

OpenQA Testing on Hardware: Automated Full System Tests

Laurence Urhegyi and James Thomas

Embedded Open Source Summit, Prague, June 27 2023



Introductions

- Laurence Urhegyi
 - Project Manager
 - laurence.urhegyi@codethink.co.uk
- James Thomas
 - Senior Engineer
 - james.thomas@codethink.co.uk

Both of us have a wealth of experience in the Automotive Industry, and are passionate about FOSS.

Outline of today's talk

Context: why testing matters in Automotive

Re-cap: some previous work presented on this topic

New advances: covering the progress made and why it's useful

Why testing matters in Automotive



Background: software complexity

Handling increasing software complexity is a huge challenge in the Automotive industry.

This [article by McKinsey](#) states:

'The latest automotive innovations, including intuitive infotainment, self-driving abilities, and electrification, depend less on mechanical ingenuity than on software quality, execution, and integration. This change is happening so rapidly that automotive OEMs and other industry stakeholders are now struggling to keep pace. The enormous cost of integrating and upgrading the features that consumers increasingly expect, including high-end onboard assistants and advanced driver-assistance systems (ADAS), is also daunting.'

Most automakers are not setup to support the software for the lifecycle of the vehicle.

Full system upgrades are rare and often avoided.

Codethink and our view

Codethink is a software services company.

We specialise in:

- open-source
- embedded Linux
- build/test/integration engineering

We see a lot of different automotive projects.

It's useful to think in terms of the Theory Of Constraints...

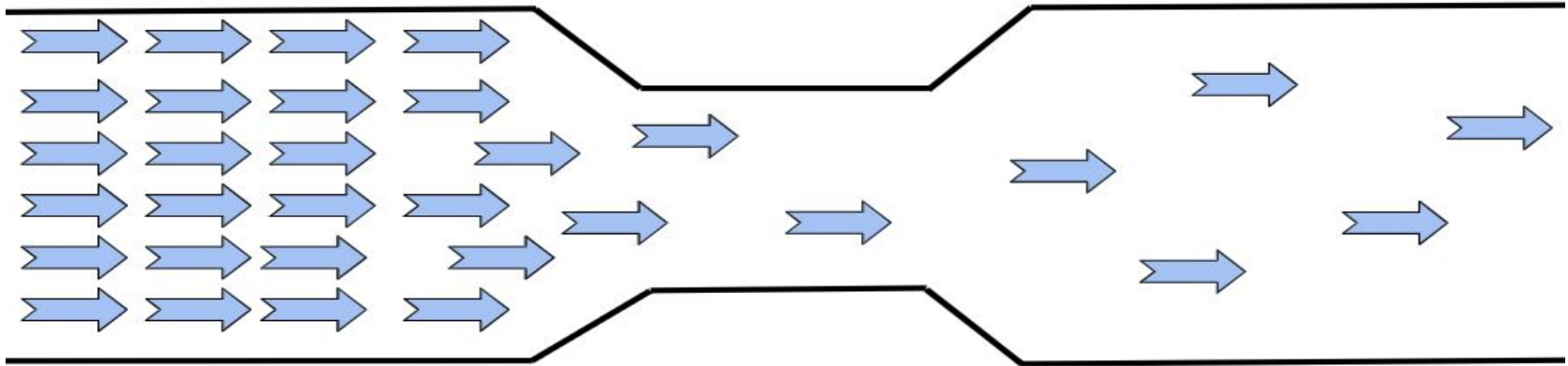
Theory of Constraints

By theory-of-constraints, the whole system is limited to the throughput of the bottleneck

Maximum capacity at the bottleneck = maximum capacity of whole system

Improvements at the bottleneck effect the entire system

Bottle-necked highway:



Removing bottlenecks

Invariably, Codethink sees the same patterns repeating.

Almost always, our involvement in projects will include **improvements to testing:**

- Helping to test earlier in delivery cycles, ie. *'shift left'*
- Improving testing reliability, via automation / CI/CD

System Validation

Testing is expensive.

Most projects rely on manual testing. Sometimes teams of hundreds or even >1000 people.

Hardware is often a prized resource,
especially in the early days of a project.

Image is free to use, credit:
<https://pixabay.com/photos/engineer-engineering-4941341/>



System Validation

We want to catch issues as early as possible.

The later we find the issue, the more expensive it becomes to resolve.

And we're trying to automate repetitive manual tasks, eg testing USB Media.

Re-cap on previous Ideas and efforts



Previously...

Codethink spoke on this topic at:

- [ELC 2021 \(Seattle\)](#)
- [FOSDEM 2022 \(Brussels\)](#)

The talk was entitled:

LAVA + OpenQA = Automated, Continuous, System Testing

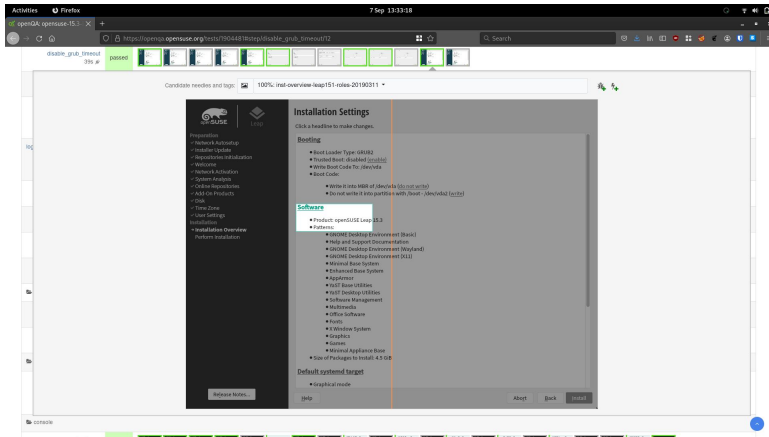
Mind the gap!

