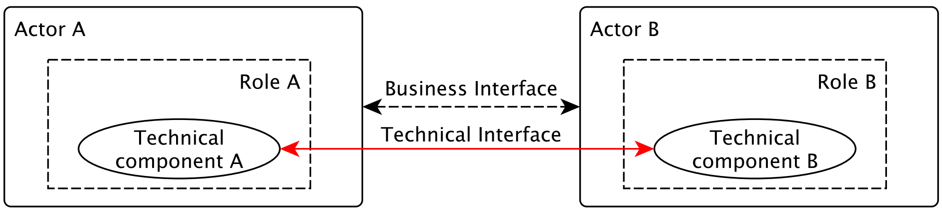


Figure 2.2: Notation for Value Network Configurations (Casey et al., 2010).

the business models — the business interfaces between the actors[1]. Furthermore, the process can also facilitate recognition of business opportunities, for example, when a role is considered economically or strategically valuable and a new actor, i.e. a new player on the market, may aim at fulfilling it (Kostopoulos et al., 2012). Therefore, VNCs are useful also for tackling the second research question (Section 1.1).

[1]The technical interfaces can largely be considered fixed, as they are closely tied to the underlying technical architecture.

Chapter 3

Background

In this chapter we present shortly the current status of real-time communications on the Internet and then introduce WebRTC as a new-coming technology in more detail.

3.1 Internet architecture and real-time communications

This section explains the general architecture of the Internet, provides a definition for real-time communication and describes how such communication is accomplished on the Internet.

3.1.1 Internet architecture

The Internet is a global set of independent, but interconnected computer networks. Due to the use of shared standard communication protocols, these interconnected networks appear to be a single, uniform network of more than 1.2 billion connected hosts (ISC, 2013). (CSTB, 2001)

At the core of the Internet protocol stack is the Internet Protocol (IP) (Deering and Hinden, 1998; Postel, 1981). Figure 3.1 illustrates the "hourglass" model of the Internet protocol stack and shows the place of IP in it (CSTB, 2001). The Internet Protocol is the unifying layer that sits between the various underlying networking technologies and protocols and the different upper layer transport and application protocols. It allows diverse independent networks to interlink seamlessly, forming the Internet (CSTB, 2001).

The Internet could be divided into two parts — the core and edge networks. The core network consists of routers and communication links between

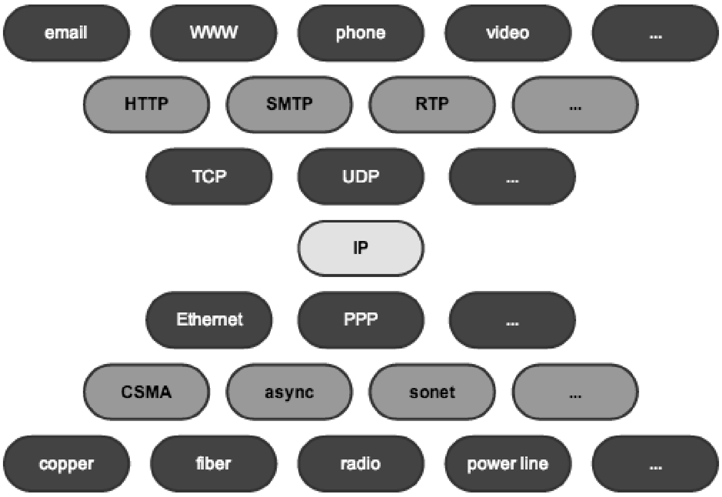


Figure 3.1: Hourglass model of Internet protocol stack (adapted from Zittrain, 2008).

them, which are operated typically by Internet Service Providers (ISPs). The edge network, on the other hand, contains the user-controlled[1] networks and devices. As a result of the hourglass architecture of the Internet and the end-to-end principle (Saltzer et al., 1984), the core network is largely only concerned with protocols up to the IP layer, the network layer, operating without regard of the upper layer protocols and applications. Those higher layer protocols are typically implemented at the end devices, which communicate to one another over the Internet (see Figure 3.2). (CSTB, 2001)

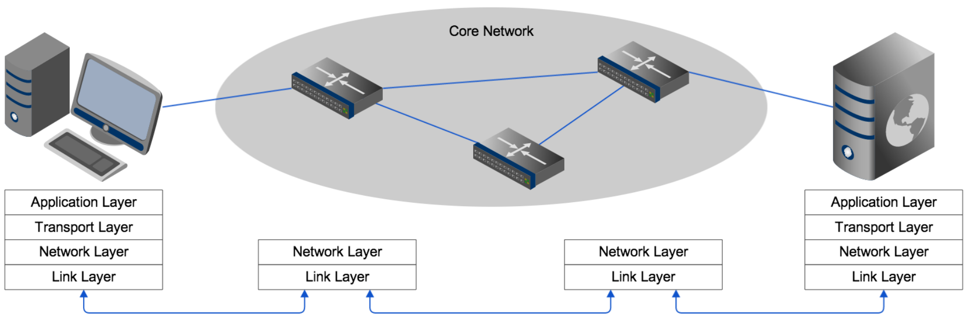


Figure 3.2: End-to-end connectivity over the Internet. Core network elements operate up to the network layer.

[1]User here denotes the customers of ISPs. They could be private individuals, as well as companies, organizations, etc.

3.1.2 Real-time communications on the Internet

In the uppermost layer of the hourglass model sit the various services or applications that utilize the Internet protocol stack. These include the Web, email and media applications, for instance. Applications for real-time communications, such as Google Hangouts, Skype and Viber, are no exception.

Definition and characteristics

Tripathi et al. (2013) define real-time communications (RTC) as *"any mode of telecommunications in which all users can exchange information instantly or with negligible latency"*. Consequently, RTC applications exhibit significantly different characteristics than non-real-time ones (Aras et al., 1994). The most important of these is *timeliness* — the requirement for short latency and minimal delay jitter (Aras et al., 1994; Kopetz, 2011).

The User Datagram Protocol (UDP) is often preferred for real-time applications (Kopetz, 2011; Postel, 1980). Being a connectionless protocol, it offers lower overhead (both delay- and traffic-wise) compared to TCP and the trade-off, related to loss of reliability, can be handled on the application layer, taking better into account the requirements of the specific application (Kopetz, 2011). Furthermore, some loss of reliability is often acceptable for real-time media transfer, because the audio and video codecs in use are typically resilient to packet loss, at least to some extent (for example, see Google, 2013b; Stockhammer et al., 2003; Xiph.Org, 2013).

VoIP and SIP

Sisalem et al. (2013) provide a historical overview of how protocols and services related to Voice over Internet Protocol (VoIP)[2] developed over the years. According to the authors, while a commercial failure, the early VoIP services in the nineties marked the first steps towards real-time communications over the Internet and towards the evolution of the Internet into the universal communications platform it is today.

Session Initiation Protocol (SIP) (Rosenberg, Schulzrinne, et al., 2002) is seen as an important milestone in the development of VoIP. SIP is the signaling protocol that together with already introduced protocols, such as Real-Time Transport Protocol (RTP) (Schulzrinne et al., 2003) and Session Description Protocol (SDP) (Handley and Jacobson, 1998), provided a standardized way for establishing voice or video calls over the Internet. However,

[2] VoIP is not a concrete protocol. It is a collective term for various protocols and technologies that implement voice (or video) service over an IP-based network.

SIP use is not constrained to VoIP — other applications that require session initiation, such as online gaming, can make use of it. (Sisalem et al., 2013)

Nevertheless, SIP-based systems are mainly seen today in corporate environments — for instance in enterprise telephony and videoconferencing solutions by vendors such as Cisco, Polycom and Radvision.

Skype and other proprietary applications

Skype is often given as the most prominent example of successful real-time communication system over the Internet (Sisalem et al., 2013). Rao et al. (2006) attribute the success of Skype to the fusion of two disruptive technologies that it embodies — VoIP and peer-to-peer (P2P). Furthermore, Sisalem et al. (2013) consider Skype's easy-to-use client application, high quality voice and its ability to successfully traverse firewalls and Network Address Translator (NAT) devices as a key factor in its success among consumers. The latter characteristic of Skype is implemented using proprietary protocols. Sisalem et al. (2013) highlight this in comparison with the struggle, at the time, of standardization bodies to agree on the best way to provide similar solution for SIP-based systems.

Reportedly, Skype-to-Skype voice and video calls account for 34% of international telephone traffic (TeleGeography, 2013). However, nowadays other contestants operate in the consumer market as well, with examples including Google's Hangouts service and Viber. Viber, for instance, started as a mobile phone application for voice and video communication or instant messaging (IM), but now offers also desktop client, similarly to Skype (Viber, 2013).

Adobe Flash and RTMFP

Secure Real-Time Media Flow Protocol (RTMFP) is a proprietary protocol, developed by Adobe for use with it's Adobe Flash, Adobe Integrated Runtime (AIR) and Adobe Media Server technologies (Thornburgh, 2013). RTMFP allows real-time communication between users of Flash-based applications.

Running Flash-based applications requires support for Adobe Flash Player. Flash Player is largely installed as a Web browser plug-in. Reportedly, Flash Player is installed on more than 90% of Internet-connected desktop computers and mobile devices (Adobe, 2011). Notably, however, Apple dropped support for Flash on iOS devices in 2010, citing reasons such as the "100% proprietary" nature of Flash-related products; poor usability on touchscreen devices; performance, security and reliability issues, etc. (Jobs, 2010). In addition Jobs (2010) argues that Apple strongly supports open standards for everything related to the Web.

3.2 WebRTC

WebRTC is a solution, currently being developed by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C). It enables browsers to establish peer-to-peer communication channels that can carry media (audio or video) and/or data streams. The control of this communication is given through the WebRTC API to the Web application that the browser is running. A basic example is given in Figure 3.3 (Rescorla, 2013b). There, at request of the application, the users' browsers are able to connect directly to each other and allow the users to communicate in real-time with little or no aid from any intermediate network entities, such as media servers or relays. From the user perspective, this type of communication experience is not new — other applications, most notably Skype, have allowed direct calling between computers already for a time. However, one important change with WebRTC is that such capability is now built-in in the Web browser, which is the most ubiquitous type of software program on Internet-connected computers or devices.

In order to implement this functionality, browsers must be capable of performing several key tasks — connection establishment, NAT traversal, call control, media control, encoding and decoding of media, codec negotiation and control (Eriksson and Håkansson, 2012; Loreto and Romano, 2012). Additionally, it is up for a decision how much of these capabilities need to be

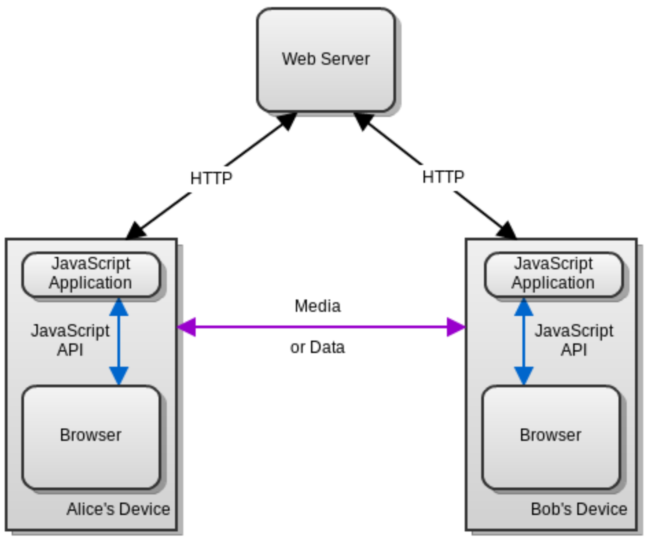


Figure 3.3: Simple case of communication with WebRTC (adapted from Rescorla, 2013b).

exposed to the web application through the API. On one hand there is the strive to keep the API simpler and hide most of the underlying complexities, against allowing fine-grained control to the application by letting it tune many parameters of these operations, on the other.

The IETF and W3C are the two standardization bodies that work closely together on WebRTC. IETF's RTCWEB Working Group is working on selecting the protocol stack for WebRTC, determining how the capabilities listed above should be achieved. The WG develops use case scenarios and identifies requirements for the implementation of WebRTC (Holmberg et al., 2012), as well as describes its threat model and proposes a security architecture to address the security and privacy issues and goals that result from the various use cases (Rescorla, 2013a,b). The W3C, on the other hand, drafts the specification for the API that opens up WebRTC functionality via JavaScript to Web applications running inside the browser (Bergkvist et al., 2012). Cooperation between the two organizations is naturally required, as protocol level issues affect the API and vice versa.

Ultimately, however, it is up to the browser vendors to decide what and how to implement and whether to conform fully to the W3C specification. With Google, Mozilla and Opera backing up WebRTC, there seems to be little doubt as to whether this feature will be widely available, which in turn lays out the path to its success (Eriksson and Håkansson, 2012).

In addition, while WebRTC is primarily targeted at use in a Web browser, it is not limited to that use case. Other applications can implement the technology and become full-featured WebRTC end-points.

3.2.1 Protocol stack

Table 3.1 lists the protocols that the RTCWEB WG has selected for implementation of WebRTC along with their functionality (Alvestrand, 2013; Jesup et al., 2013). Figure 3.4 further illustrates the protocol layering for the media and data path between WebRTC end-points. WebRTC does not introduce new protocols, it rather makes use of existing ones. Furthermore, all these protocols could be implemented in the Web browser and do not require support or modification in the lower layers, such as the operating system. Therefore, there are no special deployment actions required for adoption of WebRTC, other than the support in end-points and availability of required infrastructure. In particular, network elements, like routers and NAT boxes, do not need to be updated.

We next present in some detail each of the three main protocol (and functional) groups — connections establishment and NAT traversal; media transport; and data transport.

Table 3.1: WebRTC protocol suite (Alvestrand, 2013; Jesup et al., 2013).

Function	Protocol(s)
Data transport	SCTP, DTLS
Media transport	SRTP, DTLS-SRTP (SRTP keying)
Signaling	JSEP, other left unspecified
Connection establishment and NAT traversal	ICE, STUN, TURN
IP transport	UDP

Connection establishment and NAT traversal

UDP is the selected transport layer protocol for WebRTC (Alvestrand, 2013). The advantages of UDP over TCP for real-time applications were already discussed in Section 3.1.2.

Media connections in WebRTC (the "calls" from user perspective) follow similar offer/answer model as SIP (Alvestrand, 2013) — the caller party makes a call request, which the callee party either accepts or rejects (Rosenberg and Schulzrinne, 2002). Offer/answer protocols, however, are generally not able to operate on their own in setups where one or both of the communicating nodes are behind a NAT device, because they tend to carry IP addresses and/or TCP/UDP port numbers in their payload. Various NAT traversal techniques exist that are designed to address this shortcoming. The

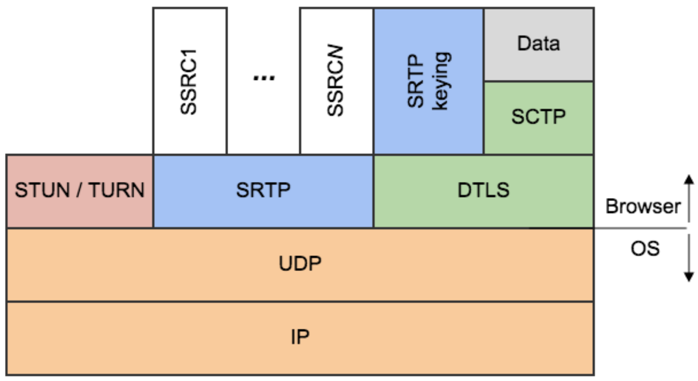


Figure 3.4: WebRTC protocol stack for media and data transport over UDP (adapted from Jennings et al., 2013). Protocols in the same functionality group (see Table 3.1) have the same color.

one selected by RTCWEB WG is Interactive Connectivity Establishment (ICE) (Alvestrand, 2013; Rosenberg, 2010).

