VRIJE UNIVERSITEIT AMSTERDAM

Green Lab

Ivano malavolta
i.malavolta@vu.nl

The Green Lab

A MASTER COURSE

Students measure real software products

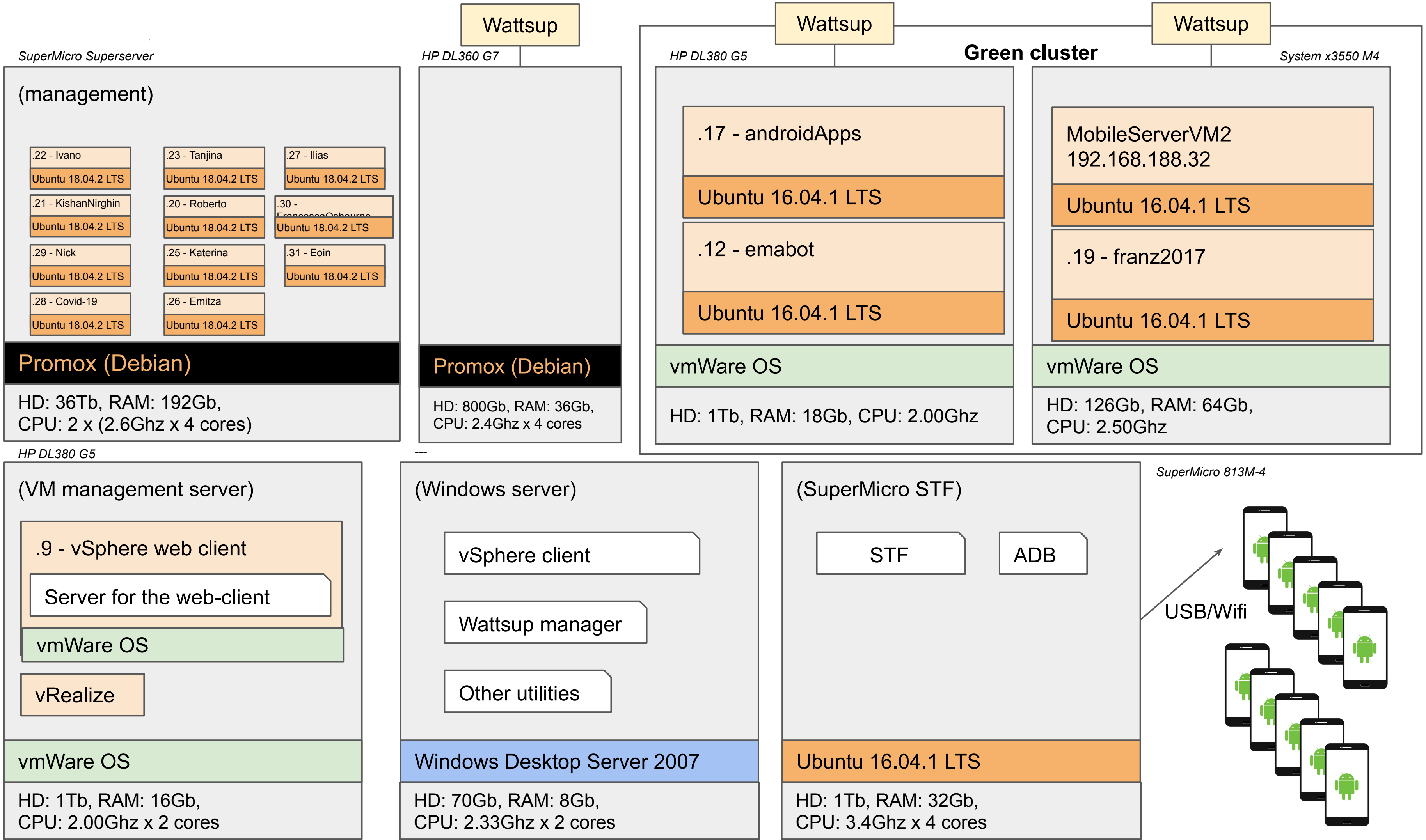
A PLATFORM

Our infrastructure for experimenting on software

- energy efficiency 🪫
- performance ⚡
- ...