- Testing modern software platforms is extremely complex and time consuming
- Companies often miss out on the latest fixes and features in Linux, for fear of the time-consuming manual regression tests required
- Upstream says 'use the latest version'
- So there's a gap...

We thought we could help to reduce this gap if we made system testing more accessible.



- OpenQA for testing *software* (UI)
- Designed for Linux desktop operating system
- Tests based on screenshot comparisons (Needles)
- Central server to view test results, and to dispatch test jobs to workers
- Workers provide a specific machine type (e.g qemu-x86_64)
- Tests performed how the majority of users actually use the software
- Designed to work with QEMU



What's a needle / test in this context?

Needs:

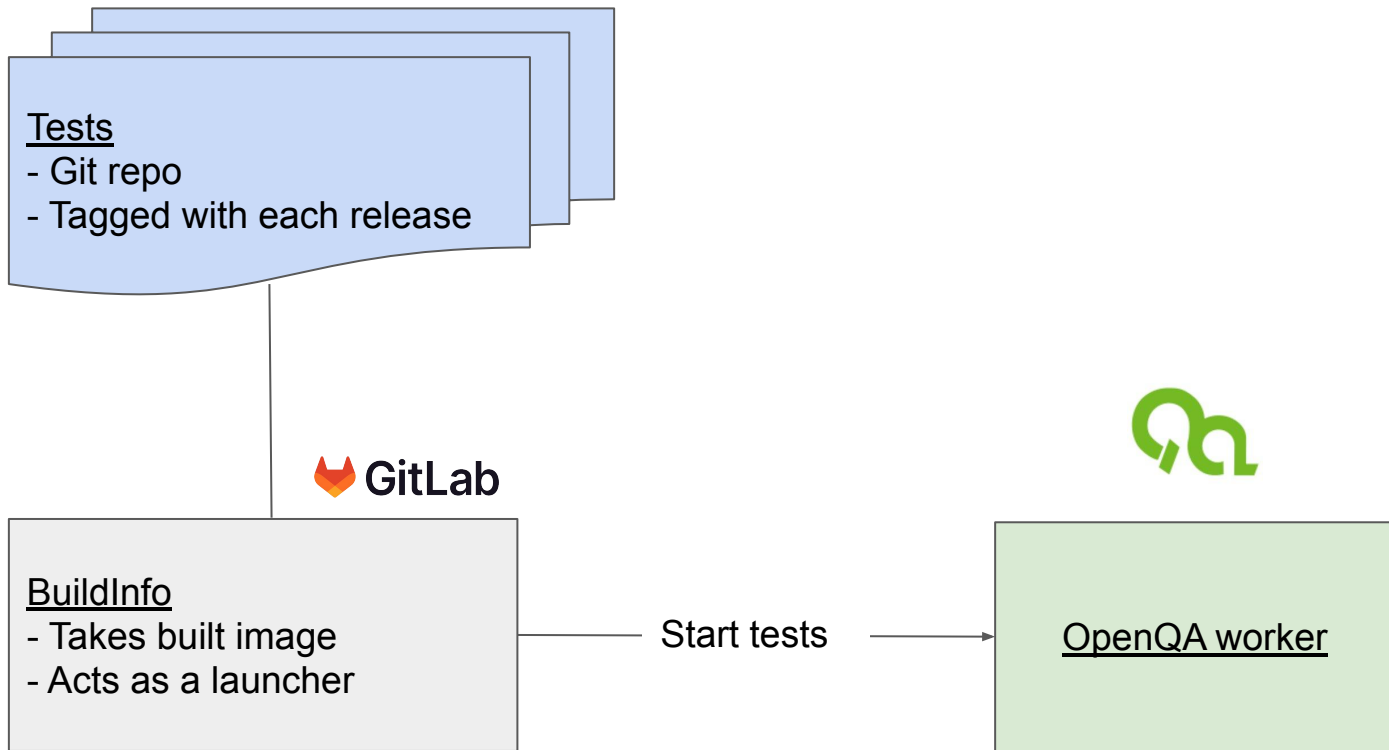
- Screenshot (picture_home)
 - Areas of the screen defined for match / click / mask
- JSON file (picture_home.json)
 - Coordinates