ICE allows two hosts to discover IP address and UDP port number pairs usable for establishing a connection between each other, regardless of the network topology and the presence of NAT devices on the path between the hosts, which solves an important requirement for WebRTC (Holmberg et al., 2012). However, in order to operate, ICE mandates the use of Session Traversal Utilities for NAT (STUN) and/or Traversal Using Relays around NAT (TURN) servers (Rosenberg, 2010), i.e. the two communicating hosts require the help of additional infrastructure to establish (and possibly maintain) the peer-to-peer connection between each other. In particular, TURN servers are the bandwidth-demanding piece of infrastructure, because they are used to relay the media or data communication between the peers when too restrictive firewalls prevent the peer-to-peer connection to be established directly.

As described below, both the media and data transport protocols are datagram-based, so ICE facilitates all connection establishment needs in WebRTC.

Media transport

WebRTC requires the use of the Real-time Transport Protocol (RTP) (Schulzrinne et al., 2003) and more specifically, RTP profiles built on top of Secure RTP (SRTP) (Baugher et al., 2004). RTP, together with its data formats, profiles and extensions, provides a flexible framework that can meet a diverse set of requirements. One of the tasks of RTCWEB WG is to specify the subset of RTP add-ons that must be implemented by WebRTC-enabled endpoints in order to ensure interoperability between them, as well as maintain compatibility with already deployed infrastructure. (Perkins et al., 2013)

Notably, the use of SRTP reflects the requirements for confidentiality and privacy of communication, set out by Rescorla (2013b). DTLS-SRTP (Fischl et al., 2010) is the mechanism used to securely establish the secret keys for use in the SRTP sessions (denoted by "SRTP keying" in Figure 3.4) (Rescorla, 2013a).

Several SRTP streams could be multiplexed over a single RTP session. Each of these streams is identified by its synchronization source (SSRC), which is part of the RTP header (see Figure 3.4). (Baugher et al., 2004)

Data transport

Apart from exchange of media streams, browsers can use WebRTC also to send arbitrary data streams to one another. The protocol mandated by RTCWEB WG to accomplish this purpose is the Stream Control Transmission Protocol (SCTP) (Stewart, 2007), encapsulated in DTLS (Rescorla and Modadugu, 2012), providing confidentiality, source authentication and integrity protection for the data channel (Jesup et al., 2013). SCTP allows for reliable and unreliable transmission of data, each of which can be utilized in WebRTC (Bergkvist et al., 2012; Jesup et al., 2013).

3.2.2 Technical architecture

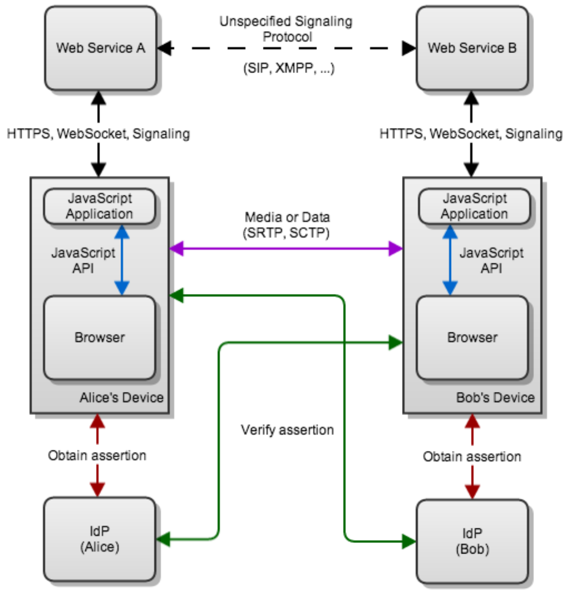
The web application that the user loads in the browser has the central role in controlling a WebRTC session. It controls when a call is established or terminated, who is being called, what user presence information is displayed (e.g. online status), how the media content is rendered and so on (Holmberg et al., 2012; Rescorla, 2013b). All this is accomplished via the WebRTC JavaScript API. Naturally, the application would often mostly respond to user-initiated actions, and some security- and privacy-sensitive operations require explicit user consent (Holmberg et al., 2012).

In the previous section, Figure 3.3 showed a basic use case for WebRTC. However, that is a high-level view, ignoring some of the details that need to be taken into account when describing a more general technical architecture of WebRTC. Some of these were already mentioned, e.g. NAT devices and nodes facilitating NAT traversal. The other important items are the standardized signaling in WebRTC, or lack thereof (Loreto and Romano, 2012), federation and identity provision.

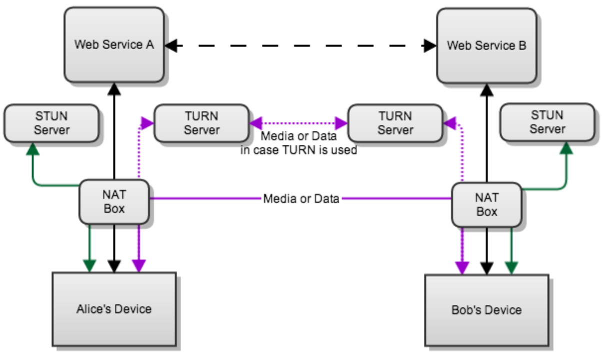
Figure 3.5 gives a more detailed representation of a two-party communication setup, including the NAT traversal infrastructure and Identity Providers. The illustrated setup is symmetric, but that does not limit its generality. For example, Bob might or might not be behind a NAT device (see Figure 3.5b), or Alice and Bob might use the same Identity Provider (see Figure 3.5a) — either one would be supported.

Signaling

Signaling is a core activity inside a communications system, concerned with establishment and control of a communication session. Signaling can be in- or out-of-band, depending on whether the signaling uses the media channel, or a dedicated one.



(a) Detailed technical architecture.



(b) Detailed media path with NAT Traversal.

Figure 3.5: Technical architecture for two-party communication with WebRTC (Rescorla, 2013a; Rosenberg, 2010).

In WebRTC the standardization is deliberately limited to specifying and controlling the media plane, leaving the signaling plane largely up to the application to implement. This is a conscious decision made in hope to free the service providers and allow them to use whatever signaling protocol may suit their application best, considering any specifics of that application. This may be an existing protocol like SIP or XMPP, or alternatively a custom purpose-built protocol that would allow for innovative features. (Loreto and Romano, 2012; Uberti and Jennings, 2013)

Even if the exact way the signaling is implemented is left to the application developers, certain steps for establishing a session need to take place in order to ensure proper operation of the full stack of protocols. Therefore, the RTCWEB WG has decided to specify the JavaScript Session Establishment Protocol (JSEP). The protocol allows signaling state to be decoupled from the browser, i.e. put in control of the application, by providing a way for the application to obtain and set the multimedia session descriptions (SDP messages) needed to establish the actual multimedia channel. (Uberti and Jennings, 2013)

Figure 3.5 shows a general case where the Web applications that each of the two communicating parties use are different, possibly coming from different service providers. Due to the lack of specification of the signaling plane, however, this is not significantly different situation compared to the simplified case from Figure 3.3 strictly from WebRTC operation perspective. The media or data connection is still formed directly between the end-points and the SDP messages anyway travel between them in an unspecified way. At the same time, the freedom to implement signaling in an arbitrary way has some implications on a higher level — it becomes less likely that two service providers would have compatible implementations. This means that interdomain federation — making it possible for users of service A to call users of service B — would require explicit agreement and support from both sides. At a minimum, gateways that can translate the exchanged signaling messages to and from the service's native format would need to be in place.

Identity provision

As mentioned above, using DTLS in the media and data transport protocols for WebRTC allows the Web browsers to establish a secure and private communication channel between themselves, using cryptographic keys that may even be ephemeral (generated by the browser on-demand, possibly for one-time use) (Fischl et al., 2010; Rescorla, 2013a). However, generally the need still remains for the end-users to be able to verify the *identity* of the person or system they are communicating with (i.e. Alice is calling and not someone else). Therefore, communicating parties need to have relationship with some Identity Provider (IdP) that can assert their identity per request of the other party. Naturally, an IdP may be provided as part of the calling service (e.g. Alice has account for the calling service and uses that account name as her identity), but can also be offered by a third party that the user has relationship with — for instance a social network or another service[3]

[3] For example Federated Google Login, Facebook Connect, OpenID providers, etc. (Ko et al., 2010; Rescorla, 2013a).

(Rescorla, 2013a).

When Alice is calling Bob (see Figure 3.5a), Alice's browser contacts Alice's IdP to obtain a token binding a fingerprint of its cryptographic key to Alice's identity. This token is then sent over the signaling plane to Bob's browser, which can in turn contact Alice's IdP to verify the token. This procedure is repeated symmetrically with Bob's IdP. (Rescorla, 2013a)

Note that Alice's and Bob's IdP may be the same, but that does not affect the message flow. Furthermore, one or both IdPs may be missing when anonymous communication is allowed (e.g. a use case where visitors on a website can contact a sales person via a call button).

Interconnection with legacy systems

Browser-to-browser communication is the main use case in WebRTC. However, Holmberg et al. (2012) allow the possibility for interconnection to other systems, for example a legacy VoIP system, PSTN (Public Switched Telephone Network) and PLMN (Public Land Mobile Network). This is achieved by introducing a gateway between WebRTC and the non-WebRTC system. In case of SIP-based VoIP systems, the gateway might only provide signaling translation (as both systems use the SDP offer/answer model), or may also bridge the media plane, in case media transcoding is required when the legacy system does not support the same codecs as the WebRTC end-point (A. B. Johnston and Burnett, 2012).

Multi-party communication

The peer-to-peer architecture that WebRTC follows fits well with the general use case of a two-party communication. Naturally, it does not impose a limit to the number of communicating parties — by establishing a full-mesh topology, any number of end-points can communicate jointly.

When the number of nodes increases, however, the constraints in bandwidth and processing power can be a limiting factor, as each end-point must send the its media streams to, and must receive the incoming streams of all other participants. In such cases Holmberg et al. (2012) and Perkins et al. (2013) lay out the possibility for using a central server that can relay and/or mix the media streams of the participants.

Chapter 4

Methods

This chapter introduces the methods and sources used for gathering information and insights about the topic of research, as well as the way the theoretical frameworks were applied in the thesis. Expert interviews represent the core of the gathered input, supplemented with various Web resources, such as white papers, blog posts, analyses and published interviews. The input is then examined within the protocol feasibility analysis framework, presented in Section 2.1, as well as used later to facilitate the analysis for the second research question (see Section 1.1).

4.1 Interview study

Interviews are a popular tool for gathering data as input to a qualitative research (King, 2004). Robson (2002) categorizes interviews in three main groups based on the level of imposed structure — fully structured, semi-structured and unstructured. Fully structured interviews have a predetermined set of questions with fixed wording and order, while in the semi-structured interviews the set of questions, the order and the exact wording may be adapted during the interview, if the interviewer sees fit. Unstructured interviews, on the other hand, only have an overall topic or area of interest, but discussion can go freely in any direction.

King (2004) refers to semi- and unstructured interviews collectively as qualitative interviews. The goal in such interviews is to view the research topic from the point of view of the interviewees. Moreover, key role in the process has the relationship between interviewer and interviewee. Unlike in quantitative studies, the interviewee is not seen as merely the 'subject' of the study, but rather a 'participant' that can affect the course of the interview.

This thesis aims at presenting a holistic view about WebRTC, which demands investigation of the entire ecosystem that may form around the technology. Therefore, a main goal with conducting our interview study was to achieve breadth by interviewing experts from different types of stakeholders.

As a new technology that is actively being standardized, the people involved in the standardization process have certainly a key role at shaping the technology, affecting its possible adoption and success. Furthermore, they are the authoritative experts on the technology and should have thorough understanding of its workings, technical capabilities and challenges. Consequently, they form the first group of stakeholders, taking part in the interview study.

The second group of stakeholders are the various industry players that are (or would be) implementing, operating, using or otherwise business-wise involved with WebRTC. They should have better grasp of the economic aspects of WebRTC — fitness of the technology to deliver value, the market, business requirements and challenges.

Naturally, these two groups are not disjoint on company level as industry experts work in the standardization organizations. We perceive this as an additional benefit, as it may imply broader viewpoint for the respective interviewees.

A total of eight interviews were conducted within the study. The interviews took part in Helsinki and Espoo between April 11, 2013 and July 09, 2013. Interviews were between 45 and 60 minutes long. Table 4.1 lists the interviewees and their affiliation.

Table 4.1: List of interviewees

Name	**Position**	**Organization**
Ari Keränen	Researcher	Ericsson
	Co-chair MMUSIC WG	IETF
Gonzalo Camarillo	Director of Data/IT Standardization	Ericsson
	Director RAI Area	IETF
Kavan Seggie	CEO	AddLive
Mika Raitola	Head of Patent Management	TeliaSonera
Mikko Kiiskilä	CEO	MeeDoc
Tapio Haantie	Product Manager, Unified Communications	TDC
Tomas Mecklin	Master Researcher	Ericsson
Varun Singh	Researcher	Aalto University / IETF

All interviews followed a high-level structure of four parts:

1. About the Study
2. Interviewee Background

3. Main Questions
4. Feedback

The main topic of the research was briefly introduced to the participants in the beginning of each interview. Next, interviewees were asked to provide some background information about themselves and their experience with real-time communications, which allowed more informed interpretation of the results. Then followed the core questions, classified into five groups. All interviews were semi-structured. Each group of questions contained generic ones that were presented to all interviewees, and possibly some stakeholder-specific questions, asked based on the type of stakeholder each interviewee represented. Finally, we prompted the interviewees to provide feedback and/ or references to other potential interviewees or resources that could be of interest to us.

Here follows an introduction to each of the five groups of core questions. A full list of questions can be found in Appendix A. The questions were formed with the help of the protocol feasibility analysis framework, as explained later in Section 4.3.

WebRTC importance

The questions in this group pursued to determine what the key features and aspects of WebRTC are and, based on those, how significant change could WebRTC bring to the market. Additionally, the demand for in-browser real-time communication was being examined.

Value proposition

With this set of questions, we looked into how WebRTC fits the value proposition of different types of companies, represented by the interviewees, as well as identify possible incentives for adoption of the technology. Furthermore, the questions concerned the attractiveness of certain technical roles (as defined in Section 2.2) from the point of view of the stakeholders and the possible fit from technical and economic perspective.

Alternative technologies

Next, we aimed at gathering opinion about competing and/or alternative technologies and how they compare to WebRTC. Traditional voice services were one particular interest area.

