Bettercast: a DIY Wireless Projector using any Raspberry PI

Goal

Turn the projector in the meeting room (the TV in the entrance, or any screen for that matter) into a cableless projector by using a Raspberry Pi.
It's been done before and it's not too complicated:
http://www.instructables.com/id/Raspberry-Pi-Wireless-Projector/
However, the solution above uses VNC and requires interaction with the Raspberry.

Ideally, we want to automate the process so that anybody in the room can start projecting from her/his own laptop with one click[1]. We can think of more meeting support functionalities we would want and adjust the project while running. You are very welcome to come up with other ICCLab-specific features.

How to get there

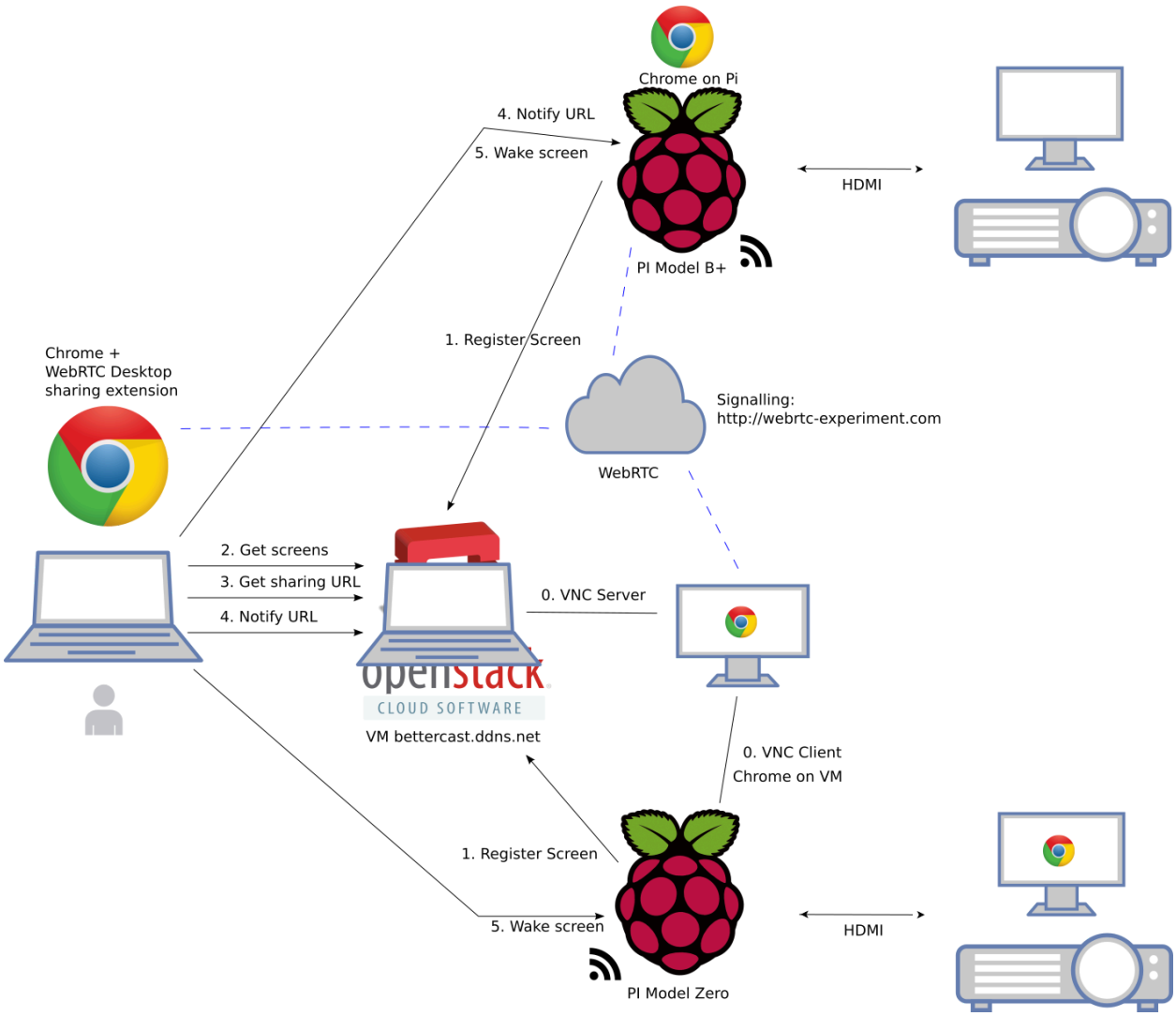
Avoiding the installation of OS-specific software on our laptops will save us a considerable development effort allowing us to concentrate on functionality. Using a commonly available teleconferencing software or a browser seem to be a good option.
Initial experiments with teleconferencing software show that the highest quality option is WebRTC.

Several examples are available online:
https://www.webrtc-experiment.com/
This extension in works well on Chrome and can be installed easily:
https://chrome.google.com/webstore/detail/webrtc-desktop-sharing
It generates a URL pointing to a WebRTC room werbrtc-experiment.com on that can be accessed for screen sharing.
All we would need to do is automate the accessing of this URL from the Raspberry Pi (a simple webserver would do). ~~*The question then is whether the browser in Raspberry Pi supports WebRTC and how well.*~~ *Initial experiments confirm that Raspberry B can run Chromium in Raspbian, successfully providing us with an initial working setup that can show the intended functionality.*

[1] We can come up with a smarter locking system in case we want

We still need to manage all the "one-click projection"™ automation which will initially consist in discovering the Raspberry PI attached to the projector, sharing the URL of the room, and opening a full screen browser.
http://stackoverflow.com/questions/28556402/webrtc-on-raspberry-pi
https://www.raspberrypi.org/forums/viewtopic.php?f=43&t=105019

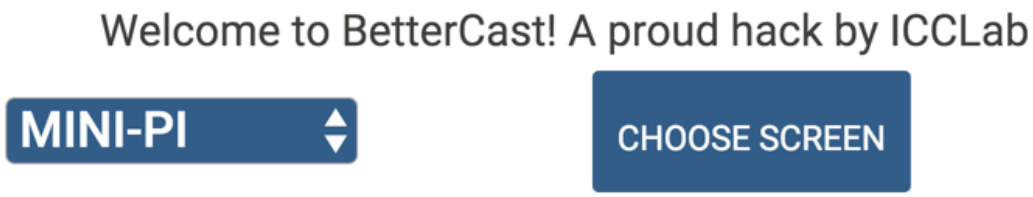
Architecture



How does it work?

Two types of Raspberry Pi were used: Zero and B+. The B+ has enough resources to run a full browser like Chrome on its own, while the Zero can't even start it. We thus had to think of different ways to broadcast a screen there.

First at startup each RPi registers on a central server, running on a VM accessible on bettercast.ddns.net.
Then when a client wants to share his screen, he goes to our webserver on the same address which allows him to choose between the registered RPis.



He is then redirected to a new page with a device specific URL, which needs to appear on the screen controlled by whichever Raspberry Pi he chose. If it's a B+ model, a wakeup (if sleeping) command is sent, then a notify action which starts a chrome browser locally on the Pi on the same URL. If it's a Zero, a VNC Server is opened on the VM, connected to a VNC client on the Pi. The actual WebRTC streaming in that case is done within the client and the VM. VNC being much more lightweight, it is able to display well on a Zero controlled screen.

When the client closes his sharing page, the page on the Pi is also automatically closed.