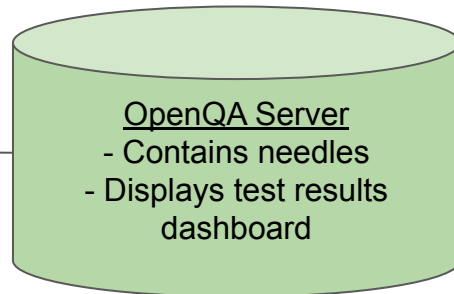
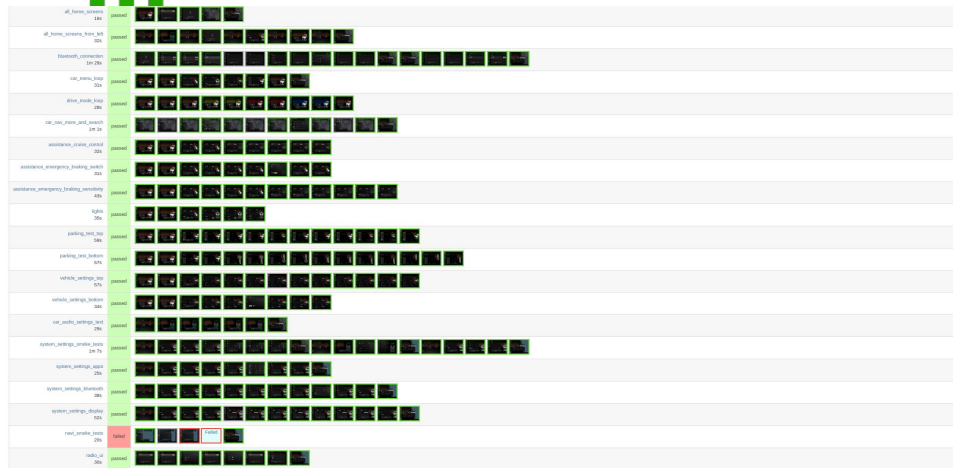
```
{  
  X: 0  
  Y: 0  
  Width: 100  
  Height: 130  
  Type: match  
}
```

Tests:

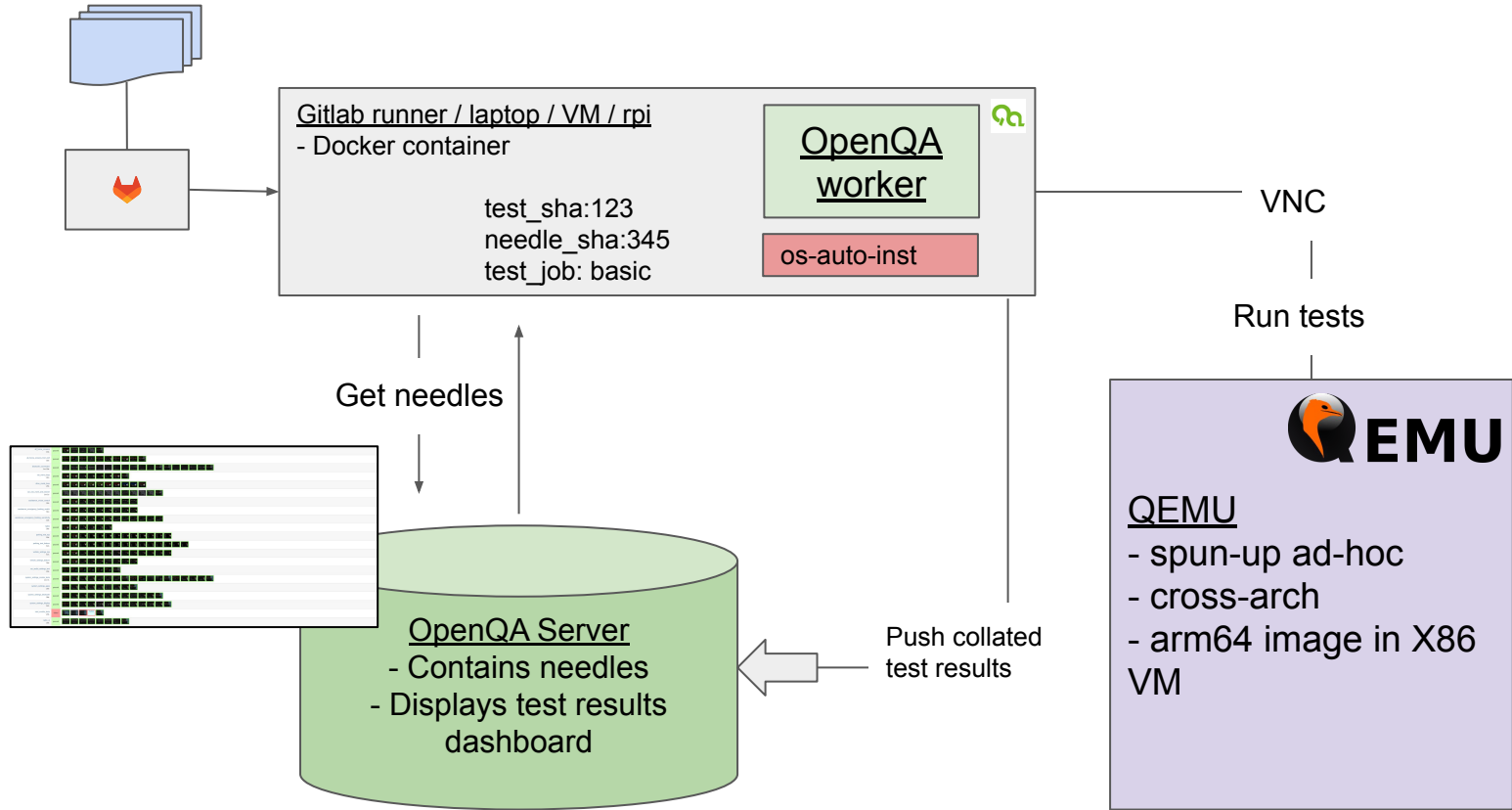
- Runs the needles
 - Click: (15)
 - Assert: (picture_home, 20)