Challenges and solutions

This group of questions was directly targeted at identifying possible challenges and shortcomings of WebRTC. In particular, challenges for adoption on mobile phones were of interest. In addition, this group contained questions aimed at provoking discussion about appropriate solutions to the challenges, discovered earlier. Depending on the situation, these questions were often omitted, as the conversation naturally included solutions.

Evolution

The closing group of questions was pointed at expanding the future outlook and speculating on the upcoming development of WebRTC. It touched on features and use cases that might not be present initially, but may find their way in later on as the technology matures.

4.2 Web resources

As an actively developed technology, WebRTC is a much discussed topic on the Web. Articles are being written by many industry experts and analysts. In this thesis, we use several websites as sources for WebRTC-related articles, that provide additional insight into the topic. Articles there often discuss similar issues to what we address with the interview study and therefore can be a suitable extension to the input, gathered through the interviews.

Firstly, Tsahi Levent-Levi's personal blog, BlogGeek.me, hosts a series of his and guest writers' posts about WebRTC (Levent-Levi, 2013j). Tsahi Levent-Levi is a technologist with years of experience with VoIP and telecommunications (Levent-Levi, 2013a). In his blog, he has also published a number of interviews he has conducted with people from various companies offering products based on WebRTC (Levent-Levi, 2013i).

Second, No Jitter is a Web portal that provides various resources on enterprise IP-telephony, unified communications (UC) and converged networking (No Jitter, 2013). The portal holds multiple WebRTC-related articles from various industry and technology experts. Additionally, the authors regularly engage in a conversation on one another's posts, often with critical opinions, expanding the discussion with complementary insights and arguments. We find this helpful in providing a more balanced view and offering better justification pro and against certain issues around WebRTC.

Third, we note Disruptive Wireless, Dean Bubley's personal blog. Dean Bubley is founder of industry analyst and consulting company Disruptive Analysis (Bubley, 2013a). His blog contains numerous articles on various topics around telecommunications, wireless and mobile industries. He has written several times about WebRTC and also is publishing the *WebRTC Market Status & Forecasts* industry report (Bubley, 2013d).

Finally, there are, naturally, numerous other sources of relevant information and opinions about WebRTC, including published white papers, articles, blog posts, interviews and discussions. While we refrain from listing them explicitly here, we will provide references where necessary.

4.3 Research process

In this section we describe the process followed when conducting the research and further explain our use of the theoretical frameworks, introduced in Chapter 2.

First, we used the framework for techno-economic feasibility analysis of Internet protocols as a guideline to form the set of questions that we asked in the interview study. The framework also helped to prioritize and structure the questions. Then, we carried out the interviews and additionally studied the various Web resources, as discussed in the previous sections. Next, following the logic of the framework, we combined the results to form the analysis of WebRTC's techno-economic feasibility.

Figure 4.1 illustrates how the steps of the protocol feasibility analysis framework relate to the thesis chapters, in which they were mostly carried out, and to the question groups of the interview study. Particularly, the basis for the use case and technical architecture analysis steps was mainly covered in the technical background study provided in Chapter 3. Chapter 5 contains the interview and Web resource study results, representing the core of the feasibility analysis.

Finally, having answered to the question of WebRTC's techno-economic feasibility, we studied the more focused second research question (see Section 1.1) using Value Network Configurations. We utilized all previously gathered insights in order to identify the key technical components and roles that need to be fulfilled in the value network. The main focus of our VNC analysis was to explore how these roles could map to the possible actors in the value network and what market opportunities these different mappings represent.

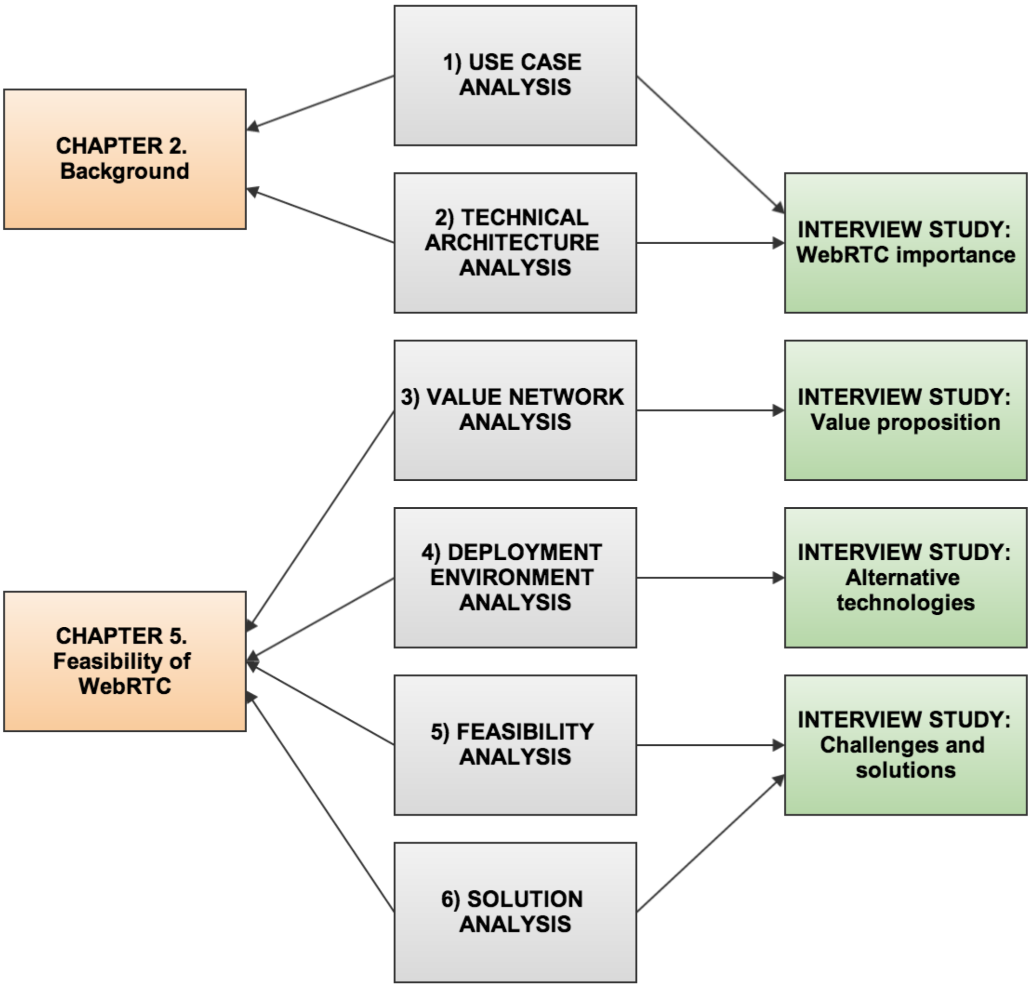


Figure 4.1: High-level mapping from the steps of the protocol feasibility analysis framework to the thesis chapters and interview study.

Chapter 5

Feasibility of WebRTC

In this chapter we present the findings obtained from the interview study and the Web sources, described in the previous chapter.

All interview participants' opinions are their own and do not necessarily express their companies' positions. The full list of the participants in the interview study was given in Table 4.1 in the previous chapter. Furthermore, we intentionally avoid attributing statements to the individual interviewees in the text.

The results are presented in a general way. Rather than looking at each conversation separately and discussing the findings from each stakeholders' individual perspective, we try to draw generalized conclusions that apply more widely.

5.1 The importance of WebRTC

Interviewees generally agreed that the most important aspect of WebRTC is the native Web browser implementation, that does not require installation of any additional third party plug-ins or standalone software. The reason is two-fold. First, being part of the browser ensures much greater reach for the technology, as a browser is present on every Internet-enabled device. Second, this simplifies the process for the end-users, because there are multiple levels of failure related to installing a plug-in or a client, that can prevent the user from accessing the communication service, such as lack of trust in the third party provider, lack of time to perform the installation and lack of access rights.

Another commonly shared view was that WebRTC would be easily accessible for the developers. The standardized JavaScript APIs enable Web developers to create communications applications, hiding the complexity of

VoIP. Thus, the deployment model for Web-based communication applications would be the same as for traditional Web applications, providing the same benefits — easy global deployment and application updates, perhaps facilitated by deployment in the cloud.

Overall, what is described above essentially means, as some interviewees pointed out, that WebRTC commoditizes a valuable communications technology, lowering costs and barriers to entry dramatically. Similar view is also supported by Levent-Levi (2013c,e). Being easily accessible technology for developers, WebRTC could be put to use in numerous use cases, from the traditional large-scale communication services to highly specialized niche communication solutions and applications. The lower costs for developing communications applications with WebRTC could further boost the competitiveness of commercial WebRTC-based systems compared to traditional solutions.

A condition that some interviewees thought was critical to the success of WebRTC is that it would be extremely reliable, in terms of connection establishment and correctly functioning voice and/or video, and easy to implement. This is important for both developers (trust in the technology) and end-users (trust in services or applications, built on top of the technology).

When asked if they consider WebRTC a game changer, interviewees had varied opinion. Some regarded it as a having potentially great impact, and some went further to state that the hugely wide reach, that WebRTC could achieve, and the openness of the technology are features of a definite game changer. However, others were more reserved, describing WebRTC rather as evolutionary, but still an enabler for services that could be disruptive in their own market segment.

The same polarity of opinions regarding WebRTC's disruptiveness is evident also in the Web resources that we follow. For example, Michels (2013a,e) and Michels et al. (2012) argue that WebRTC can be seen at best as an evolutionary technology, because it does not bring any new functionality that is not possible to achieve with existing technology. On the other hand, Kelly (2013) and Levent-Levi (2013h) feel less reserved and go as far as stating that WebRTC could turn the communications industry on its head.

One-to-one browser-to-browser communication on desktop PCs within a single service, as the most basic use case of WebRTC, was widely regarded as the most important. This is especially true for the initial adoption phase of WebRTC, which would be mainly driven by desktop use, according to Bubley (2013d). Furthermore, similarly to the reliability requirement mentioned above, WebRTC must be successful in this use case. Failure to do so could undermine the trust in the technology and prevent more sophisticated use cases, like, for example, massive multimedia streaming applications and multi-party conferencing, from ever reaching wide adoption. Therefore, taking all this into account, we focus the techno-economic feasibility analysis of WebRTC to this particular use case. Scoping the analysis in such a way when using the framework for techno-economic feasibility analysis of Internet protocols is also advised by Levä and Suomi (2013).

Michels et al. (2012) argue that service federation could often not even be required with WebRTC, as simply providing the caller with a URL to the callee's WebRTC-enabled Web page can be enough, effectively keeping all communication within a single service domain. Naturally, this would depend on the service itself — whether non-registered users would be able to make calls — but nevertheless highlights that WebRTC is an enabling technology, rather than a specific communications service.

Communicating with legacy systems, such as PSTN or VoIP end-points was also mentioned in some interviews as an important use case, largely due to the huge existing installed base. Therefore, it has been recognized as such from the very beginning in the standardization process and there has been effort to minimize interoperability problems.

Specifically for use in enterprise communication solutions, some participants expressed a view that WebRTC has the potential to improve interoperability between different systems. This could solve one of the major problems of current systems and eliminate a key reason for concern when purchasing such solutions.

Assessing the market demand for a technology like WebRTC, most interviewees saw some need for ease of access and ease of use from both consumers and enterprise users, that is not fully met by the currently available technologies or products. Particularly, enterprise users recognize benefit from reduced amount of communication application clients. Having the Web browser as a single point of access was seen as an improvement. Another area where some interviewees see strong demand is in-context communication — adding communication capabilities into existing Web services or applications.

5.2 Value proposition

Most interviewees agreed that WebRTC is technology that should be on the radar of many different companies. Communications equipment vendors, service providers and telecoms need to be aware of new communication technologies and shifts in the communication philosophies in order to be able to react and adapt their products or networks to cater for the new services.

5.2.1 Infrastructure provisioning

While the Web browser's implementation of WebRTC hides most of the complexity of VoIP and presents developers with a plain JavaScript API, there is still need for some additional infrastructure to be in place. Such infrastructure may be, for example, STUN and TURN servers or media relays (in case of e.g. multi-party conferencing). In the case of TURN and media relay servers, due to the requirements on bandwidth capacity and latency, the question of who should provide this infrastructure arises. Some interviewees mentioned that from network optimization perspective, this may be a local ISP or Telecom, while others suggested that companies with global infrastructure — global IaaS providers or multinational telecom operators[1] — might be better suited, especially for WebRTC-based services that are offered internationally. A third model was brought up in one conversation, combining the previous two. That is, a global service provider having negotiated access to infrastructure from multiple smaller, local network operators as the underlying layer, similarly to how some Content Delivery Network service providers operate.

5.2.2 WebRTC service provisioning

Another opportunity arising from WebRTC in the B2B market is for providing turnkey solutions for application developers. These solution providers could bundle WebRTC implementations together with the required infrastructure to deliver a full service, offer Software Development Kits (SDKs) for the mobile platforms and possibly additional ready-made components like signaling, discovery or IdP. Some of the interviewees expected that the first commercial WebRTC implementations will come in this form and there are already companies having similar offering[2].

5.2.3 Specialized niche services

Several interviewees recognized WebRTC's strength to deliver unique value particularly for applications requiring in-context communication capabilities, where e-commerce and customer service are often mentioned at the top of the list of examples. For instance, calling one's bank customer service directly via the online banking website could directly deliver to the customer service team information of who the user is (as the user is already authenticated) and what information they might be looking for (based on currently viewed

[1] Amazon Web Services, Akamai, Vodafone, Telefónica, TeliaSonera, TDC, etc.
[2] AddLive.com, TokBox, Voxeo, etc.

page in the website). If a traditional phone call would be made for the same scenario, then the user would typically need to authenticate again and go through several levels of Interactive Voice Response (IVR) choices, before being connected with the correct person. Similar ideas are presented by Levent-Levi (2013b,f,h) and Vitek (2012), for example.

Additionally, some interviewees expressed opinion that this market segment of specialized niche applications and embedded communication services may become a long tail, driving a significant share of the WebRTC's adoption and usage.

5.3 Alternative technologies

There are many technologies that can be considered alternatives to WebRTC. Interviewees frequently mentioned Adobe Flash (with its RTMFP support), Skype, traditional phone calls and SIP based products, among others. If we limit ourselves to the technologies that can provide in-browser communication capabilities, all current solutions require third party plug-ins. Compared to WebRTC's native implementation, these solutions are at a disadvantage in terms of install base and are often more closed systems that do not allow developers to build their own applications on top (as, for instance, with Skype plug-in).

Abandoning the in-browser requirement, traditional phone calls and VoIP systems can, naturally, be considered alternatives. Given the effort to make WebRTC as interoperable as possible with these existing systems, gateways to PSTN and SIP-based systems, for example, can be a way to turn WebRTC into a complementary technology, extending the reach of both the WebRTC and the legacy network. While WebRTC can be seen as direct competitor to mobile phone calls, especially in its most basic peer-to-peer use case, some interviewees pointed out that it can not replace them completely, at least for the time being. Reasons included that traditional mobile calls can be used for emergency calls and are generally more reliable, both in terms of establishing and maintaining a call and trust that a call is actually possible, given one knows the remote party's phone number. However, considering the more advanced use cases of WebRTC, such as video conferencing, opportunities for the network providers exist, as discussed above.

