



Funded by the
European Union

WebRTC Testing: State of the Art

ICSOFT 2017

Boni García

`boni.garcia@urjc.es`

<http://elastest.io>

ICSOFT | 26th July 2017 | Madrid

Table of contents

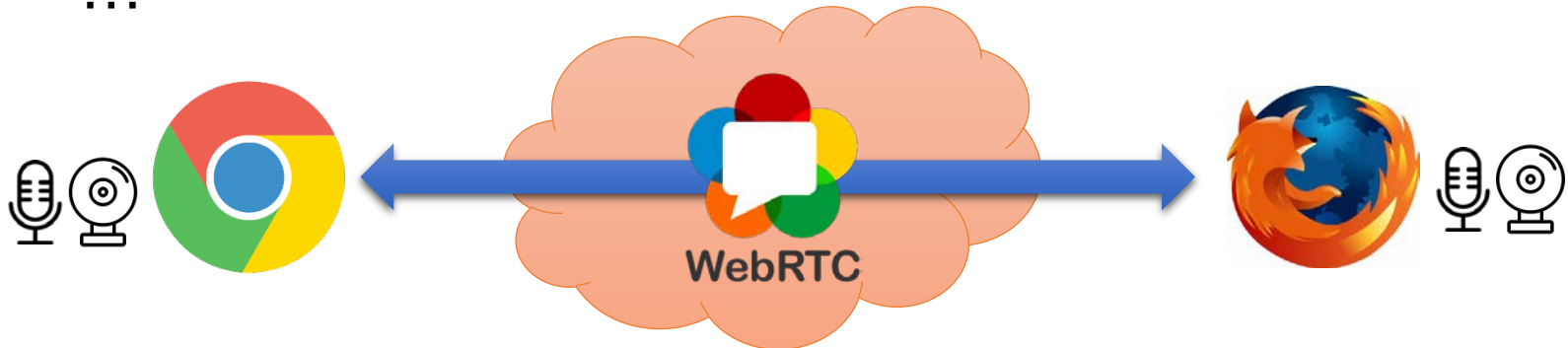


1. Introduction
2. Background
3. Scientific and academic research
4. Tools
5. Grey literature
6. Conclusions and future work

1. Introduction



- **WebRTC** is the set umbrella term for a number of novel technologies having the ambition of bringing high-quality *Real Time Communications* to the Web
 - W3C (JavaScript APIs): *getUserMedia*, *PeerConnection*, *DataChannels*
 - IETF (protocol stack): ICE, SDP, TURN, STUN, DTLS, ...