Architecture: start tests






























































































































































































































































































































































































































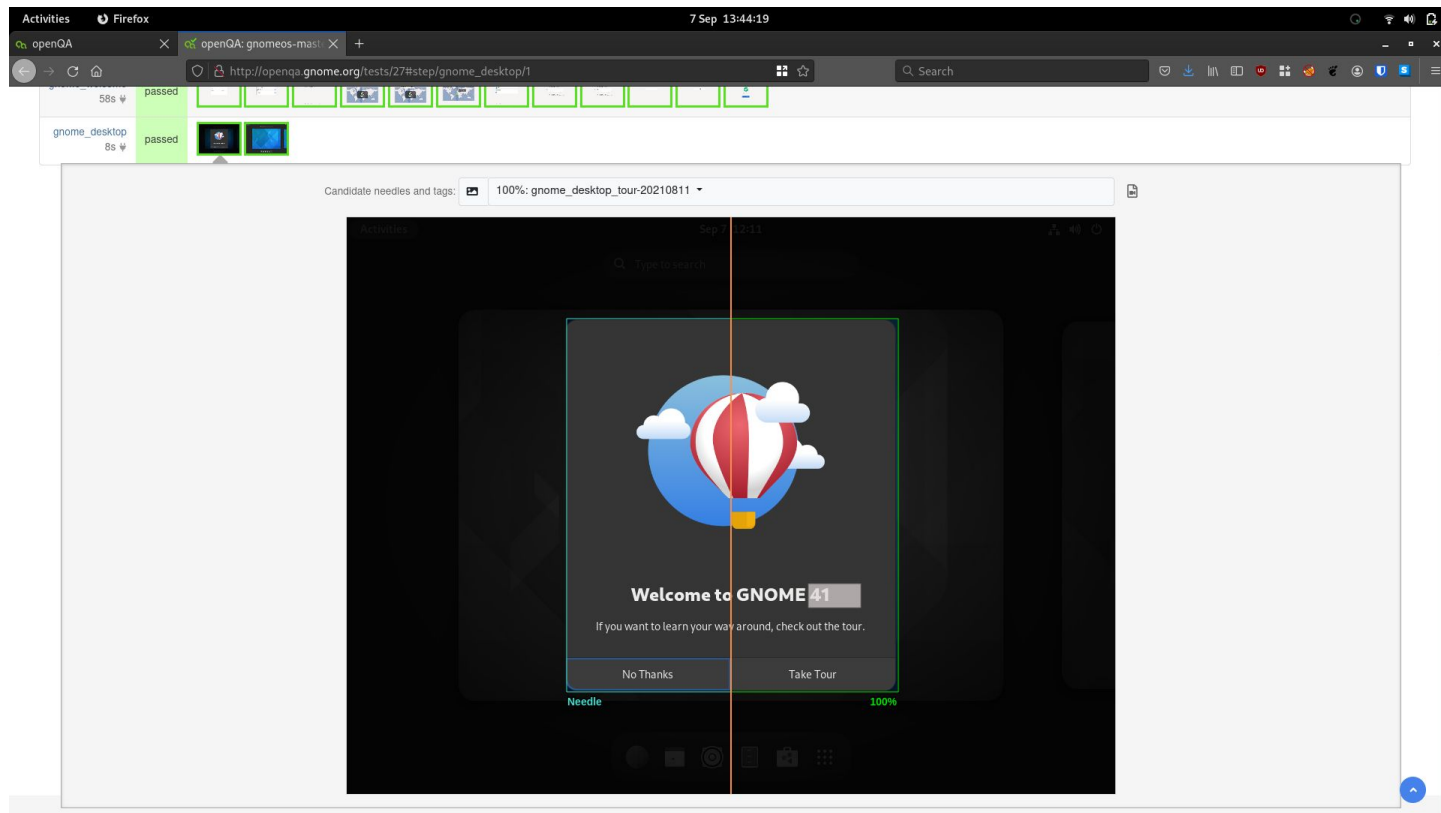
Architecture: OpenQA client: emulation



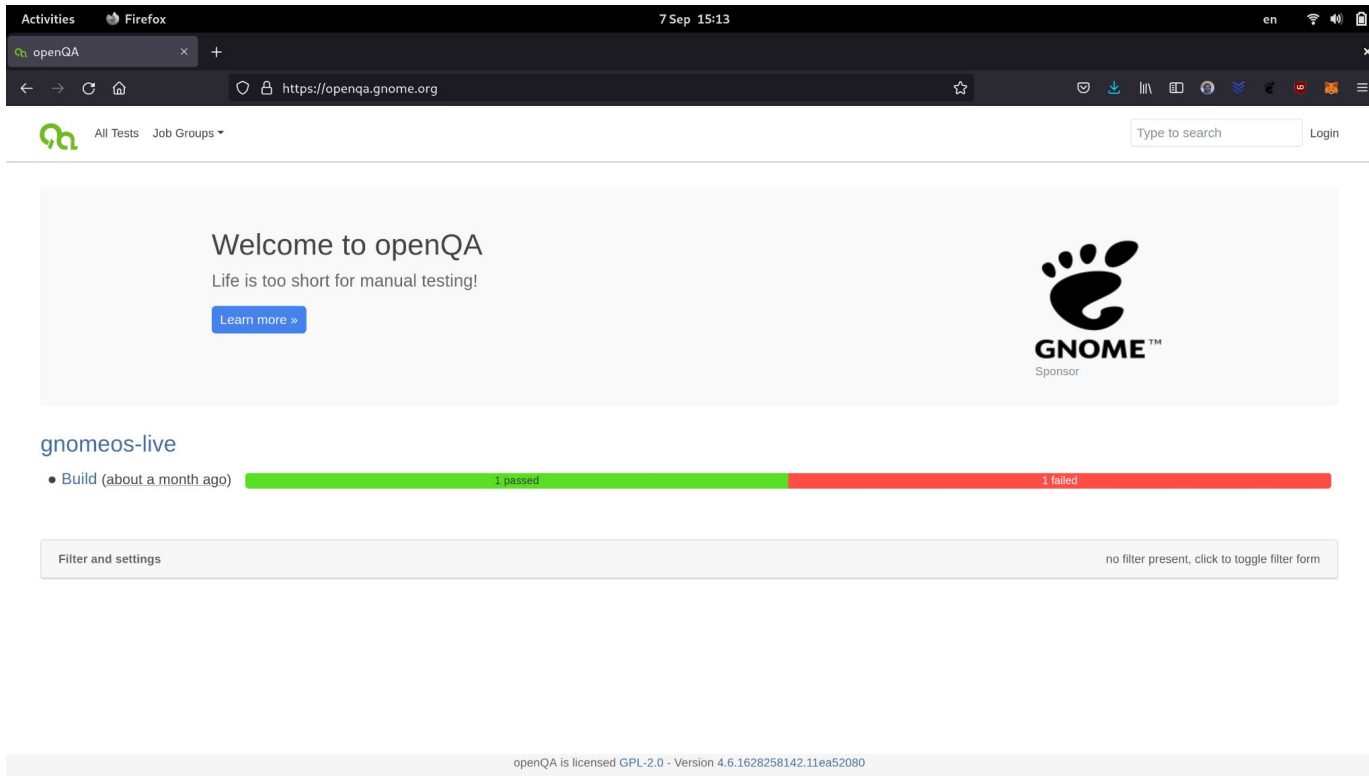
OpenQA Test Results View

all_home_screens 16s	passed	    
all_home_screens_from_left 32s	passed	         
bluetooth_connection 1m 29s	passed	                      
car_menu_loop 31s	passed	       
drive_mode_loop 28s	passed	         
car_nav_move_and_search 1m 1s	passed	                      
assistance_cruise_control 32s	passed	                      
assistance_emergency_braking_switch 31s	passed	                      
assistance_emergency_braking_sensitivity 43s	passed	                      
lights 35s	passed	       
parking_test_top 59s	passed	                      
parking_test_bottom 57s	passed	                      
vehicle_settings_top 57s	passed	                      
vehicle_settings_bottom 34s	passed	                      
car_audio_settings_test 29s	passed	                      
system_settings_smoke_tests 2m 7s	passed	                      
system_settings_apps 25s	passed	                      