In the Web, discussions about WebRTC alternatives often center around Skype, when consumer-centric services are in the focus. As Michels (2013c,d) points out, Skype can be a threat to WebRTC as much as WebRTC to Skype. A key point is that direct comparison needs to be done cautiously, as WebRTC is a technology, while Skype is an application (or a service).

However, WebRTC, lowering the barriers of entry, would enable a variety of Web services, offering communication capabilities. Some of those may be in direct competition with Skype, while others, as discussed above, may target niche markets and aim to solve some particular pain point (Levent-Levi, 2013f).

Some interviewees raised the question of trust, when comparing WebRTC with the alternatives. The Web browser is typically considered a trusted platform by the users. They feel confident enough to perform secure bank transactions, online payments and other sensitive operations using their Web browser of choice. A third party plug-in or application does not necessarily receive the same level of trust. This results in an advantage for WebRTC, as it is inherently part of the Web browser.

5.4 Challenges and solutions

Generally, the interviewees agreed that the standardization work on WebRTC is going forward well and few open technical issues remain. However, several issues were most often mentioned as the major remaining challenges for WebRTC.

5.4.1 Selection of MTI video codec

Firstly, there is still a struggle within the IETF working group to select a mandatory to implement (MTI) video codec or decide whether one should be selected at all. This is also the most often raised issue in the articles on the Web. The rivaling codecs are H.264 and VP8.

On one hand, H.264 is well established codec, widely used in many video systems (Burman, 2012; Levent-Levi, 2012a). Furthermore, it has wider hardware acceleration support, which is especially important for mobile devices, due to performance and power constraints. However, implementing H.264 would require payment of patent licensing royalties to its patent holders, which include large companies like Apple, Ericsson, Cisco and Microsoft (MPEG LA, 2013). On the other hand, VP8 is freely licensed by Google (WebM, 2013a). Google and Mozilla, as main VP8 proponents, argue that having free and open technology is of utmost importance, as it allows the technology to be used without prior approval (InfoWorld, 2012). Answering to the issue of hardware support for VP8, Google provides IPR for hardware encoders and decoders for VP8 freely and has announced partnership with AMD, ARM and Nvidia (NVIDIA, 2013; The Register, 2010; WebM, 2013b).

Broadcom has also announced support for VP8 in its VideoCore processor family (Broadcom, 2010).

In an attempt to protect VP8 implementations from patent threats, Google reached agreement with MPEG LA — the private entity, managing the H.264 patent pool — that grants Google a license for any patents within the H.264 patent pool that are essential to VP8 (Business Wire, 2013). Further, the agreement allows Google to sublicense these patents to any user of VP8. Nevertheless, other patent claims against VP8 coming from Nokia leave its royalty-free status still controversial (Nokia Corporation, 2013). Michels (2013b) argues, however, that Nokia's claim may not have a significant impact, quoting the difficulty to pursue patent violations.

Similarly, Cisco has made an effort to eliminate some of the concerns, preventing H.264 from being selected as MTI video codec. The company announced that it would provide binary modules for its H.264 implementation that would allow developers to add H.264 support to their applications for free. Cisco would absorb the licensing costs towards MPEG LA, provided that it is the sole party distributing the binary modules. Furthermore, Cisco would make the source code, from which these modules are built, open source. (Cisco, 2013)

While many members of the RTCWEB WG agreed that this is an admirable step forward, this announcement did not seem to be convincing enough, so that consensus could be reached to adopt H.264 as MTI codec in the WG meeting during the IEETF's 88th meeting (see the WG mailing list, e.g. Roach, 2013).

According to Levent-Levi (2013d), the MTI video codec is indeed a challenge for WebRTC for two reasons. First, having too many MTI codecs may result in high complexity and complicate testing between different WebRTC implementations, which could lead to interoperability issues. Second, and probably the more important reason, as discussed also with some of the interviewees, is that having no MTI video codec could lead to incompatible implementations that require transcoding. Transcoding would naturally introduce delay and would break the P2P architecture, as a central node would always be required. That would result in higher bandwidth costs for the service provider, as calls which would otherwise be routed directly between peers, would go through the central infrastructure, and thus have negative effect on adoption. Nevertheless, other interviewees were less concerned, opining that the industry will most likely decide the matter itself, even if the IETF does not mandate an MTI video codec.

Another possible solution to the issue comes from the Moving Picture

Experts Group (MPEG)[3]. There is an ongoing effort in the MPEG to have a royalty-free video coding standard. Internet Video Coding (IVC), Web Video Coding (WebVC) and Video Coding for Browsers (VCB), which is based on VP8, are three exploratory activities within that effort. However, there is no promise on whether or when the effort would succeed. (MPEG, 2013a)

5.4.2 Interoperability

The second challenge is the interoperability between WebRTC implementations. Some interviewees agreed that if WebRTC would be to succeed and reach the promised ubiquity, it needs to work seamlessly on all the major browsers. This would help maximize the positive network effects, arising from the wider adoption of the technology. Similar concerns are raised by McDonald (2013) and Michels (2013a).

Particularly, the issue is whether Microsoft's proposed alternative specification (CU-RTC-Web) would end up compatible with Google's and Mozilla's WebRTC implementation and whether or not Apple will choose to implement (some form of) WebRTC at all. Microsoft's CU-RTC-Web has lower level API than WebRTC, pushing more complexity to the developers, but allowing more flexibility. According to an article by TokBox (2013b), and also supported by several of the interviewees, a compatibility layer would not be difficult to build between the two specifications and third party vendors (like AddLive or TokBox) would respond to the need. However, it would be a greater challenge, if the media capabilities of the different implementations are incompatible. This relates back to the first issue of MTI video codecs, as Microsoft is in favor of H.264, while Google and Mozilla support VP8.

Regarding Apple's stance on WebRTC, some interviewees felt less concerned. They often explained that software development kits (SDKs) developed by Google or third party vendors could solve the problem of WebRTC implementation on iOS, which according to them is what really matters, when it comes to the Apple ecosystem.

Most interviewees agreed that the issue of interoperability with existing systems should not be ignored. While WebRTC is a new technology, the current install base of legacy systems is big enough to justify some effort to make WebRTC at least to some extent compatible. As discussed above, gateways would most probably be a required part of the solution. As some

[3]Note that the MPEG is a working group formed by the International Standardization Organization (ISO) and the International Electrotechnical Commission (IEC) (MPEG, 2013b). MPEG LA, which is the private firm managing the H.264 patent pool, is not in any way affiliated with the MPEG.

interviewees pointed out, the legacy market is big enough so that there will be players who aim to bridge the two.

5.4.3 WebRTC on mobile devices

Another challenge, perhaps more so in the long term, is the use of WebRTC on mobile devices. According to Bubley (2013c,d), initial effort and adoption for WebRTC would be in the PC (desktop) environment, but further growth would come from mobile devices. Several interviewees supported that view as well. They also noted that care has been taken in the IETF RTCWEB working group to avoid introducing specifications that would end up hurting mobile use.

However, they agreed that there are open issues that would need to be addressed, perhaps in later revisions of WebRTC specification. These include power consumption and processing resources needed for real-time video conversations, congestion control and video quality guarantees. Some interviewees were confident that such challenges would be resolved when attention is turned to them and pointed out that applications supporting video conversations (Skype and Viber, for instance) are already in use on mobile devices. They concluded that there are no fundamental differences preventing WebRTC to be used in that context as well. As mentioned above, improvements in hardware support for video encoding and decoding would definitely help for having high quality WebRTC experience on mobile devices.

5.4.4 Security and privacy

Continuing the discussion from Section 5.3, some interviewees viewed trust as a minor challenge as well. While building support for WebRTC inside the Web browser, it itself becomes a more complicated piece of software, which opens up more opportunity for mistakes and errors. This could eventually lead to part of the trust being lost, especially if security-related issues were discovered. At the same time, the interviewees felt that the major browser vendors already have the right experience maintaining high quality software products and are, therefore, well suited to execute WebRTC implementation.

However, none of the interviewees raised concerns about security or privacy issues on the protocol and API levels of WebRTC. Presumably, this is because security and privacy have been well in focus in both IETF and W3C (Bergkvist et al., 2013; Jennings et al., 2013; Rescorla, 2013a,b).

5.4.5 Business

On the business side, some interviewees raised concerns that WebRTC, being just an enabling technology, does not possess any inherent business model. Therefore, companies that do not thoroughly understand what the promise of WebRTC is, what the technology can and can not do, or how to utilize it, could suffer. At the same time, the interviewees also view this as a major opportunity, opening up innovation possibilities for those willing to experiment.

Overall, the more technical oriented background of most of the interviewees could have been a factor explaining why they discussed fewer business-oriented challenges than technical ones. We regarded that as further motivation to explore the business side in more detail, studying the second research question.

5.5 Evolution

The first complete version of the WebRTC specifications from IETF and W3C would be an important milestone in the development of the technology. Interviewees were enthusiastic that that would not conclude the work, though. New features, use cases and improvements would surely follow. Improved multiparty video conferencing capabilities and features like screen/ desktop sharing were often mentioned as natural evolutions of WebRTC. More focused attention to mobile platforms is a logical step, as well, given that mobile devices are becoming more and more the preferred choice for Web browsing and application usage for many users.

Some interviewees expected to see WebRTC utilized in massive streaming and broadcasting applications, especially where a degree of interactivity is required. According to them, current streaming solutions work well as a one-way channel, but typically operate with a delay (coming, for instance, from buffering and transcoding) that is prohibitive to real-time interaction. WebRTC could provide interesting alternative with its inherent real-time nature.

Many interviewees felt excited to see how some key aspects of WebRTC will develop further. For example, how the data channel might reach wider application (gaming, P2P CDNs, etc.), where the user directories and presence information would be, how would identity providers develop and whether a new type of identity would emerge (for example, not only identifying a user, but also identifying user's individual devices). Furthermore, with the development of image recognition algorithms, for instance, what kinds of new applications would be developed, enhancing or re-inventing real-time communications or building the future Web.

Overall, interviewees were confident that, whether or not WebRTC would reach the promised ubiquitous use, it is a important step in the evolution of the Web and will have its place in the technological landscape in the coming years.

5.6 Summary

As mentioned in Section 5.1, the scope of our feasibility analysis is the basic use case of WebRTC, described previously in Sections 3.2 and 5.1. Therefore, challenges related to, for example, use of WebRTC on mobile devices are not relevant within this scope.

Deployment of WebRTC is relatively simple to achieve in technical sense. As discussed earlier in Section 3.2.1, the only requirements are support from the end-points (Web browsers) and the availability of required infrastructure (STUN and TURN servers). Furthermore, given the selected use case, service providers utilizing WebRTC are independent and, for example, can implement signaling without regard of interoperability with third parties.

WebRTC's most notable advantage over alternative technologies is its native in-browser implementation. This allows for both ability to seamlessly integrate the technology into existing or new Web services, as well as simplify installation for end-users.

The two challenges relevant to the basic use case are the selection of MTI video codec and inter-browser compatibility with regard to Microsoft's CU-RTC-Web specification and Apple's unknown position on whether they would implement WebRTC or not (see Section 5.4). Both of these issues potentially have solutions.

Overall, based on the presented results, we believe that the answer to our first research question — *"Is WebRTC techno-economically feasible?"* — is positive. We did not identify and major roadblocks that would prevent the adoption of WebRTC. Consequently, the business opportunities arising from the adoption of WebRTC are interesting subject for further investigation. We study these in the following chapter.

Chapter 6

Value Network Configurations for WebRTC

In this chapter we focus on the second research question, introduced in Section 1.1: *"What market opportunities will likely occur as a result of WebRTC adoption for the relevant stakeholders in the ecosystem?"*.

We begin our Value Network Configurations analysis by introducing the roles that we have identified when examining the WebRTC technical architecture, presented in Section 3.2, with the help of the discussions and findings from our interview study.

Next, we introduce four main Value Network Configurations related to WebRTC. These VNCs were selected based on their importance and likelihood of occurrence, as well as to highlight some opportunities and possibilities for the various stakeholders in the WebRTC ecosystem.

In turn, we examine each of the VNCs in more details, emphasizing on the mapping between roles and actors. Further, we examine who these actors may be, since in practice different market players may participate as a given actor in the value network.

6.1 Role analysis

When constructing the VNCs, we no longer look at WebRTC as a technology in general, but rather focus on some concrete service that includes real-time audio, video or data communication components, based on WebRTC. In other words, we look at a particular application of WebRTC to address some technical or business need. Below, we refer to this concrete service as "core service". As WebRTC could be utilized in a myriad of use cases, here we are not interested in what this core service might actually be, but rather at how the building blocks (the roles and their technical components in the VNC) could be arranged in the market and what conclusions we may draw therefrom.

Without any loss of generality, we can assume that the core service is delivered as a Web-based service, accessed by the users via a Web browser. We believe that this assumption does not alter or limit the roles that need to be fulfilled in order to deliver the core service in any significant way, at least in the scope of our analysis. The technical components in each role may need to change, for example, if the core service is delivered as an application for mobile phones, but the roles and their allocation to actors would remain intact.

Consequently, the five main roles that need to be fulfilled by the different actors in the value network are listed below.

- Application provisioning

  This role is mainly concerned with the applications providing the core service. These applications cover the business logic of the service, but exclude the components, strictly related to facilitating WebRTC communication. The applications would typically have both server- and client-side parts (for instance, the back-end application and the JavaScript application that runs in the user's Web browser).

- WebRTC component provisioning

  This role covers provisioning of the WebRTC components, that may be utilized by both the server- and client-side applications in the core service. These include, for example, signaling implementation, libraries, SDKs and APIs. In the most simple use case the WebRTC APIs, implemented in a Web browser, are used directly (i.e., without using any additional components), and thus the role requires only signaling to be implemented in the core service provider's application. However, the role's importance increases significantly once third party components are used in order to enhance WebRTC functionality, simplify use or improve integration and compatibility, for instance.

- WebRTC infrastructure provisioning

  This role covers maintaining and operating the infrastructure required by WebRTC, such as STUN/TURN servers and media relay servers. Due to performance requirements, pursuing network latency and bandwidth optimization may result in some actors being better suited to assume this role than others.

- User account provisioning

  This role encompasses maintaining the primary user database and authentication service, bundled with identity provisioning service, used for WebRTC communication (see Section 3.2.2). The role therefore includes establishing the main identifier of each user, which can be, for example, a username, e-mail address, full name or phone number.

- WebRTC usage

  This role denotes the consumption of the core service, including its WebRTC components.

Apart from the main roles, which are included in all VNCs, we identify also three additional *optional* roles, relating to providing and connecting to a legacy system, such as PSTN, PLMN or VoIP-based solutions. We include these roles, because we consider the VNCs incorporating integration to legacy services one of the interesting cases.

- Legacy service provisioning

  This is a broadly defined role, accounting for all infrastructure and activities required for providing some existing (legacy) communication system.

- Gateway provisioning

  This role covers the infrastructure and activities required to bridge WebRTC-based communication services with legacy systems. The solution may include gateways for signaling, media, authentication, etc.