A COLLABORATION PLATFORM

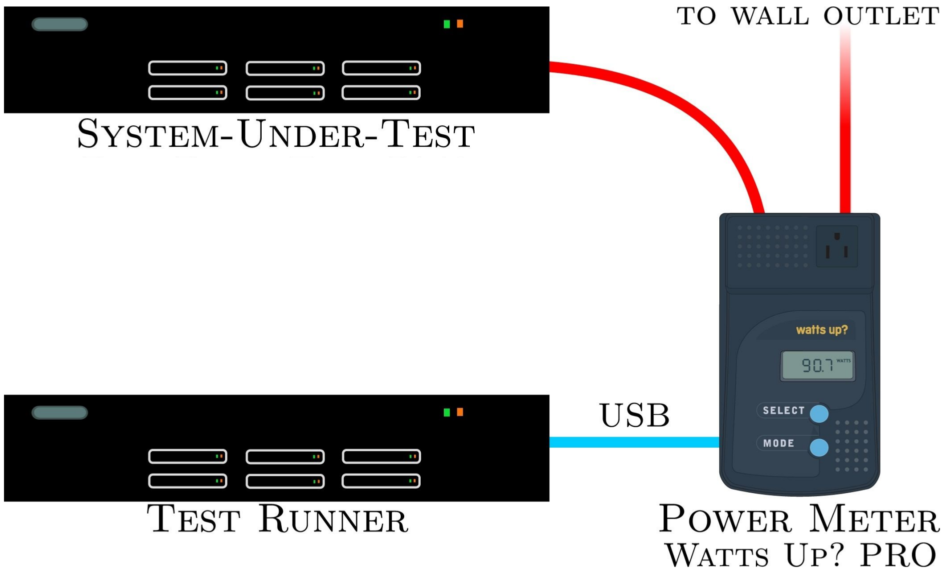
Industry-driven experiments


Wattsup
Wattsup
Wattsup
SuperMicro Superserver
HP DL360 G7
HP DL380 G5
Green cluster
System x3550 M4
(management)
.22 - Ivano
Ubuntu 18.04.2 LTS
.23 - Tanjina
Ubuntu 18.04.2 LTS
.27 - Ilias
Ubuntu 18.04.2 LTS
.21 - KishanNirghin
Ubuntu 18.04.2 LTS
.20 - Roberto
Ubuntu 18.04.2 LTS
.30 -
Ubuntu 18.04.2 LTS
.29 - Nick
Ubuntu 18.04.2 LTS
.25 - Katerina
Ubuntu 18.04.2 LTS
.31 - Eoin
Ubuntu 18.04.2 LTS
.28 - Covid-19
Ubuntu 18.04.2 LTS
.26 - Emitza
Ubuntu 18.04.2 LTS
Promox (Debian)
HD: 36Tb, RAM: 192Gb,
CPU: 2 x (2.6Ghz x 4 cores)
Promox (Debian)
HD: 800Gb, RAM: 36Gb,
CPU: 2.4Ghz x 4 cores
.17 - androidApps
Ubuntu 16.04.1 LTS
.12 - emabot
Ubuntu 16.04.1 LTS
vmWare OS
HD: 1Tb, RAM: 18Gb, CPU: 2.00Ghz
MobileServerVM2
192.168.188.32
Ubuntu 16.04.1 LTS
.19 - franz2017
Ubuntu 16.04.1 LTS
vmWare OS
HD: 126Gb, RAM: 64Gb,
CPU: 2.50Ghz
HP DL380 G5
(VM management server)
.9 - vSphere web client
Server for the web-client
vmWare OS
vRealize
vmWare OS
HD: 1Tb, RAM: 16Gb,
CPU: 2.00Ghz x 2 cores

(Windows server)
vSphere client
Wattsup manager
Other utilities
Windows Desktop Server 2007
HD: 70Gb, RAM: 8Gb,
CPU: 2.33Ghz x 2 cores
(SuperMicro STF)
STF
ADB
Ubuntu 16.04.1 LTS
HD: 1Tb, RAM: 32Gb,
CPU: 3.4Ghz x 4 cores
SuperMicro 813M-4
USB/Wifi