system_settings_bluetooth 38s	passed	                      
system_settings_display 52s	passed	                      
navi_smoke_tests 20s	failed	                      
radio_ui 30s	passed	                      

CI Integration: GNOME



Being used already: openqa.gnome.org



The screenshot shows a Firefox browser window with the address bar displaying `https://openqa.gnome.org`. The page features a dark header with the openQA logo, navigation links for "All Tests" and "Job Groups", a search bar, and a "Login" button. The main content area has a light gray background with a "Welcome to openQA" message, the slogan "Life is too short for manual testing!", and a "Learn more" button. To the right is the GNOME logo with the text "GNOME™ Sponsor". Below this, the "gnomeos-live" section shows a build status for "Build (about a month ago)" with a progress bar indicating "1 passed" (green) and "1 failed" (red). A "Filter and settings" bar at the bottom states "no filter present, click to toggle filter form". The footer contains the license information: "openQA is licensed GPL-2.0 - Version 4.6.1628258142.11ea52080".

Activities Firefox 7 Sep 15:13 en

openQA x +


← → ↻ 🏠 🔒 https://openqa.gnome.org ☆

🔍 All Tests Job Groups ▼ Type to search Login

Welcome to openQA

Life is too short for manual testing!

[Learn more »](#)

 **GNOME™**
Sponsor

gnomeos-live

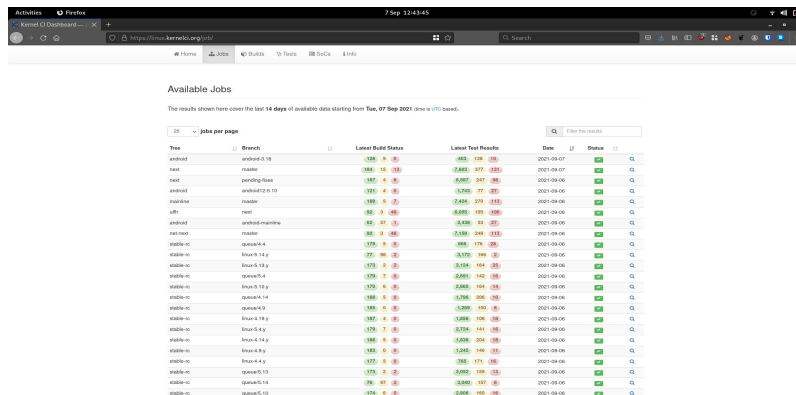
- Build (about a month ago) 1 passed 1 failed