- Legacy service usage

  This role denotes the consumption of the legacy service.

6.2 Basic VNC

The first VNC that we consider is the one where all roles (except for the user-side one) are fulfilled by a single actor — the provider of the core service.

Let us take an imaginary housing service as an example. On the housing service Web site, housing owners offer accommodation for rent and people seeking accommodation can get in contact with the offering party. The site implements voice and video calling through WebRTC, relying on the WebRTC availability in the Web browsers. The housing provider operates on its own all the required infrastructure.

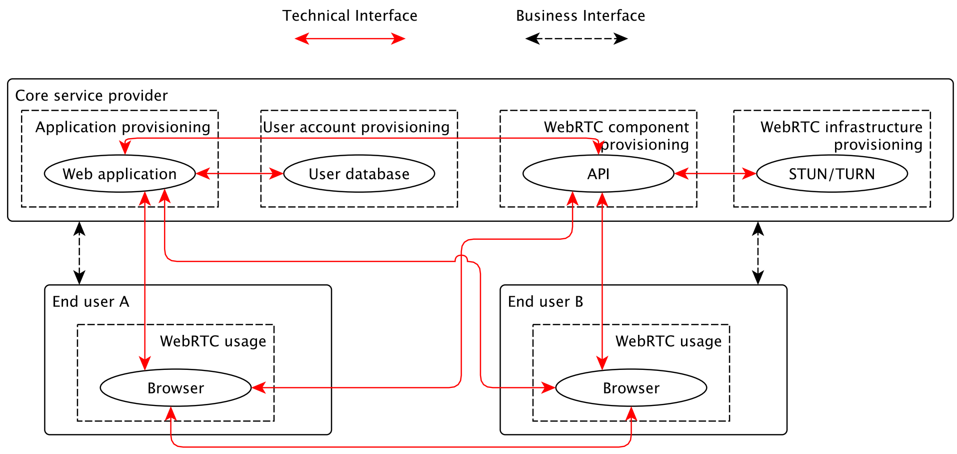


Figure 6.1: Basic VNC with single service provider.

Figure 6.1 shows this VNC, corresponding to the given example. In order to avoid reducing readability by cluttering the figure with too many connecting arrows, the some of the media path and identity provisioning service interfaces, which were described in more detail earlier in Section 3.2.2 and Figure 3.5, are omitted here and in the following sections.

Notably, the business interface between the service's users and the service provider does not necessarily involve money exchange. However, even if the service is offered for free, there is typically still exchange of intangible value in both directions — the users gain value by using the service and the service provider may find value in simply having the users' attention, profiling the users or gaining understanding of their behavior.

6.2.1 Considerations

The basic VNC is most probably going to be seen quite commonly in the market. An example of this VNC could be service providers who are able to work directly with the WebRTC APIs in browsers. Such providers could be building a dedicated communications service or enhancing their new or existing core service with real-time communication capabilities.

This VNC can be considered basic in terms of technology as well. Here, the core service depends on the general availability of WebRTC implementations in the Web browser(s). The core service provider has the standard WebRTC functionality and APIs at their disposal and must implement on their own the additional necessary components (e.g signaling) and maintain server infrastructure (STUN/TURN, etc). Therefore, we can conclude that core service providers operating in this VNC would need to have highest level of competence and skills in regard to WebRTC, compared to the alternative VNCs described later in this chapter. At the same time, however, this ensures greater level of independence by freeing the service provider from reliance on third party providers, at least as far as WebRTC is concerned. Whether this would also have direct impact on cost is, nevertheless, questionable because of the possibly higher personnel costs.

The importance of the WebRTC infrastructure provisioning role here is directly dependent on the service provider's target audience. On one hand, there are the service providers, who would target relatively small, geographically contained markets, such as local bank websites or university student organization portals. The geographical locality of the user base would mean that there are much lower requirements on the infrastructure side of the service provider.

On the other hand, there are those service providers, who would target a larger geographical area — national, international or even global. This directly results in need to provision and maintain infrastructure close enough to the users. Service providers who are already operating infrastructure in diverse locations in order to support their core service have an advantage here. In their case, the addition of the WebRTC server-side components would most probably not introduce much greater complexity or operational costs.

On the other hand, generic Infrastructure as a Service (IaaS) clouds, such as Amazon Web Services or Microsoft Azure, can greatly reduce the burden of running own global infrastructure. We consider such services the obvious choice for startups, seeking fast time-to-market and to be able to expand quickly, should their new WebRTC-enabled service proves to be successful.

6.2.2 Actors

Newly established communication-oriented core service providers might find it challenging to acquire the multi-million user base that some existing communication services boast[1]. However, the large social networking services, such as Google+, Facebook, LinkedIn and VK, could become an interesting case of large-scale WebRTC adopters. Google, as one of the main supporters of WebRTC, seems the obvious early adopter, especially after migrating its Hangouts service — the RTC components of Google+ — to use VP8 video codec (Gigaom, 2013). Each of these services has a substantial user base and if

[1]For example, Skype: 663 million in the end of 2010 (Telecompaper, 2011); Viber: 200 million (Viber, 2013).

some decide to introduce voice and/or video communication[2] features, based on WebRTC, they could become the first multi-million user deployments of the new technology.

Coming back to the issue of trust in WebRTC as an emerging technology, such large scale "installations" would serve as excellent proof of concept, signaling others that WebRTC is indeed deployable and reliable for production use. This would be important especially for more conservative players, such as large enterprises, who would generally refrain from adopting a new and unproven technology.

As discussed briefly in the previous chapter, specialized niche service providers are very likely to show interest in WebRTC. Most likely, the majority of these would not have as a top priority to attract as many customers as the Internet giants, discussed in the previous paragraph, but rather focus on solving a particular issue, perhaps charging premium from their customers.

6.3 WebRTC PaaS VNC

In the second VNC, the two WebRTC-related roles are assigned to a separate actor — a WebRTC provider. This change illustrates an opportunity in the market for new entrants[3], who would aim to relieve the core service provider(s) from the need to deal with the infrastructure complexity and further enhance the functionality they could get from WebRTC. Such providers would become essentially Platform as a Service (PaaS) providers in the WebRTC ecosystem.

As illustrated in Figure 6.2, a business interface is formed between the core service provider and the WebRTC provider. The business interfaces towards the end-users, as well as all technical interfaces, remain intact. Therefore, the WebRTC vendor here is a B2B market player.

Continuing with the example of the housing service from the previous section, here the housing service provider chooses not to invest time and effort into researching how to build a WebRTC application. Instead, it purchases WebRTC as a service from a WebRTC provider. Thereby, it obtains simpler (JavaScript) APIs which are ready to be plugged in into the rest of the housing service's application. All infrastructure required for WebRTC is handled by the WebRTC provider.

---

[2]Granted, Facebook's video chat service is already powered by Skype (Facebook, 2011).

[3]There are already companies in that space, like TokBox, AddLive, Priologic, XirSys, etc.

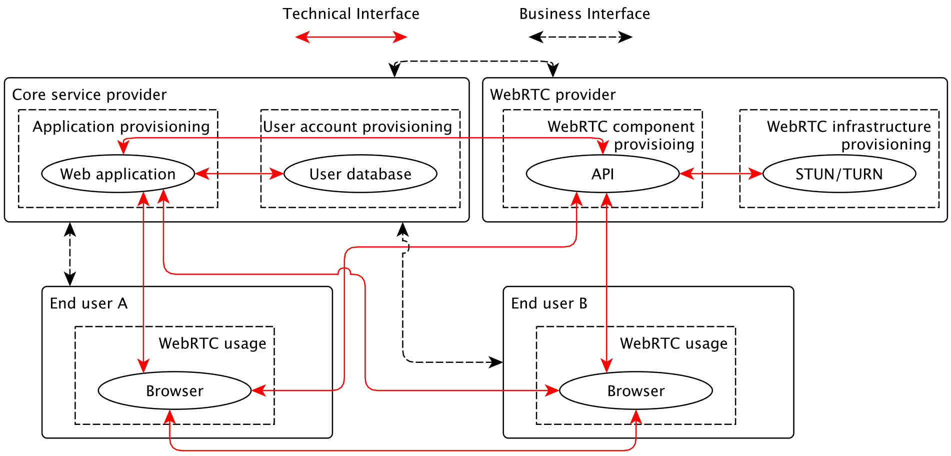


Figure 6.2: VNC with third party WebRTC provider.

6.3.1 Considerations

The role of the WebRTC provider could be beneficial to the WebRTC ecosystem in several ways. Firstly, while the technology is still in its early stages of maturity, such providers could shield their customers from any underlying changes in the WebRTC API or build features that standardization organizations have not yet agreed upon or put into the standards. Companies in this space are already offering, for example, screen sharing (AddLive, 2013) and archiving (TokBox, 2013a) as features of their WebRTC-based communication platforms. The WebRTC provider could also extend their offering with value-added features, such as plug-in support for non WebRTC-enabled browsers, SDKs for native development of desktop or mobile phone applications, analytics and technical support.

Secondly, the WebRTC provider's production-ready infrastructure and software could help the adoption of WebRTC for core service providers, who would otherwise be hesitant to adopt a new and immature technology, or simply would not have the technical skills required to take it into use by themselves.

Thirdly, especially in cases where international or global scale is required in the core service, a well-equipped WebRTC provider with world-wide infrastructure could be a critical partner in the value network.

6.3.2 Actors

While the role of a WebRTC provider is a new opportunity, the provider does not need to be a newly established company. Existing companies with expertise in VoIP or telecommunications could have relevant knowledge and experience to be successful entrants in the market. In particular, telecoms and ISPs might see opportunity to diversify or find new revenue sources. Their position at the edge of the network, close to the users, makes them feasible players to host the latency and bandwidth challenging components of the WebRTC infrastructure — the TURN servers and the media relay servers, used in case of multiparty conferencing. In many cases the telecoms are already operating internationally and have extensive infrastructure in place across wide geographical area, which, as discussed earlier, can be leveraged to support the scale that some core service providers may require. An example that may be a confirmation of this is the acquisition of TokBox by Telefónica in 2012 (TokBox, 2012).

An interesting byproduct of having strong WebRTC PaaS providers might be that they could have the market power to influence the direction of WebRTC development and evolution, together with other major stakeholders, like the browser vendors. This could indeed be the case, when standardization organizations deliberately or as a result of lack of consensus do not explicitly mandate concrete requirements on certain aspects of the technology[4].

This PaaS VNC is in a way (intentionally) simplified, because it does not show in detail the relationships that the WebRTC provider may have or need. Looking one abstraction layer below the WebRTC provider, it is easily possible that their (global) infrastructure is in turn supported by an IaaS provider's cloud. Furthermore, it may actually be supported by multiple cloud services and/or hosted infrastructure by other partners (such as telecoms or ISPs). The reason is that relying on single cloud provider (however big) might not always ensure presence close enough to the customer, which may be required for providing good quality of service for real-time video communication.

In the previous subsection we already mentioned why certain core service providers may choose to use a WebRTC platform provider, instead of maintaining the required WebRTC components on their own. Overall, we consider it likely that for such providers, voice and video communication is just a feature of their application. It may perhaps be even a central one, but still only an enhancement to their core solution.

[4] As discussed in the previous chapter, some interviewees suggested that the issue of MTI video codec might be left to the industry to settle.

6.4 Extended PaaS VNC

With the Extended PaaS VNC, we take a step further and examine a VNC where the user account provisioning role is assigned to the WebRTC provider, instead of the core service provider.

The Extended PaaS VNC is presented in Figure 6.3. Compared to the previous VNCs, we accent here that the PaaS provider could not only support multiple core service providers, but could also facilitate interoperability (or service federation) between them, illustrated by technical interface between end-users of two different core service providers. To support this, a business relationship would exist between the end users and the WebRTC provider itself. In other words, the users have registered accounts with the WebRTC provider in addition to the accounts they may have with the core service providers. Those "secondary" service accounts, however, are only used to hold the service-specific information about the users, and do not provide their primary identity. The primary identity is established by their account with the WebRTC provider.

Let us return to the familiar example of the housing service. This instance is similar to the one from the previous section, the WebRTC PaaS VNC. However, this time, in order to log in into the housing service, customers use their account with another service provider (perhaps a social network), which in this case happens to be the WebRTC service provider as well.

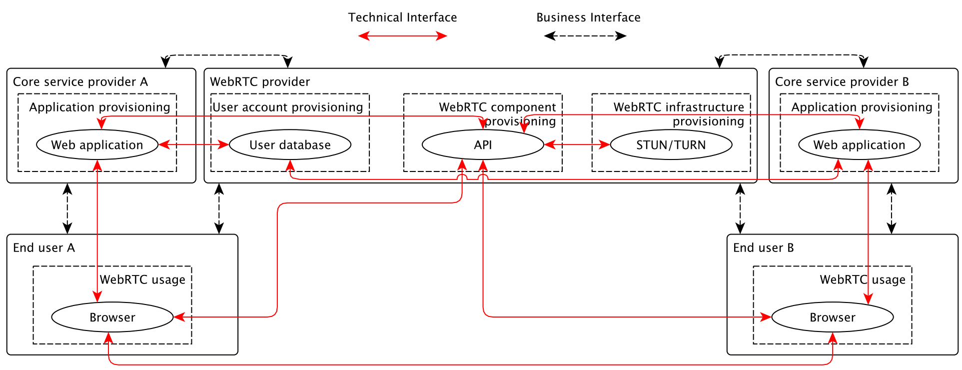


Figure 6.3: Extended VNC with third party WebRTC PaaS provider. The two core service providers have identical (symmetrical) interfaces with the WebRTC provider and the end users.

6.4.1 Considerations

We consider this VNC interesting to examine mostly because of our belief that since here the PaaS provider controls a user accounts database (and thus acts as Identity Provider for the WebRTC communication, see Section 3.2.2), its role in the ecosystem could naturally extend to allow interoperability between the different services, using the WebRTC PaaS. This interoperability comes in the form that users of service A could communicate transparently with users of service B, without the strict need to build service federation and bridge signaling between the services, since from WebRTC perspective, they both use the same complete WebRTC infrastructure.

We would like to point out that controlling the user account provisioning role is not a strict requirement in order for the PaaS to provide interoperability. In the previous VNC — WebRTC PaaS VNC — the WebRTC provider could still facilitate interoperability, but then greater technical effort would be required from its users (the core service providers).

6.4.2 Actors

Another reason to consider this VNC is with regard on who the central actor in it, the WebRTC provider, could be. Intuitively, continuing from the VNC from Section 6.3, the WebRTC provider could be a company doing just that — having its core business in enhancing WebRTC features and providing a communications platform for others to utilize. In that sense, the Extended PaaS VNC is conceivable a step further.

Such WebRTC provider could be offering both options at the same time for its customer to choose from. Some core service providers might prefer the deeper integration with the WebRTC provider's platform, while other may choose to utilize a minimum set of features, in order to avoid requiring users to maintain yet another online account[5]. Indeed, this extra relationship that the end users need to be involved in could be a prohibitive factor, if the WebRTC provider is a new entrant. The provider would need to not only sell their service to the core service providers, but also convince end users to establish the relationship with itself, directly or perhaps automatically via the core service provider(s).