Watts up Pro

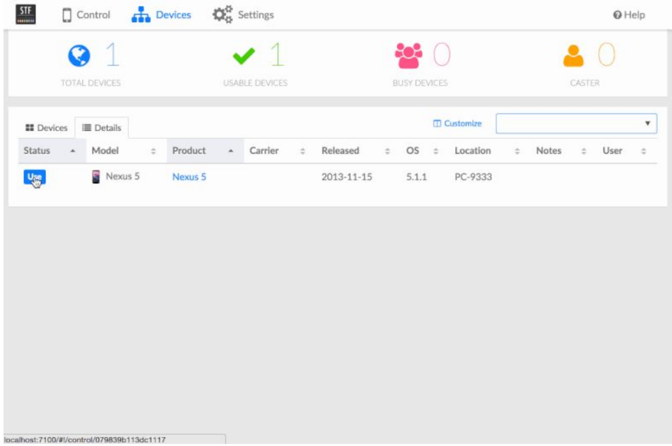
- Hardware Watt meter
- Measures instantaneous power
  - Logs it via USB[1]
  - Via Python or C utilities
- 60 Hz frequency
- Highly used in the literature[2]



[1] https://github.com/yyongpil/wattsup
[2] https://scholar.google.it/scholar?q=%22Watts+up%3F+Pro%22&hl=it&as_sdt=0,5
Image from: https://doi.org/10.1016/j.jss.2018.07.077

Smartphone Test Farm (STF)

- Debug/control several mobile devices remotely
- Web-based UI
- Real-time screen view
- Execute remote shell commands
- Manage device inventory

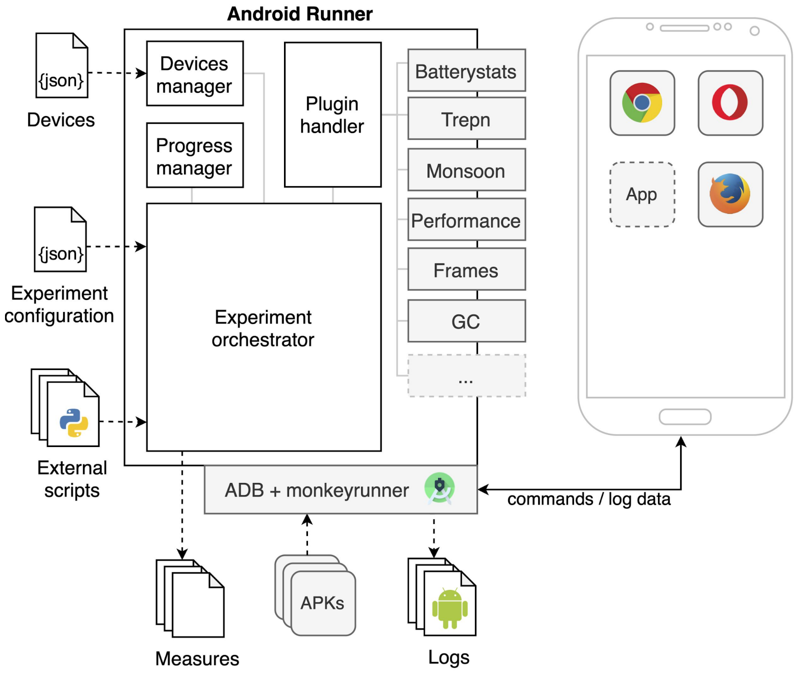


https://github.com/DeviceFarmer/stf

Experiments on *Android/web apps*

https://github.com/S2-group/android-runner

Automation	No interaction from the researcher
Incremental experiments	Pause/resume mechanism
Usability	Experiments defined in *descriptively*
Customizability	You can add your own business logic or automated testing tool
Profiler independence	Support for both HW/SW profilers, even in parallel
Experiments replicability	Easily execute an already-performed experiment



Android Runner as a learning platform

- Community of learners
- Discover and fix bugs
- Android Runner always up to date
- Learn the basics of OSS development

Green Lab

- **Green Lab** = Master course on empirical software engineering for energy efficient software
- Students use Android Runner as **black-box tool** for their own experiments