Filter and settings no filter present, click to toggle filter form

openQA is licensed GPL-2.0 - Version 4.6.1628258142.11ea52080

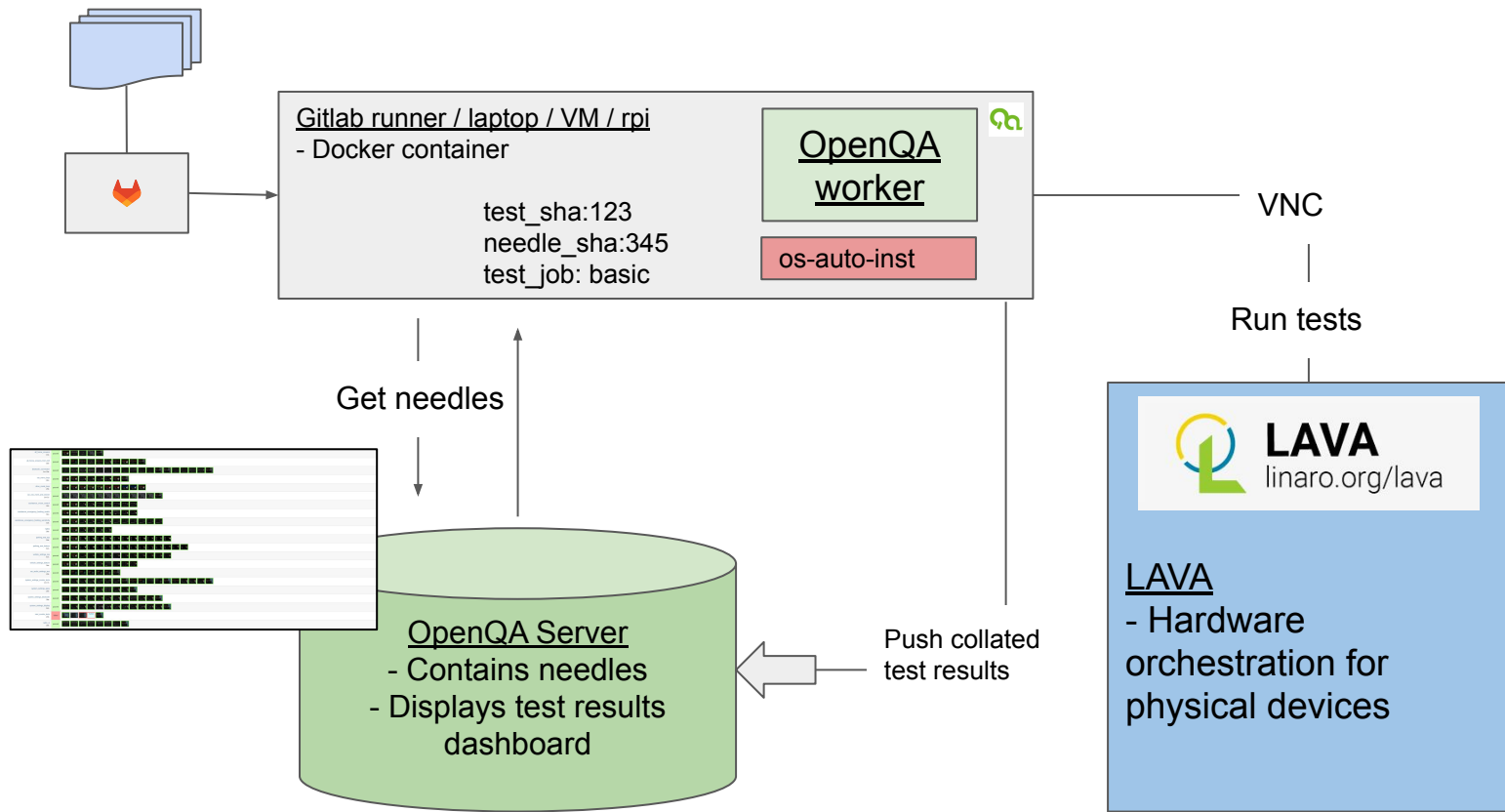
LAVA

- LAVA “continuous integration system for deploying operating systems onto physical and virtual hardware for running tests”
- Hardware orchestration for physical devices
- Central server that dispatches jobs to clients
- Used by KernelCI (<https://linux.kernelci.org/job/>) to test new kernel versions