Therefore, a more feasible alternative could be presented when a company which already has relationship with the end users enters the WebRTC ecosystem as a PaaS Provider. Next, we present a few examples.

[5]The one with the WebRTC provider.

Skype as a WebRTC PaaS Provider

Skype currently offers limited integration options for core service providers with its Skype URIs (or “Skype Buttons”) (Skype, 2013). However, as noted in Section 5.3, that solution has the disadvantage of requiring browser plug-in and furthermore does not provide as deep and context-aware integration as a WebRTC-based solution could.

Microsoft has been involved in WebRTC standardization, and as owner of Skype it might not be unimaginable to foresee a future Skype version compatible with WebRTC. If that would happen, Skype could be offering third party services opportunity for much better integration and open up the Skype infrastructure as a platform for others to build on top. Therefore, by extension, Microsoft would become a WebRTC PaaS Provider, having the advantage of millions of existing users (Telecompaper, 2011) familiar with the brand. Furthermore, they could also inherently bring Skype’s integration with Facebook to WebRTC (Facebook, 2011).

Whether Microsoft would head in that direction with Skype is, naturally, open question. Even if Skype continues to be the relatively closed, proprietary service that it is now, other actors might have similar opportunity in front.

Google as a WebRTC PaaS Provider

As discussed throughout this thesis, Google is betting high on WebRTC. After transitioning its Hangouts service to VP8, the next obvious step would be to migrate it to use WebRTC (Gigaom, 2013). Similarly to the Skype example above, Google could open up Hangouts to integration with third party services. That would not be unprecedented, given the integration the company already allows for instant messaging, which is used by, for example, Microsoft’s Outlook.com (Microsoft, 2013).

According to Levent-Levi (2012b), Google’s stated commitment to promoting an open Web (Google, 2009, 2013a) supports the company’s strategic objective of keeping people’s attention in the Web, where it can serve them advertisements. Even though supporting WebRTC alone is already aligned with that objective, having Hangouts as a platform, open for other service providers to integrate to, does not contradict with it either. Google has been in the process of integrating more and more of its services closely with Google+ (of which Hangouts is part) ever since its launch (CNN, 2013). Consequently, driving additional usage to its social network through deeper integration not only with their own, but with third party services as well, may very well be an attractive opportunity.

Additionally, Google has already been allowing third party services to authenticate users via Google's IdP (Ko et al., 2010) for a while. Therefore, we could assume that end users are more or less familiar with the concept of using their Google account to access different services, reducing the risk, discussed earlier, of requiring a separate WebRTC provider account being a barrier for adoption.

Telecoms as WebRTC PaaS Providers

Over-the-top (OTT) service providers are offering voice and video communication capabilities or messaging through the Internet directly to mobile phone users, largely bypassing control from the telecoms. Examples include services like Skype, Viber and WhatsApp. Accordingly, Bertin et al. (2011) argue that the importance of telephony may decline in the future, under the threats of these OTT providers. They also note that key assets for the telcos has been maintaining unified numbering, culture of interoperability and their ability to provide unified communications over diverse devices and services. The authors further suggest that a possible strategy for telcos could be to partner with OTT providers or compete with them, offering unified communications.

WebRTC, being an open standard promoting interoperability, correlates well with these traits of the telecom operators. We already noted in Section 6.3.2 that telcos could be viable player, assuming the role of WebRTC PaaS providers. Here, we emphasize that their key asset of controlling the numbering — uniquely tying a universally recognizable identifier to individual person's mobile devices — could be leveraged to become the main identity that users would have across multitude of services. Moreover, OTT players are already taking advantage of the ubiquitousness of phone numbers. For example, services like WhatsApp or Viber use phone numbers to identify users. Considering this key asset, combined with the geographical reach of telecoms and their experience in running diverse infrastructure, becoming WebRTC PaaS providers is perhaps an attractive opportunity.

6.5 Gateway VNC

In this section, we extend the VNC with the roles related to legacy system integration, presented in the beginning of the chapter. The Gateway VNC is based on the WebRTC PaaS VNC from Section 6.3, with the addition of some legacy system (e.g. PSTN). The WebRTC PaaS provider takes on the role of integrator (gateway provisioning role) that ensures interoperability

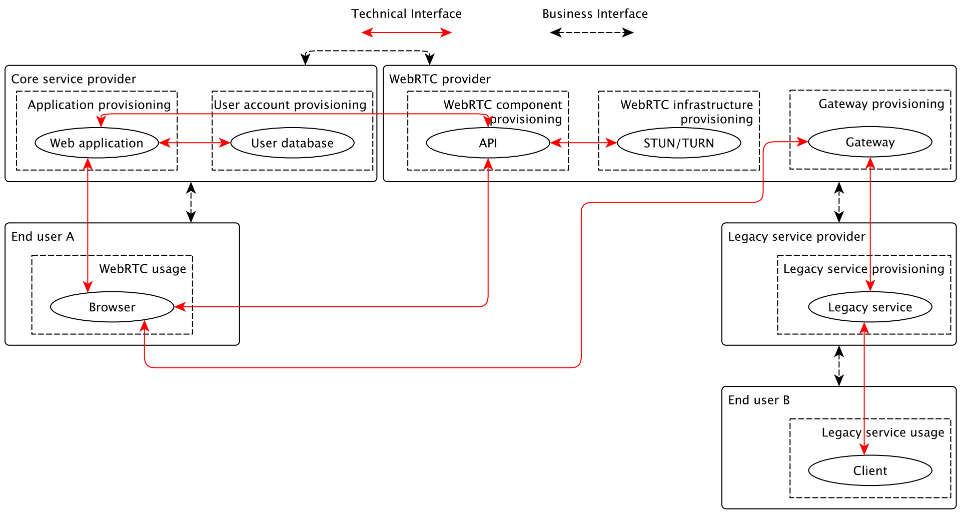


Figure 6.4: VNC with WebRTC PaaS Provider integrating to a legacy service.

between the WebRTC-based and the legacy communication services.

Figure 6.4 illustrates this configuration. Here, the second WebRTC-enabled end-point that User A's browser is communicating with is actually the gateway, rather than the User B's device. User B's client device, on the other hand is purely communicating with the legacy system infrastructure, which in turn allows communication with the gateway.

In the housing service example, here the WebRTC provider integrates with a local mobile network operator. This allows users of the housing service to call owners directly on their mobile phones when they are not online on the service's Web site (or vice versa).

6.5.1 Considerations

During our interview study, we often heard opinions that interoperability with existing systems should be considered important when standardizing WebRTC, at least to the extent of avoiding intentionally introducing things that could break compatibility. Still, integration with legacy systems may require at least signaling to pass through a gateway, because, by design, WebRTC does not mandate strict requirements on how signaling is done. Sandgren et al. (2012), for example, illustrate a WebRTC-to-IMS gateway, allowing users to call mobile phones from a web page.

Furthermore, legacy systems can be attractive resource to tap into. For example, extending connectivity of a WebRTC service to PSTN's or PLMN's huge existing user base could be a significant incentive for adoption, by making the service more attractive to join. Therefore, we consider it obvious that many players in the ecosystem may aim at building gateway products or services[6], filling in for this market need. Levent-Levi (2013g) and Zmora (2013) support this view as well.

Naturally, integration with legacy systems could be achieved in many different ways and VNCs. Gateways could be sold by vendors to core service providers as products or could be offered as hosted service, for example. In this particular VNC, we consider the case where the WebRTC PaaS provider extends its offering, by providing gateway solution in combination with the "standard" WebRTC infrastructure and components. We believe this is a good example of the value-added services that PaaS providers may wish to implement in order to differentiate.

Depending on the concrete legacy system in question, the PaaS provider may need to reach contractual agreements with the legacy operators (note the business interface between the PaaS Provider and the Legacy Service Provider in Figure 6.4). For example, in order to get access to PLMN, they may need to establish partnership with mobile operators. The implications here are twofold. First, it is important for the PaaS provider to pursue such partnerships, which then could be leveraged to provide better and more functional service. Second, this opens up opportunity for legacy system providers to sell wholesale access to their service.

Depending on the relative size of the legacy network, compared to the WebRTC-enabled one, positive network effects may provide incentive for interconnection in either direction.

6.5.2 Actors

Building on the discussion about telecoms' possible role in the WebRTC ecosystem from the previous sections, we would like to point out that telecom operators could easily acquire the technical capability to offer the interfaces, required in the Gateway VNC. This is because telcos generally rely on vendors for the technology they need and, as mentioned earlier, there are already offerings that support integration between the WebRTC and the operators sides.

However, as a more interesting case, let us consider the Extended PaaS VNC (Figure 6.3) with a telecom, fulfilling the role of WebRTC provider and including the integration with PLMN. Figure 6.5 presents this alternative VNC. With this configuration, the telecom becomes an all-in-one WebRTC

[6] Many already have offerings in this space, including gateways to IMS and RCS (GENBAND, 2013; Huawei, 2013; Requestec, 2013).

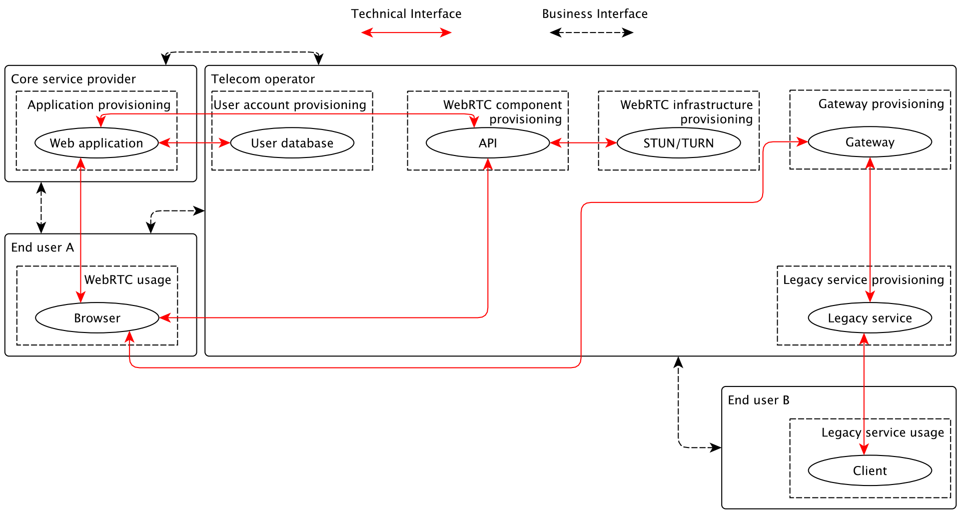


Figure 6.5: VNC with telecom operator as all-in-one WebRTC PaaS Provider.

PaaS provider. Furthermore, apart from just offering the WebRTC platform as a service only to third party service providers, it could also engage in new service creation and build own services to offer to its direct customers on top of the same platform.

Indeed, Bubley (2013b) advises that telcos should not focus exclusively on IMS integration in their WebRTC plans. According to him, IMS would not often be the basis for new services with which the telcos would pursue the fresh revenue streams they seek to obtain. Rather, IMS integration is only part of the picture (as also our VNC suggests), perhaps easing the transition from the declining importance of telephony towards the next generation messaging or communications platforms.

6.6 Summary

Table 6.1 lists the four main VNCs we presented, as well as the main market opportunities and considerations. The considerations are annotated as follows — a + or – sign precedes each item, illustrating whether it is considered positive or negative from the point of view of the respective stakeholder(s).

Rather than focusing on possible ways of using WebRTC and the opportunity for innovation in the WebRTC-based services themselves, the Value Network Configurations discussed in this chapter take on a different task. Using them, we illustrate several market opportunities arising around the adoption and deployment of WebRTC, which in our opinion exist regardless of what each specific end service might be.

The Basic VNC, as the most simple case, allows core service providers to adopt WebRTC as an enabling technology and use it in whatever way they see fit. This freedom comes at the cost of having to implement and maintain all necessary components for themselves. Consequently, with the WebRTC PaaS VNC we demonstrate the market opportunity for WebRTC providers to take care of these components on behalf of their customers — the core service providers. Naturally, this introduces a level of dependence, which however, may be mitigated in case there would be enough competing WebRTC providers on the market.

With the Extended PaaS VNC, the focus moves away from the core service providers. Here, we showed opportunities for several existing players — Skype, Google (or other similar types of companies) and telecoms. Having already built strong relationship with millions of users, these companies could easily take on a WebRTC provider role. Additionally, leveraging the central user identity, the WebRTC provider here can offer inherent interoperability or "federation" between the different core service providers using its services.

Finally, the Gateway VNC accentuates the possibility to utilize interconnection with legacy communication systems in order to bootstrap a new service. This can be used by telecoms as a staging point for a graceful migration towards next generation communication services.

Table 6.1: VNC comparison

VNC	**Market Opportunities**	**Considerations**
Basic VNC: Core service provider assumes all WebRTC related roles	Social networks as first large scale WebRTC adopters; Enabling technology for smaller scale specialized niche service providers;	+ Independence from third party providers + Opportunity for differentiation – Higher level of technical skill required – Take care of infrastructure complexity
WebRTC PaaS VNC: Providing WebRTC components and infrastructure as a service	Providing WebRTC platform as a service is a new role in the market; Allow easy adoption for smaller core service providers;	+ Easier access to WebRTC + Commercial implementations and value-added services – Dependency on WebRTC provider
Extended PaaS VNC: All-in-one WebRTC offering and identity provisioning	WebRTC provider could become central player; Players like Microsoft (Skype), Google or telecoms in position to assume the role;	+ Opportunity for WebRTC providers + Inherent “federation” + “All-in-one” service – Lock-in for core service providers – Additional user accounts may increase complexity
Gateway VNC: Connecting WebRTC services with legacy communication systems	Opportunity for telecoms to offer new services and seek new revenue sources; VoIP providers could allow integration and sell wholesale access;	+ Access to existing communications network + Potential stronger positive network effects boost adoption + Graceful migration towards new communication services for telcos – Carry through legacy system complexity

Chapter 7

Conclusion

In this chapter we summarize our key findings and provide discussion on the limitations of the study. Additionally, we propose some areas that might be worth investigating in future research.

7.1 Key findings

One of the key findings is that WebRTC is a techno-economically feasible technology, at least in its basic use case of one-to-one browser-to-browser communication on desktop computers. It arrives as a natural step in the evolution of the Web, which is inevitably linked with the evolution of the Internet itself.

WebRTC possesses a simple deployment path, where availability of the technology only relies on implementation in certain key software applications, such as Web browsers and SDKs. Its key advantage over alternative technologies is the native implementation in Web browsers, which allows it to reach the ubiquitous deployment of a Web browser itself without requiring installation of plug-ins or third party applications.

Still, WebRTC faces a few challenges. First and foremost, the struggle to select mandatory to implement video codec for the specification opens up a possibility that different WebRTC implementations might not be compatible with one another on the media level. Second, the ubiquitousness of the technology may be hurt if Apple decides not to implement WebRTC on its browser, or if Microsoft's CU-RTC-Web alternative specification ends up fundamentally incompatible with WebRTC. Third, WebRTC would need to prove itself as a reliable and trustworthy technology. Nevertheless, the overall outlook for WebRTC's feasibility and applicability is positive.