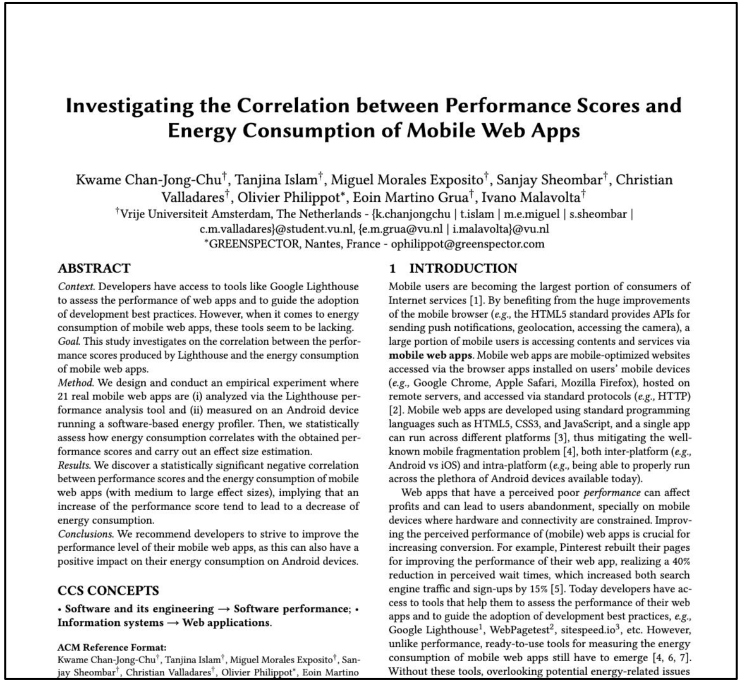
Android Runner

Final projects

- Students go deeper on run-time profiling
  - Plugins for new profilers
  - Improve Android Runner

VRIJE UNIVERSITEIT AMSTERDAM

Example of experiment on Web apps on Android



Investigating the Correlation between Performance Scores and Energy Consumption of Mobile Web Apps

Kwame Chan-Jong-Chu[†], Tanjina Islam[†], Miguel Morales Exposito[†], Sanjay Sheombar[†], Christian Valladares[†], Olivier Philippot[*], Eoin Martino Grua[†], Ivano Malavolta[†]
[†]Vrije Universiteit Amsterdam, The Netherlands - {k.chanjongchu | t.islam | m.e.miguel | s.sheombar | c.m.valladares}@student.vu.nl, {e.m.grua@vu.nl | i.malavolta}@vu.nl
[*]GREENSPECTOR, Nantes, France - ophilippot@greenspector.com

ABSTRACT

Context. Developers have access to tools like Google Lighthouse to assess the performance of web apps and to guide the adoption of development best practices. However, when it comes to energy consumption of mobile web apps, these tools seem to be lacking.
Goal. This study investigates on the correlation between the performance scores produced by Lighthouse and the energy consumption of mobile web apps.
Method. We design and conduct an empirical experiment where 21 real mobile web apps are (i) analyzed via the Lighthouse performance analysis tool and (ii) measured on an Android device running a software-based energy profiler. Then, we statistically assess how energy consumption correlates with the obtained performance scores and carry out an effect size estimation.
Results. We discover a statistically significant negative correlation between performance scores and the energy consumption of mobile web apps (with medium to large effect sizes), implying that an increase of the performance score tend to lead to a decrease of energy consumption.
Conclusions. We recommend developers to strive to improve the performance level of their mobile web apps, as this can also have a positive impact on their energy consumption on Android devices.

CCS CONCEPTS

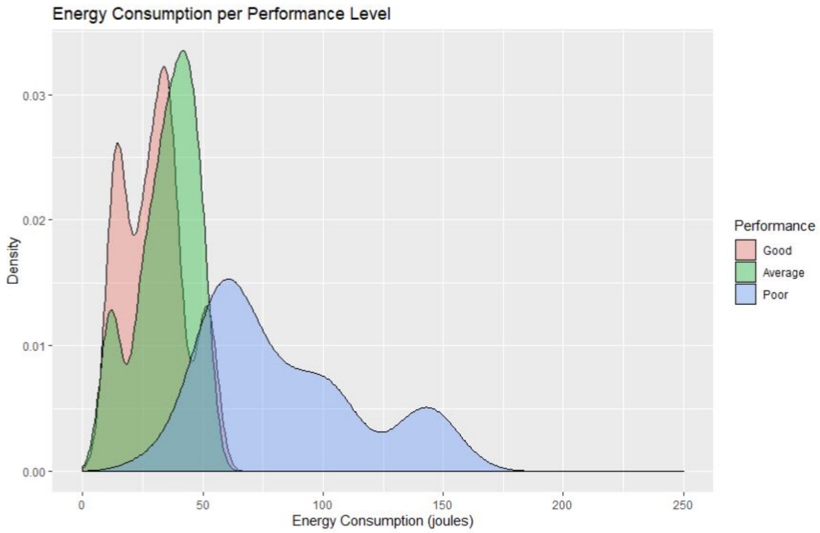
• **Software and its engineering → Software performance;** • **Information systems → Web applications.**