1. Introduction



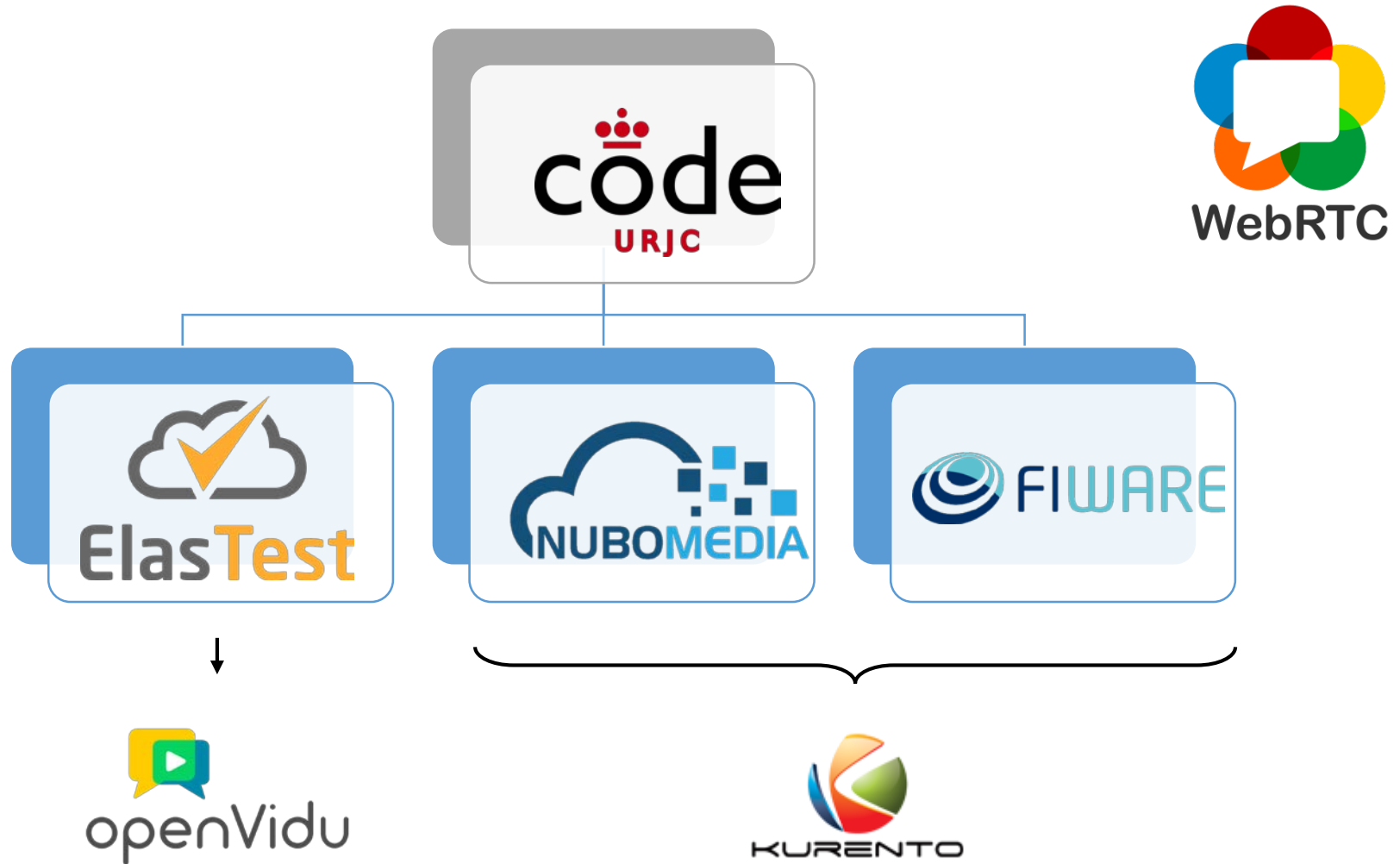
- Strong rate of growth of WebRTC since its inception 2011
 - IP video traffic will be 82 percent of all consumer Internet traffic by 2020 (Cisco Index, 2016)
 - 7 billion devices compliant WebRTC by 2020 (Sal and Rebbeck, 2014)
- It is imperative to have a strategy in place in order to assess WebRTC
- Nevertheless, testing WebRTC based application in a consistently automated fashion is a challenging problem

1. Introduction



- Objective: survey the state of the art of testing for WebRTC-based applications
- This survey aggregates 3 different sources of information:
 1. **Scientific papers** and articles in peer-reviewed journals, magazine, and international conferences
 2. Public available **WebRTC testing tools**, both commercial and open source
 3. Contributions available in the so-called “**grey literature**” (technical reports, white papers, newsletters, blogs, etc.)

2. Background



3. Scientific and academic research



- Search engines used:
 1. Google Scholar
<http://scholar.google.com/>
 2. CiteSeerx
<http://citeseerx.ist.psu.edu/>
 3. Microsoft Academic Research
<http://academic.research.microsoft.com/>
 4. ScienceDirect
<http://www.sciencedirect.com/>

3. Scientific and academic research



- **Results:**

Title	Keywords	Reference
On-Demand WebRTC Tunneling in Restricted Networks	Black-box testing, QoS, networking	(Sandholm et al., 2013)
WebRTC quality assessment: Dangers of black-box testin	Black-box testing, QoS, objective QoE	(Cinar and Melvin, 2014)
The impact of mobile device factors on QoE for multi-party video conferencing via WebRTC	Subjective QoE	(Vucic and Skorin-Kapov, 2015)
WebRTCbench: a benchmark for performance assessment of WebRTC implementations	Performance testing, framework, open source	(Taheri et al., 2015)

3. Scientific and academic research



- **Results:**

Title	Keywords	Reference
Jattack: a WebRTC load testing tool	Load testing, QoS, framework	(Amirante et al., 2016)
Performance comparison of a WebRTC server on Docker versus virtual machine	Load testing, QoS	(Spoiala et al., 2016)
Testing Framework for WebRTC Services	Black-box testing, QoE, QoS, framework, open source	(García et al., 2016a)
Analysis of Video Quality and End-to-End Latency in WebRTC	Load testing, QoS, objective QoE, framework, open source	(García et al., 2016b)
WebRTC Testing: Challenges and Practical Solutions	Load testing, QoS, objective QoE, framework, networking	(García et al., 2017)