WebRTC deployment opens up opportunities for various players in the market. First and foremost, WebRTC allows developers to easily integrate voice and video into their existing or new applications. Being openly available to everyone, the technology is an enabler for future innovation. Furthermore, a new role in the market emerges — that of the WebRTC provider. The WebRTC provider is a B2B player — a PaaS provider, who offers implementation, APIs, infrastructure and value-added features for other service providers to use in their applications. This is clear opportunity for newly established companies, but interesting possibilities also lie with existing large communication service providers, such as Microsoft (Skype), Google (Hangouts) or telecom operators. In those cases, a well-established communications platform could be opened up to support deep integration with third party providers, further boosting adoption and, especially in the case of telecom operators, possibly opening up valuable new revenue streams.

While not a priority goal, interoperability with legacy communication systems is considered as an important feature of WebRTC. WebRTC-to-legacy gateway implementations allow interconnection between the WebRTC and existing services, such as PSTN and PLMN. The motivation to allow this is mainly because of the positive network effects that the existing service's user base brings, which can be key factor for success especially for newly created WebRTC-based services. In addition, this opens up opportunity for telecom operators to explore creating new ways of communication for their customers, possibly offsetting the decline of plain telephony.

7.2 Discussion

Our aim with this thesis was to evaluate WebRTC, as a technology which is actively being developed, from both technical and economic perspectives and discover whether there are any critical roadblocks that could prevent adoption. The main tool we utilized the protocol feasibility framework by Levä and Suomi (2013). We found the framework's core value in providing a checklist of important questions to consider, as part of its iterative steps.

The main source of information that we used to answer these questions was the interview study we conducted. Therefore, the reliability of our results largely depends on the reliability of that input. We acknowledge that there might be some bias as a result of the set of participants in the study. The same applies to our selection of resources from the Web, which supplemented the interview study, despite our effort to provide balanced view.

Our feasibility analysis focuses on a single WebRTC use case — one-to-one browser-to-browser communication on desktop. While this is clearly an important use case, it is not the only one, nor is it, perhaps, going to be the most widely encountered. However, perhaps even more general conclusion could be drawn. Even though further work and focus on outstanding issues is required in the standardization organizations, our findings suggest that there are no insurmountable obstacles in front of WebRT in regard to other use cases, such as mobile use and multiparty communication. Overall, we believe that WebRTC has received enough industry backing that it can be part of the future Web.

We used VNCs as a tool to facilitate answering to our second research question. We found greatly beneficial the VNC method's fitness to generate possible configurations out of relatively simple building blocks, such as the technical components, roles and actors. Admittedly, the five[1] VNCs that we presented are not exhaustive, nor did our analysis of each aim at being thorough and complete in all details. Nevertheless, we are of opinion that the possibilities and opportunities those VNCs represented — and our comments highlighted — were indeed highly illustrative of the potential impact of WebRTC adoption.

7.3 Future research

Based on the presented work, we can suggest several tracks for possible future research. First, a closer examination of WebRTC's use cases may be beneficial. Such research could, for example, attempt to analyze those use cases in more concrete context, such as consumer or enterprise use. The enterprise market is interesting in that it is home of diverse set of products and solutions from a number of vendors — from video conferencing equipment, through hardware and software infrastructure to client software, etc. — where WebRTC could have significant impact.

Second, our work has been purely qualitative. Therefore, quantitative research would be an obvious step forward to building a truly holistic view of WebRTC. In particular, we see that quantitative methods or models could be applied in studying, for example, WebRTC adoption rate, impact on the enterprise communications market and impact on mobile operator business.

Third, some of the presented cases for WebRTC PaaS provider role being taken by Google (with Hangouts), Microsoft (with Skype) or telecom operators (with their existing networks) are essentially examples of two-sided markets. On one side, they have the communication platform users and on the other, the third party service providers that might want to integrate with the platform. Although at the time of writing these are purely hypotheti-

[1]Four main VNCs, plus one alternative Gateway VNC (see Section 6.5.2).

cal cases, further research may shed light into their respective probability of occurrence.

References

AddLive (2013). *AddLive Features*. Accessed 10.10.2013. URL: http://www.addlive.com/addlive-features/.

Adobe (June 2011). *Statistics : Penetration by version*. Accessed 04.11.2013. URL: http://www.adobe.com/fi/products/flashplatformruntimes/statistics.displayTab3.html.

Alvestrand, H. (Feb. 20, 2013). *Overview: Real Time Protocols for Brower-based Applications*. Internet-Draft draft-ietf-rtcweb-overview-06. IETF Secretariat.

Amirante, A. et al. (2013). "On the seamless interaction between webRTC browsers and SIP-based conferencing systems". In: *Communications Magazine, IEEE* 51.4, pp. 42–47. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6495759.

Aras, C. et al. (1994). "Real-time communication in packet-switched networks". In: *Proceedings of the IEEE* 82.1, pp. 122–139. ISSN: 00189219. DOI: 10.1109/5.259431. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=259431.

Baugher, M. et al. (Mar. 2004). *The Secure Real-time Transport Protocol (SRTP)*. RFC 3711 (Proposed Standard). Updated by RFC 5506. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc3711.txt.

Bergkvist, A. et al. (Aug. 21, 2012). *WebRTC 1.0: Real-time Communication Between Browsers*. W3C Working Draft. http://www.w3.org/TR/2012/WD-webrtc-20120821/. Accessed 22.02.2013. W3C.

Bergkvist, A. et al. (Aug. 30, 2013). *WebRTC 1.0: Real-time Communication Between Browsers*. W3C Editor's Draft. http://dev.w3.org/2011/webrtc/editor/webrtc.html. Accessed 28.10.2013. W3C.

Bertin, E., Crespi, N., and L'Hostis, M. (Oct. 2011). "A few myths about telco and OTT models". In: *2011 15th International Conference on Intelligence in Next Generation Networks*, pp. 6–10. DOI: 10.1109/ICIN.2011.6081106. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6081106.

Broadcom (May 2010). *Broadcom Accelerates WebM Video on Mobile Phones.* Accessed 10.10.2013. URL: http://www.broadcom.com/press/release.php?id=s471536.

Bubley, D. (2013a). *About Disruptive Analysis.* Accessed 10.10.2013. URL: http://disruptivewireless.blogspot.com/p/about.html.

Bubley, D. (Oct. 2013b). *For telcos, IMS integration should only be a small % of WebRTC effort.* Accessed 10.10.2013. URL: http://disruptivewireless.blogspot.fi/2013/10/for-telcos-ims-integration-should-only.html.

Bubley, D. (Nov. 2013c). *WebRTC codecs & Cisco open-sourcing H.264.* Accessed 11.11.2013. URL: http://disruptivewireless.blogspot.fi/2013/11/webrtc-codecs-cisco-open-sourcing-h264.html.

Bubley, D. (Feb. 2013d). *WebRTC Market Status & Forecasts: The hype is justified: it will change telecoms.* Accessed 10.10.2013. URL: http://disruptive-analysis.com/webrtc.htm.

Burman, B. (July 2012). *Ericsson's position on codecs.* Accessed 10.10.2013. URL: http://www.ietf.org/mail-archive/web/rtcweb/current/msg04938.html.

Business Wire (Mar. 2013). *Google and MPEG LA Announce Agreement Covering VP8 Video Format.* Accessed 10.10.2013. URL: http://www.businesswire.com/news/home/20130307006192/en/Google-MPEG-LA-Announce-Agreement-Covering-VP8.

Casey, T., Smura, T., and Sorri, A. (June 2010). "Value Network Configurations in wireless local area access". In: *2010 9th Conference of Telecommunication, Media and Internet.* IEEE, pp. 1–9. ISBN: 978-1-4244-7988-7. DOI: 10.1109/CTTE.2010.5557719. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5557719.

Cisco (Oct. 2013). *OpenH264 FAQ.* Accessed 08.11.2013. URL: http://www.openh264.org/faq.html.

CNN (May 2013). *How Google+ is pushing its way into all Google products.* Accessed 10.10.2013. URL: http://edition.cnn.com/2013/05/17/tech/web/google-plus-vic/index.html.

CSTB (2001). *The Internet's Coming of Age.* Committee on the Internet in the Evolving Information Infrastructure, Computer Science and Telecommunications Board, Commission on Physical Sciences, Mathematics, and Applications, National Research Council. The National Academies Press. ISBN: 9780309069922. URL: http://www.nap.edu/openbook.php?record_id=9823.

Deering, S. and Hinden, R. (Dec. 1998). *Internet Protocol, Version 6 (IPv6) Specification.* RFC 2460 (Draft Standard). Updated by RFCs 5095, 5722, 5871, 6437, 6564. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc2460.txt.

D'Esposito, J. E. and Gardner, R. M. (1999). "University students' perceptions of the Internet: an exploratory study". In: *The Journal of Academic Librarianship* 25.6, pp. 456 –461. ISSN: 0099-1333. DOI: http://dx.doi.org/10.1016/S0099-1333(99)00078-6. URL: http://www.sciencedirect.com/science/article/pii/S0099133399000786.

Eriksson, G. and Håkansson, S. (2012). *WebRTC: enhancing the web with real-time communication capabilities*. Tech. rep. http://www.ericsson.com/res/thecompany/docs/publications/ericsson_review/2012/er-webrtc-html5.pdf Accessed 13.02.2013.

Facebook (July 2011). *Call Your Friends Right From Facebook*. Accessed 10.10.2013. URL: https://www.facebook.com/notes/facebook/call-your-friends-right-from-facebook/10150223135777131.

Fischl, J., Tschofenig, H., and Rescorla, E. (May 2010). *Framework for Establishing a Secure Real-time Transport Protocol (SRTP) Security Context Using Datagram Transport Layer Security (DTLS)*. RFC 5763 (Proposed Standard). Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc5763.txt.

GENBAND (Apr. 2013). *GENBAND Fuels New Product Category with Launch of SPiDR, an Advanced WebRTC Gateway*. Accessed 10.10.2013. URL: http://www.genband.com/media-center/press-releases/genband-fuels-new-product-category-launch-spidr-advanced-webrtc-gateway.

Gigaom (Aug. 2013). *Exclusive: Google+ Hangouts goes HD as it switches from H.264 to VP8. Next up, WebRTC.* Accessed 10.10.2013. URL: http://gigaom.com/2013/08/28/hangouts-hd-vp8-webrtc/.

Google (Dec. 2009). *The meaning of open*. Accessed 10.10.2013. URL: http://googleblog.blogspot.fi/2009/12/meaning-of-open.html.

Google (2013a). *Supporting a Competitive Web*. Accessed 10.10.2013. URL: https://www.google.com/competition/competitiveweb.html.

Google (2013b). *WebRTC FAQ*. Accessed 10.10.2013. URL: http://www.webrtc.org/faq.

Handley, M. and Jacobson, V. (Apr. 1998). *SDP: Session Description Protocol*. RFC 2327 (Proposed Standard). Obsoleted by RFC 4566, updated by RFC 3266. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc2327.txt.

Holmberg, C., Hakansson, S., and Eriksson, G. (Dec. 19, 2012). *Web Real-Time Communication Use-cases and Requirements*. Internet-Draft draft-ietf-rtcweb-use-cases-and-requirements-10.txt. IETF Secretariat.

Huawei (Apr. 2013). *Huawei announced the industry's first WebRTC based rich communications capability exposure gateway.* Accessed 10.10.2013. URL: http://pr.huawei.com/en/news/hw-259848-webrtc.htm.

InfoWorld (July 2012). *Google backs open codecs for WebRTC.* Accessed 10.10.2013. URL: http://www.infoworld.com/d/application-development/google-backs-open-codecs-webrtc-198986.

ISC (2013). *ISC Domain Survey.* Internet Systems Consortuim. Accessed 31.10.2013. URL: https://www.isc.org/services/survey/.

Jennings, C., Hardie, T., and Westerlund, M. (2013). "Real-time communications for the web". In: *Communications Magazine, IEEE* 51.4, pp. 20–26. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6495756.

Jesup, R., Loreto, S., and Tuexen, M. (Feb. 25, 2013). *RTCWeb Data Channels.* Internet-Draft draft-ietf-rtcweb-data-channel-04.txt. IETF Secretariat.

Jobs, S. (Apr. 2010). *Thoughts on Flash.* Accessed 04.11.2013. URL: http://www.apple.com/hotnews/thoughts-on-flash/.

Johnston, A. B. and Burnett, D. C. (2012). *WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web.* Digital Codex LLC. ISBN: 0985978805, 9780985978808.

Johnston, A., Yoakum, J., and Singh, K. (2013). "Taking on webRTC in an enterprise". In: *Communications Magazine, IEEE* 51.4, pp. 48–54. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6495760.

Kelly, B. (May 2013). *Preparing for Disruption with WebRTC.* Accessed 10.10.2013. URL: http://www.nojitter.com/post/240155570/preparing-for-disruption-with-webrtc.

King, N. (2004). "Using interviews in qualitative research". In: *Essential guide to qualitative methods in organizational research.* Ed. by C. Cassell and G. Symon. SAGE Publications Limited, pp. 11–22.

Ko, M. N. et al. (2010). "Social-Networks Connect Services". In: *Computer* 43.8, pp. 37–43. ISSN: 0018-9162. DOI: 10.1109/MC.2010.239.

Kopetz, H. (2011). *Real-Time Systems.* Real-Time Systems Series. Boston, MA: Springer US. ISBN: 978-1-4419-8236-0. DOI: 10.1007/978-1-4419-8237-7. URL: http://link.springer.com/10.1007/978-1-4419-8237-7.

Kostopoulos, A. et al. (2012). "A Tussle Analysis for Information-Centric Networking Architectures". In: *The Future Internet.* Ed. by F. Álvarez et al. Vol. 7281. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 6–17. ISBN: 978-3-642-30240-4. DOI: 10.1007/978-3-642-30241-1_2. URL: http://dx.doi.org/10.1007/978-3-642-30241-1_2.

Levä, T., Komu, M., et al. (2013). "Adoption Barriers of Network-layer Protocols: the Case of Host Identity Protocol". In: *Computer Networks* 57.10, pp. 2218–2232.

Levä, T. and Suomi, H. (2013). "Techno-economic feasibility analysis of Internet protocols: Framework and tools". In: *Computer Standards & Interfaces* 36.1, pp. 76 –88. ISSN: 0920-5489. DOI: http://dx.doi.org/10.1016/j.csi.2013.07.011. URL: http://www.sciencedirect.com/science/article/pii/S0920548913000779.

Levä, T., Warma, H., et al. (2010). "Business Aspects of Multipath TCP Adoption". In: *Towards the Future Internet*. Ed. by G. Tselentis et al., pp. 21–30.

Levent-Levi, T. (Sept. 2012a). *H.264 vs VP8: Which is the Better Codec for WebRTC?* Accessed 10.10.2013. URL: http://bloggeek.me/webrtc-h264-vp8/.

Levent-Levi, T. (Sept. 2012b). *WebRTC: a new game-changer, disrupting Telcos and OTTs*. Accessed 10.10.2013. URL: http://www.visionmobile.com/blog/2012/09/webrtc-a-new-game-changer-disrupting-telcos-and-otts/.