ACM Reference Format:
Kwame Chan-Jong-Chu[†], Tanjina Islam[†], Miguel Morales Exposito[†], Sanjay Sheombar[†], Christian Valladares[†], Olivier Philippot[*], Eoin Martino

1 INTRODUCTION

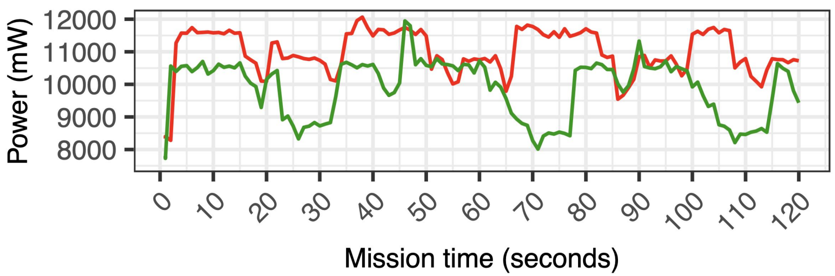
Mobile users are becoming the largest portion of consumers of Internet services [1]. By benefiting from the huge improvements of the mobile browser (*e.g.*, the HTML5 standard provides APIs for sending push notifications, geolocation, accessing the camera), a large portion of mobile users is accessing contents and services via **mobile web apps**. Mobile web apps are mobile-optimized websites accessed via the browser apps installed on users' mobile devices (*e.g.*, Google Chrome, Apple Safari, Mozilla Firefox), hosted on remote servers, and accessed via standard protocols (*e.g.*, HTTP) [2]. Mobile web apps are developed using standard programming languages such as HTML5, CSS3, and JavaScript, and a single app can run across different platforms [3], thus mitigating the well-known mobile fragmentation problem [4], both inter-platform (*e.g.*, Android vs iOS) and intra-platform (*e.g.*, being able to properly run across the plethora of Android devices available today).

Web apps that have a perceived poor *performance* can affect profits and can lead to users abandonment, specially on mobile devices where hardware and connectivity are constrained. Improving the perceived performance of (mobile) web apps is crucial for increasing conversion. For example, Pinterest rebuilt their pages for improving the performance of their web app, realizing a 40% reduction in perceived wait times, which increased both search engine traffic and sign-ups by 15% [5]. Today developers have access to tools that help them to assess the performance of their web apps and to guide the adoption of development best practices, *e.g.*, Google Lighthouse[1], WebPagetest[2], sitespeed.io[3], etc. However, unlike performance, ready-to-use tools for measuring the energy consumption of mobile web apps still have to emerge [4, 6, 7]. Without these tools, overlooking potential energy-related issues

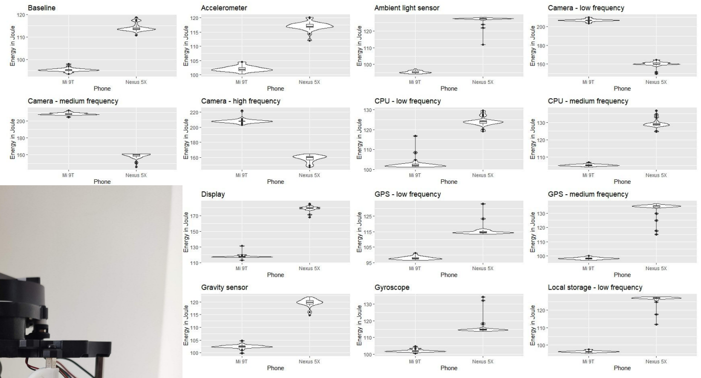


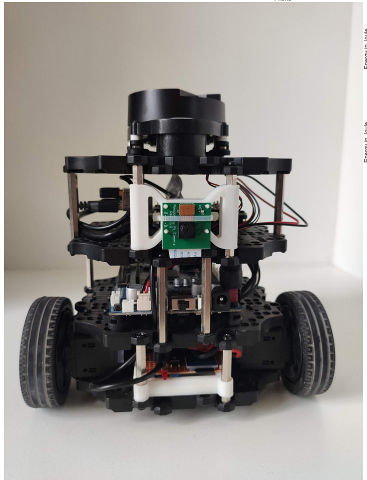
https://github.com/S2-group/ease-2020-replication-package