The screenshot shows the Linux KernelCI job page. The browser address bar displays <https://linux.kernelci.org/job/>. The page title is "Available Jobs". Below the title, a note states: "The results shown here cover the last 14 days of available data starting from Tue, 07 Sep 2021 (time is UTC based)." There is a search bar and a "jobs per page" dropdown set to 25. The main content is a table with columns: Tree, Branch, Latest Build Status, Latest Test Results, Date, and Status. The table lists various jobs, including android, i386, mips, riscv, and various kernel versions (e.g., 5.10, 5.11, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.18, 5.19, 5.20, 5.21, 5.22, 5.23, 5.24, 5.25, 5.26, 5.27, 5.28, 5.29, 5.30, 5.31, 5.32, 5.33, 5.34, 5.35, 5.36, 5.37, 5.38, 5.39, 5.40, 5.41, 5.42, 5.43, 5.44, 5.45, 5.46, 5.47, 5.48, 5.49, 5.50, 5.51, 5.52, 5.53, 5.54, 5.55, 5.56, 5.57, 5.58, 5.59, 5.60, 5.61, 5.62, 5.63, 5.64, 5.65, 5.66, 5.67, 5.68, 5.69, 5.70, 5.71, 5.72, 5.73, 5.74, 5.75, 5.76, 5.77, 5.78, 5.79, 5.80, 5.81, 5.82, 5.83, 5.84, 5.85, 5.86, 5.87, 5.88, 5.89, 5.90, 5.91, 5.92, 5.93, 5.94, 5.95, 5.96, 5.97, 5.98, 5.99, 6.00, 6.01, 6.02, 6.03, 6.04, 6.05, 6.06, 6.07, 6.08, 6.09, 6.10, 6.11, 6.12, 6.13, 6.14, 6.15, 6.16, 6.17, 6.18, 6.19, 6.20, 6.21, 6.22, 6.23, 6.24, 6.25, 6.26, 6.27, 6.28, 6.29, 6.30, 6.31, 6.32, 6.33, 6.34, 6.35, 6.36, 6.37, 6.38, 6.39, 6.40, 6.41, 6.42, 6.43, 6.44, 6.45, 6.46, 6.47, 6.48, 6.49, 6.50, 6.51, 6.52, 6.53, 6.54, 6.55, 6.56, 6.57, 6.58, 6.59, 6.60, 6.61, 6.62, 6.63, 6.64, 6.65, 6.66, 6.67, 6.68, 6.69, 6.70, 6.71, 6.72, 6.73, 6.74, 6.75, 6.76, 6.77, 6.78, 6.79, 6.80, 6.81, 6.82, 6.83, 6.84, 6.85, 6.86, 6.87, 6.88, 6.89, 6.90, 6.91, 6.92, 6.93, 6.94, 6.95, 6.96, 6.97, 6.98, 6.99, 7.00, 7.01, 7.02, 7.03, 7.04, 7.05, 7.06, 7.07, 7.08, 7.09, 7.10, 7.11, 7.12, 7.13, 7.14, 7.15, 7.16, 7.17, 7.18, 7.19, 7.20, 7.21, 7.22, 7.23, 7.24, 7.25, 7.26, 7.27, 7.28, 7.29, 7.30, 7.31, 7.32, 7.33, 7.34, 7.35, 7.36, 7.37, 7.38, 7.39, 7.40, 7.41, 7.42, 7.43, 7.44, 7.45, 7.46, 7.47, 7.48, 7.49, 7.50, 7.51, 7.52, 7.53, 7.54, 7.55, 7.56, 7.57, 7.58, 7.59, 7.60, 7.61, 7.62, 7.63, 7.64, 7.65, 7.66, 7.67, 7.68, 7.69, 7.70, 7.71, 7.72, 7.73, 7.74, 7.75, 7.76, 7.77, 7.78, 7.79, 7.80, 7.81, 7.82, 7.83, 7.84, 7.85, 7.86, 7.87, 7.88, 7.89, 7.90, 7.91, 7.92, 7.93, 7.94, 7.95, 7.96, 7.97, 7.98, 7.99, 8.00, 8.01, 8.02, 8.03, 8.04, 8.05, 8.06, 8.07, 8.08, 8.09, 8.10, 8.11, 8.12, 8.13, 8.14, 8.15, 8.16, 8.17, 8.18, 8.19, 8.20, 8.21, 8.22, 8.23, 8.24, 8.25, 8.26, 8.27, 8.28, 8.29, 8.30, 8.31, 8.32, 8.33, 8.34, 8.35, 8.36, 8.37, 8.38, 8.39, 8.40, 8.41, 8.42, 8.43, 8.44, 8.45, 8.46, 8.47, 8.48, 8.49, 8.50, 8.51, 8.52, 8.53, 8.54, 8.55, 8.56, 8.57, 8.58, 8.59, 8.60, 8.61, 8.62, 8.63, 8.64, 8.65, 8.66, 8.67, 8.68, 8.69, 8.70, 8.71, 8.72, 8.73, 8.74, 8.75, 8.76, 8.77, 8.78, 8.79, 8.80, 8.81, 8.82, 8.83, 8.84, 8.85, 8.86, 8.87, 8.88, 8.89, 8.90, 8.91, 8.92, 8.93, 8.94, 8.95, 8.96, 8.97, 8.98, 8.99, 9.00, 9.01, 9.02, 9.03, 9.04, 9.05, 9.06, 9.07, 9.08, 9.09, 9.10, 9.11, 9.12, 9.13, 9.14, 9.15, 9.16, 9.17, 9.18, 9.19, 9.20, 9.21, 9.22, 9.23, 9.24, 9.25, 9.26, 9.27, 9.28, 9.29, 9.30, 9.31, 9.32, 9.33, 9.34, 9.35, 9.36, 9.37, 9.38, 9.39, 9.40, 9.41, 9.42, 9.43, 9.44, 9.45, 9.46, 9.47, 9.48, 9.49, 9.50, 9.51, 9.52, 9.53, 9.54, 9.55, 9.56, 9.57, 9.58, 9.59, 9.60, 9.61, 9.62, 9.63, 9.64, 9.65, 9.66, 9.67, 9.68, 9.69, 