3. Scientific and academic research



- Findings/conclusions:
 - Each contribution is focused in an specific domain (load testing, black-box, QoS, networking, ...)
 - Implementation not always available
 - Fully integrated solution not available
 - Quality of Experience (QoE) is a promising research line applied to WebRTC

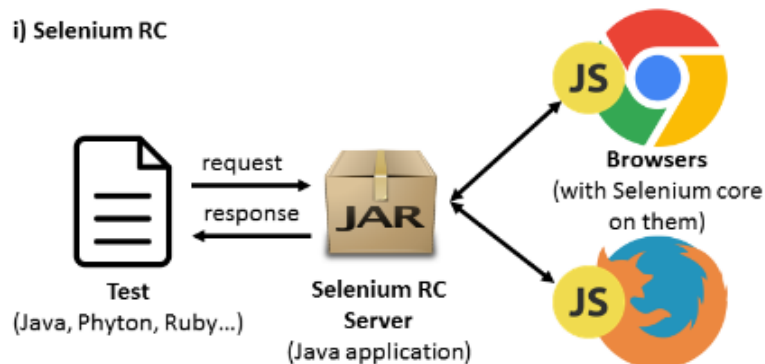
4. Tools



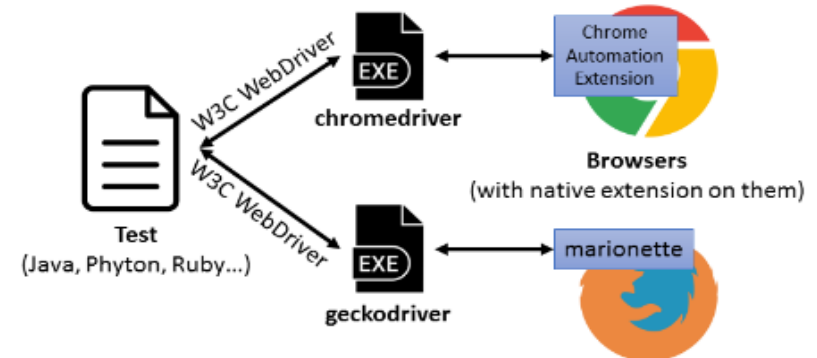
1. Selenium

- Open source testing framework which allows to control real browsers using different programming languages

i) Selenium RC



ii) Selenium WebDriver



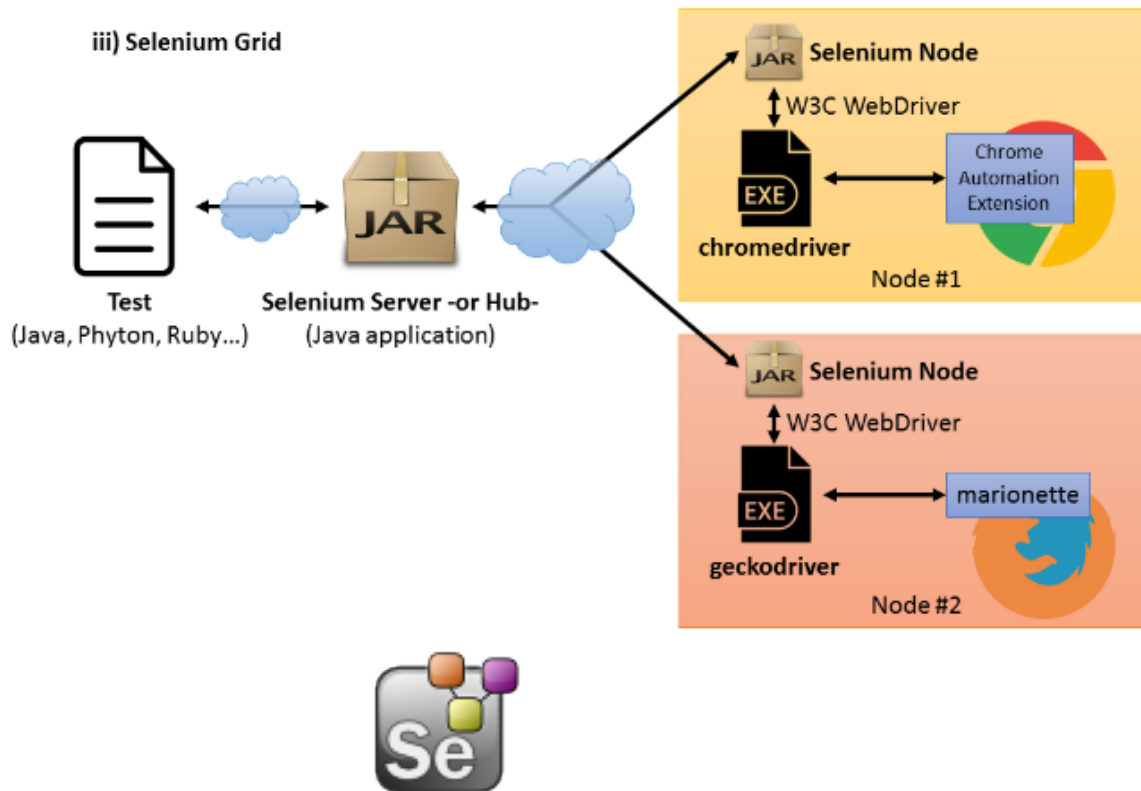
Deprecated!