→ Experiments on Robotics



Energy measures

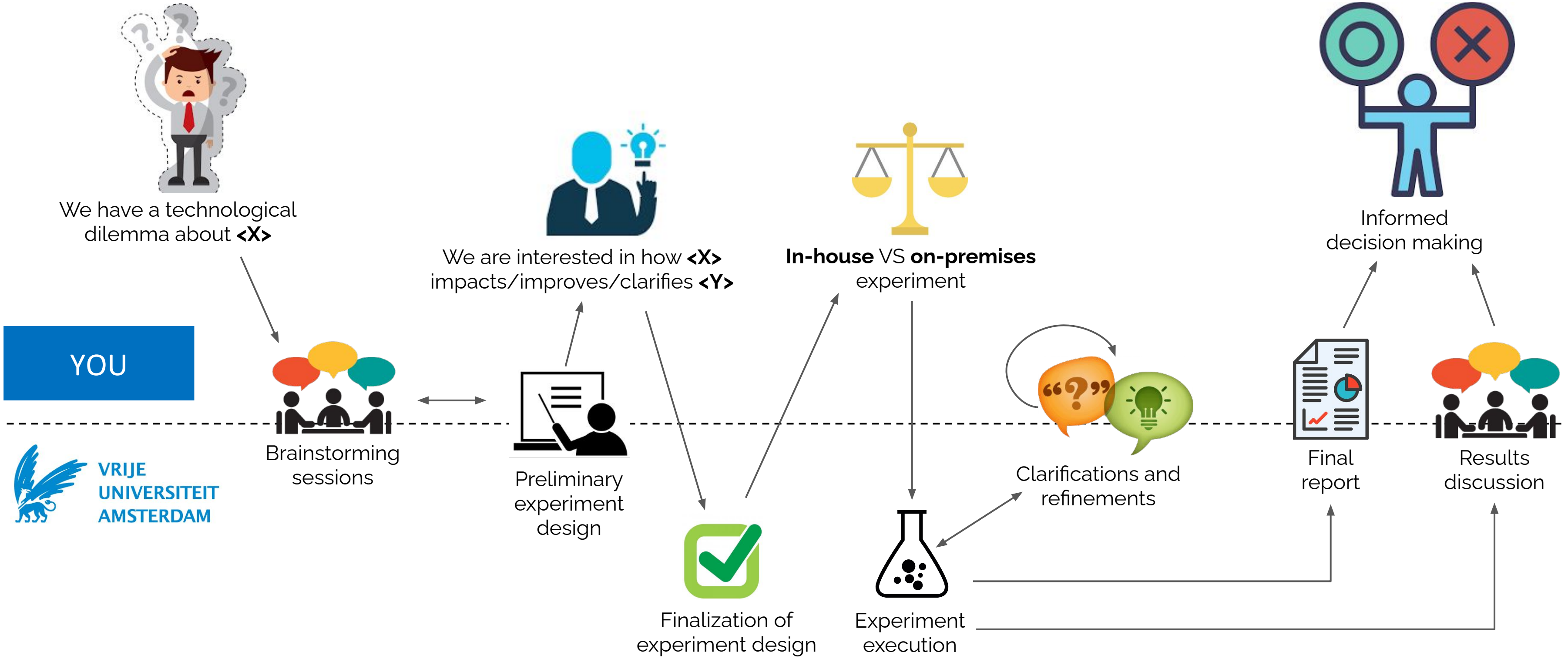




ROS

Robot Operating System

Collaboration model with companies



Contacts

Design/conduct experiments on your software?

Technological dilemmas?

Mining software repositories?

Group website
http://s2group.cs.vu.nl

Email
i.malavolta@vu.nl