Levent-Levi, T. (2013a). *About this blog*. Accessed 22.10.2013. URL: http://bloggeek.me/about/.

Levent-Levi, T. (June 2013b). *Can WebRTC Disrupt the Contact Center?* Accessed 10.10.2013. URL: http://blog.ucstrategies.com/2013/06/19/can-webrtc-disrupt-the-contact-center/.

Levent-Levi, T. (June 2013c). *How WebRTC Changed the Media Engine Market*. Accessed 10.10.2013. URL: http://bloggeek.me/webrtc-media-engine/.

Levent-Levi, T. (Mar. 2013d). *Optional Codecs in WebRTC is Science Fiction*. Accessed 10.10.2013. URL: http://bloggeek.me/optional-codecs-webrtc/.

Levent-Levi, T. (Aug. 2013e). *Panacea or not, Video Conferencing Vendors Can't Ignore WebRTC*. Accessed 10.10.2013. URL: http://bloggeek.me/webrtc-panacea/.

Levent-Levi, T. (Feb. 2013f). *There Will Be No WebRTC Skype Killer*. Accessed 10.10.2013. URL: http://bloggeek.me/webrtc-skype-killer/.

Levent-Levi, T. (May 2013g). *The Week that was in WebRTC: Gateways*. Accessed 10.10.2013. URL: http://bloggeek.me/webrtc-gateways-week/.

Levent-Levi, T. (June 2013h). *WebRTC Hype: Recheck...* Accessed 10.10.2013. URL: http://www.nojitter.com/post/240156061/webrtc-hype-recheck.

Levent-Levi, T. (2013i). *WebRTC Interviews*. Accessed 22.10.2013. URL: http://bloggeek.me/webrtc-interviews/.

Levent-Levi, T. (2013j). *WebRTC Posts*. Accessed 22.10.2013. URL: http://bloggeek.me/webrtc-posts/.

Loreto, S. and Romano, S. (Sept. 2012). "Real-Time Communications in the Web: Issues, Achievements, and Ongoing Standardization Efforts". In: *Internet Computing, IEEE* 16.5, pp. 68 –73. ISSN: 1089-7801. DOI: 10.1109/MIC.2012.115.

McDonald, W. (July 2013). *WebRTC: Is it the Panacea for the Video Conferencing Industry?* Accessed 10.10.2013. URL: http://www.nojitter.com/post/240158629/webrtc-is-it-the-panacea-for-the-video-conferencing-industry.

Michels, D. (Mar. 2013a). *5 Threats That Could Derail WebRTC*. Accessed 10.10.2013. URL: http://www.nojitter.com/post/240150516/5-threats-that-could-derail-webrtc.

Michels, D. (Apr. 2013b). *H.265 May Never Happen*. Accessed 10.10.2013. URL: http://www.nojitter.com/post/240153449/h265-may-never-happen.

Michels, D. (July 2013c). *Is Skype a WebRTC Killer?* Accessed 10.10.2013. URL: http://www.nojitter.com/post/240158376/is-skype-a-webrtc-killer.

Michels, D. (Apr. 2013d). *Is WebRTC a Threat to Skype?* Accessed 10.10.2013. URL: http://www.ucstrategies.com/unified-communications-strategies-views/is-webrtc-a-threat-to-skype.aspx.

Michels, D. (June 2013e). *WebRTC Hype Check*. Accessed 10.10.2013. URL: http://www.nojitter.com/post/240155921/webrtc-hype-check.

Michels, D. et al. (Nov. 2012). *WebRTC: "Revolutionary," or All Hype? (Part 1)*. Accessed 10.10.2013. URL: http://www.ucstrategies.com/industry-buzz/webrtc-revolutionary-or-all-hype.aspx.

Microsoft (May 2013). *Outlook.com now lets you chat with Google friends – one more reason to make the switch*. Accessed 10.10.2013. URL: http://blogs.office.com/b/microsoft-outlook/archive/2013/05/14/outlook-com-now-lets-you-chat-with-google-friends.aspx.

MPEG (Nov. 2013a). *Liaison on Video Coding for Browsers*. Accessed 11.11.2013. URL: http://www.ietf.org/mail-archive/web/rtcweb/current/msg09439.html.

MPEG (2013b). *The Moving Picture Experts Group website*. Accessed 08.11.2013. URL: http://mpeg.chiariglione.org/.

MPEG LA (2013). *AVC/H.264 Licensors*. Accessed 10.10.2013. URL: http://www.mpegla.com/main/programs/AVC/Pages/Licensors.aspx.

No Jitter (2013). *About No Jitter*. Accessed 10.10.2013. URL: http://www.nojitter.com/about/.

Nokia Corporation (Mar. 2013). *Nokia Corporation's Statement about IPR related to RFC 6386*. Accessed 10.10.2013. URL: https://datatracker.ietf.org/ipr/2035/.

NVIDIA (Feb. 2013). *NVIDIA Tegra 4 Family GPU Architecture*. Accessed 27.10.2013. URL: http://www.nvidia.com/docs/IO/116757/Tegra_4_GPU_Whitepaper_FINALv2.pdf.

Perkins, C., Westerlund, M., and Ott, J. (Feb. 25, 2013). *Web Real-Time Communication (WebRTC): Media Transport and Use of RTP*. Internet-Draft draft-ietf-rtcweb-rtp-usage-06. IETF Secretariat.

Postel, J. (Aug. 1980). *User Datagram Protocol*. RFC 768 (INTERNET STANDARD). Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc768.txt.

Postel, J. (Sept. 1981). *Internet Protocol*. RFC 791 (INTERNET STANDARD). Updated by RFCs 1349, 2474. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc791.txt.

Rao, B., Angelov, B., and Nov, O. (Apr. 2006). "Fusion of Disruptive Technologies:" in: *European Management Journal* 24.2-3, pp. 174–188. ISSN: 02632373. DOI: 10.1016/j.emj.2006.03.007. URL: http://linkinghub.elsevier.com/retrieve/pii/S0263237306000193.

Requestec (May 2013). *Requestec's WebRTC MCU Now Shipping*. Accessed 10.10.2013. URL: http://uk.prweb.com/releases/webrtc/mcu/prweb10788272.htm.

Rescorla, E. (Jan. 22, 2013a). *RTCWEB Security Architecture*. Internet-Draft draft-ietf-rtcweb-security-arch-06.txt. IETF Secretariat.

Rescorla, E. (Jan. 22, 2013b). *Security Considerations for RTC-Web*. Internet-Draft draft-ietf-rtcweb-security-04.txt. IETF Secretariat.

Rescorla, E. and Modadugu, N. (Jan. 2012). *Datagram Transport Layer Security Version 1.2*. RFC 6347 (Proposed Standard). Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc6347.txt.

Roach, A. (2013). *On the form of the question*. Accessed 11.11.2013. URL: http://www.ietf.org/mail-archive/web/rtcweb/current/msg09626.html.

Robson, C. (2002). *Real world research : a resource for social scientists and practitioner-researchers*. Ed. by N. Ed. Vol. 2nd. Blackwell, p. 624. URL: http://books.google.com/books?id=DkplMcAysFQC.

Rosenberg, J. (Apr. 2010). *Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols*. RFC 5245 (Proposed Standard). Updated by RFC 6336. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc5245.txt.

Rosenberg, J. and Schulzrinne, H. (June 2002). *An Offer/Answer Model with Session Description Protocol (SDP)*. RFC 3264 (Proposed Standard). Updated by RFC 6157. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc3264.txt.

Rosenberg, J., Schulzrinne, H., et al. (June 2002). *SIP: Session Initiation Protocol*. RFC 3261 (Proposed Standard). Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621, 5626, 5630, 5922, 5954, 6026, 6141, 6665. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc3261.txt.

Saltzer, J. H., Reed, D. P., and Clark, D. D. (Nov. 1984). “End-to-end arguments in system design”. In: *ACM Transactions on Computer Systems* 2.4, pp. 277–288. ISSN: 07342071. DOI: 10.1145/357401.357402. URL: http://portal.acm.org/citation.cfm?doid=357401.357402.

Sandgren, N., Mecklin, T., and Mäenpää, J. (Feb. 2012). *WebRTC Interworking with Traditional Telephony Services*. Accessed 10.10.2013. URL: http://labs.ericsson.com/blog/webrtc-interworking-with-traditional-telephony-services.

Schulzrinne, H. et al. (July 2003). *RTP: A Transport Protocol for Real-Time Applications*. RFC 3550 (INTERNET STANDARD). Updated by RFCs 5506, 5761, 6051, 6222. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc3550.txt.

Sisalem, D., Kuthan, J., and Ott, J. (2013). “A Short History of VoIP Services”. In: *Evolution of Telecommunication Services*. Ed. by E. Bertin, N. Crespi, and T. Magedanz. Vol. 7768. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 90–110. ISBN: 978-3-642-41568-5. DOI: 10.1007/978-3-642-41569-2_5. URL: http://dx.doi.org/10.1007/978-3-642-41569-2_5.

Skype (2013). *Skype URIs*. Accessed 10.10.2013. URL: https://developer.skype.com/skype-uris.

Stewart, R. (Sept. 2007). *Stream Control Transmission Protocol*. RFC 4960 (Proposed Standard). Updated by RFCs 6096, 6335. Internet Engineering Task Force. URL: http://www.ietf.org/rfc/rfc4960.txt.

Stockhammer, T., Hannuksela, M., and Wiegand, T. (July 2003). “H.264/AVC in wireless environments”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 13.7, pp. 657–673. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2003.815167. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1218198.

Telecompaper (2011). *Skype grows FY revenues 20%, reaches 663 mln users*. Accessed 10.10.2013. URL: http://www.telecompaper.com/news/skype-grows-fy-revenues-20-reaches-663-mln-users--790254.

TeleGeography (Feb. 2013). *The Bell Tolls for Telcos?* Accessed 04.11.2013. URL: http://www.telegeography.com/press/press-releases/2013/02/13/the-bell-tolls-for-telcos/index.html.

The Register (May 2010). *Google open sources $124.6m video codec*. Accessed 27.10.2013. URL: http://www.theregister.co.uk/2010/05/19/google_chrome_announcement/.

Thornburgh, M. (July 10, 2013). *Adobe's Secure Real-Time Media Flow Protocol*. Internet-Draft draft-thornburgh-adobe-rtmfp-10. IETF Secretariat.

TokBox (Oct. 2012). *TokBox – now a Telefónica Digital company*. Accessed 10.10.2013. URL: `http://www.tokbox.com/blog/tokbox-now-a-telefonica-digital-company/`.

TokBox (2013a). *Introduction to WebRTC and OpenTok*. Accessed 10.10.2013. URL: `http://tokbox.com/opentok/intro/`.

TokBox (Feb. 2013b). *What the CU-RTC-Web vs. WebRTC debate means for developers*. Accessed 10.10.2013. URL: `http://www.tokbox.com/blog/what-the-cu-rtc-web-vs-webrtc-debate-means-for-developers/`.

Tripathi, S. et al. (2013). "TCP-over-UDP for Real Time Applications". In: *International Journal of Scientific & Engineering Research - IJSER* 4.7, pp. 357–360.

Uberti, J. and Jennings, C. (Feb. 25, 2013). *Javascript Session Establishment Protocol*. Internet-Draft draft-ietf-rtcweb-jsep-03. IETF Secretariat.

Vetter, R., Spell, C., and Ward, C. (Oct. 1994). "Mosaic and the World Wide Web". In: *Computer* 27.10, pp. 49–57. ISSN: 0018-9162. DOI: `10.1109/2.318591`. URL: `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=318591`.

Viber (2013). *Viber Website*. Accessed 10.10.2013. URL: `http://www.viber.com/about`.

Vitek, C. (Nov. 2012). *Why IVR Cannot Survive in the WebRTC Era*. Accessed 10.10.2013. URL: `http://stcblog.com/2012/11/13/why-ivr-cannot-survive-in-the-webrtc-era/`.

WebM (2013a). *VP8 Bitstream Specification License*. Accessed 10.10.2013. URL: `http://www.webmproject.org/license/bitstream/`.

WebM (2013b). *WebM Video Hardware RTLs*. Accessed 10.10.2013. URL: `http://www.webmproject.org/hardware/`.

Xiph.Org (2013). *Opus FAQ*. Accessed 10.10.2013. URL: `http://wiki.xiph.org/OpusFAQ`.

Zittrain, J. (2008). *The Future of the Internet–And How to Stop It*. New Haven, CT, USA: Yale University Press. ISBN: 0300124872, 9780300124873.

Zmora, A. (May 2013). *WebRTC Gateways – Trivial But Can't Live Without Them*. Accessed 10.10.2013. URL: `http://bloggeek.me/webrtc-gateways-trivial/`.

Appendix A

Interview questions

A.1 Interview structure

In this section we show the main parts of the interviews and list the set of questions, which were used as basis for the interviews.

A.1.1 About the study

This study is done as part of my master thesis on the topic of techno-economic feasibility of WebRTC. The main focus of the thesis is to analyse whether WebRTC can be a successful technology, accounting for various stakeholder viewpoints; and assuming it is going to be a success, what areas of the market it will affect and how existing or new players can cope with that.

A.1.2 Interviewee background

1. Could you tell me a little bit about your background and current work?

2. What experience do you have with real-time communications?

3. In what activities around WebRTC are you involved?

A.1.3 Main questions

- WebRTC importance

    1. What is the most important feature of WebRTC?
    2. Depending on answer, ask about reach, interoperability, changing the architecture of the Web, platform for innovation?

3. Would you call WebRTC a game-changer? Why?
4. What use cases for WebRTC would you consider most important?
5. Is WebRTC a technology push or a market pull?

- Value proposition

1. Why is WebRTC important for your company?
2. How does WebRTC fit in your company's value proposition towards its customers?
3. How important is in-browser voice and video communication for enterprise video conferencing customers?
4. WebRTC applications may require certain infrastructure to be in place — STUN servers, relay servers, media servers (for conferencing). Who do you think is best positioned to provide these? Telecoms, ISPs, the application/service provider? Why? Who, if anyone, should pay for this?
5. What do you see as biggest costs in implementing your WebRTC service? Infrastructure, development, operational?

- Alternatives

1. What alternatives are there to WebRTC?
2. What are advantages or disadvantages of WebRTC compared to the alternatives?
3. Have you considered an alternative solution, providing you with similar features?
4. Do you consider WebRTC to be a competing or complementary solution to the traditional voice services?

- Challenges and solutions

1. What do you see as the main challenges in front of WebRTC?
2. What are specific challenges for WebRTC on mobile? Congestion control, hardware support for video encoding/decoding, QoS, inclusion in browsers?
3. Should WebRTC be concerned too much with interoperability with legacy systems? Is it time to start afresh?
4. How big a threat is non-compliance from Microsoft and/or Apple?
5. How big a problem is are legacy browser users?

6. What do you think might be feasible solutions to the challenges discussed?
7. What would you recommend to the standardization bodies to consider when continuing work on WebRTC?

- Future evolution

1. How do you see WebRTC's evolution in the future?

A.1.4 Feedback

1. Would you like to share your thoughts on something that was missed in our discussion?
2. Would you recommend that I interview someone else in your company?
3. Are you aware of some academic papers or industry whitepapers what are relevant to the study of WebRTC?