4. Tools



1. Selenium



Cloud providers



4. Tools



2. TestRTC

- Commercial integrated platform aimed to test, monitor and analyze WebRTC-based communications
 - Use of real browsers
 - JavaScript API (built on the top of Nightwatch.js)
 - Network awareness
 - WebRTC tests at scale
 - Monitor Key Performance Indicators (KPIs) such as channel types, bitrate, timing, packet loss, and jitter
 - WebRTC-internals analyzer
 - Live preview of the remote browser
 - ...



4. Tools



- Findings/conclusions:
 - In the open source arena, Selenium is a must-know tool to carry out automated testing of WebRTC-based application, but:
 - Specific capabilities for WebRTC are not available
 - Usually, QA teams builds its own testing frameworks on the top of Selenium (WebDriver/Grid)
 - In the commercial arena, TestRTC offers an integrated powerful solution for testing WebRTC-based applications
 - Convenient for companies and large projects
 - You need to pay for it, so it is difficult to assume for open source or small project

5. Grey literature



- Sources:

1. Google Testing Blog

<https://testing.googleblog.com/>

2. Google Test Automation Conference (GTAC)

<https://developers.google.com/google-test-automation-conference/>

3. WebRTC Conference

<https://webrtc-conference.com/>

4. BlogGeek.me

<https://bloggeek.me/>

5. WebRtcHacks.com

<https://webrtchacks.com/>

5. Grey literature



- Results

Title	Keywords	Reference
WebRTC Audio Quality Testing	Black-box testing, objective QoE	(Höglund, 2013a)
Automated Video Quality Measurements	Black-box testing, objective QoE	(Höglund, 2013b)
Chrome-Firefox WebRTC Interop Test	Interoperability testing	(Höglund, 2014)
Audio Testing - Automatic Gain Control	Black-box testing	(Höglund, 2015)
The WebRTC Troubleshooter: test.webrtc.org	Black-box testing, QoS	(Pascual, 2015)
Overcoming the Challenges in Testing WebRTC Services	Testing methodology	(Levent-levi, 2015)
Quality Assurance for WebRTC Services	Testing methodology	(Levent-levi, 2016)

5. Grey literature



- Findings/conclusions:
 - Spread contributions (tools, methodologies, ...)
 - Quite interesting to follow the right people to be in touch with the latest trends

6. Conclusions and future work



- WebRTC is a set of technologies aimed to provide real time media capabilities to web applications
 - WebRTC applications are more and more demanded
 - They involve complex, distributed and heterogeneous network topologies (testing is not trivial)
- In the light of results, we conclude there are significant effort in the field of WebRTC testing
- There is room for improvement in several aspects, such as QoE and integrated open source solutions

6. Conclusions and future work



- Future work: ElasTest (H2020 project)
 - The objective of ElasTest is to provide a flexible open source testing platform aimed to simplify the end-to-end testing processes for different types of applications (among them, WebRTC-based applications)





Funded by the
European Union

Thank you

QA

Boni García

`boni.garcia@urjc.es`

<http://elastest.io>

ICSOFIT | 26th July 2017 | Madrid