9.70, 9.71, 9.72, 9.73, 9.74, 9.75, 9.76, 9.77, 9.78, 9.79, 9.80, 9.81, 9.82, 9.83, 9.84, 9.85, 9.86, 9.87, 9.88, 9.89, 9.90, 9.91, 9.92, 9.93, 9.94, 9.95, 9.96, 9.97, 9.98, 9.99, 10.00, 10.01, 10.02, 10.03, 10.04, 10.05, 10.06, 10.07, 10.08, 10.09, 10.10, 10.11, 10.12, 10.13, 10.14, 10.15, 10.16, 10.17, 10.18, 10.19, 10.20, 10.21, 10.22, 10.23, 10.24, 10.25, 10.26, 10.27, 10.28, 10.29, 10.30, 10.31, 10.32, 10.33, 10.34, 10.35, 10.36, 10.37, 10.38, 10.39, 10.40, 10.41, 10.42, 10.43, 10.44, 10.45, 10.46, 10.47, 10.48, 10.49, 10.50, 10.51, 10.52, 10.53, 10.54, 10.55, 10.56, 10.57, 10.58, 10.59, 10.60, 10.61, 10.62, 10.63, 10.64, 10.65, 10.66, 10.67, 10.68, 10.69, 10.70, 10.71, 10.72, 10.73, 10.74, 10.75, 10.76, 10.77, 10.78, 10.79, 10.80, 10.81, 10.82, 10.83, 10.84, 10.85, 10.86, 10.87, 10.88, 10.89, 10.90, 10.91, 10.92, 10.93, 10.94, 10.95, 10.96, 10.97, 10.98, 10.99, 11.00, 11.01, 11.02, 11.03, 11.04, 11.05, 11.06, 11.07, 11.08, 11.09, 11.10, 11.11, 11.12, 11.13, 11.14, 11.15, 11.16, 11.17, 11.18, 11.19, 11.20, 11.21, 11.22, 11.23, 11.24, 11.25, 11.26, 11.27, 11.28, 11.29, 11.30, 11.31, 11.32, 11.33, 11.34, 11.35, 11.36, 11.37, 11.38, 11.39, 11.40, 11.41, 11.42, 11.43, 11.44, 11.45, 11.46, 11.47, 11.48, 11.49, 11.50, 11.51, 11.52, 11.53, 11.54, 11.55, 11.56, 11.57, 11.58, 11.59, 11.60, 11.61, 11.62, 11.63, 11.64, 11.65, 11.66, 11.67, 11.68, 11.69, 11.70, 11.71, 11.72, 11.73, 11.74, 11.75, 11.76, 11.77, 11.78, 11.79, 11.80, 11.81, 11.82, 11.83, 11.84, 11.85, 11.86, 11.87, 11.88, 11.89, 11.90, 11.91, 11.92, 11.93, 11.94, 11.95, 11.96, 11.97, 11.98, 11.99, 12.00, 12.01, 12.02, 12.03, 12.04, 12.05, 12.06, 12.07, 12.08, 12.09, 12.10, 12.11, 12.12, 12.13, 12.14, 12.15, 12.16, 12.17, 12.18, 12.19, 12.20, 12.21, 12.22, 12.23, 12.24, 12.25, 12.26, 12.27, 12.28, 12.29, 12.30, 12.31, 12.32, 12.33, 12.34, 12.35, 12.36, 12.37, 12.38, 12.39, 12.40, 12.41, 12.42, 12.43, 12.44, 12.45, 12.46, 12.47, 12.48, 12.49, 12.50, 12.51, 12.52, 12.53, 12.54, 12.55, 12.56, 12.57, 12.58, 12.59, 12.60, 12.61, 12.62, 12.63, 12.64, 12.65, 12.66, 12.67, 12.68, 12.69, 12.70, 12.71, 12.72, 12.73, 12.74, 12.75, 12.76, 12.77, 12.78, 12.79, 12.80, 12.81, 12.82, 12.83, 12.84, 12.85, 12.86, 12.87, 12.88, 12.89, 12.90, 12.91, 12.92, 12.93, 12.94, 12.95, 12.96, 12.97, 12.98, 12.99, 13.00, 13.01, 13.02, 13.03, 13.04, 13.05, 13.06, 13.07, 13.08, 13.09, 13.10, 13.11, 13.12, 13.13, 13.14, 13.15, 13.16, 13.17, 13.18, 13.19, 13.20, 13.21, 13.22, 13.23, 13.24, 13.25, 13.26, 13.27, 13.28, 13.29, 13.30, 13.31, 13.32, 13.33, 13.34, 13.35, 13.36, 13.37, 13.38, 13.39, 13.40, 13.41, 13.42, 13.43, 13.44, 13.45, 13.46, 13.47, 13.48, 13.49, 13.50, 13.51, 13.52, 13.53, 13.54, 13.55, 13.56, 13.57, 13.58, 13.59, 13.60, 13.61, 13.62, 13.63, 13.64, 13.65, 13.66, 13.67, 13.68, 13.69, 13.70, 13.71, 13.72, 13.73, 13.74, 13.75, 13.76, 13.77, 13.78, 13.79, 13.80, 13.81, 13.82, 13.83, 13.84, 13.85, 13.86, 13.87, 13.88, 13.89, 13.90, 13.91, 13.92, 13.93, 13.94, 13.95, 13.96, 13.97, 13.98, 13.99, 14.00, 14.01, 14.02, 14.03, 14.04, 14.05, 14.06, 14.07, 14.08, 14.09, 14.10, 14.11, 14.12, 14.13, 14.14, 14.15, 14.16, 14.17, 14.18, 14.19, 14.20, 14.21, 14.22, 14.23, 14.24, 14.25, 14.26, 14.27, 14.28, 14.29, 14.30, 14.31, 14.32, 14.33, 14.34, 14.35, 14.36, 14.37, 14.38, 14.39, 14.40, 14.41, 14.42, 14.43, 14.44, 14.45, 14.46, 14.47, 14.48, 14.49, 14.50, 14.51, 14.52, 14.53, 14.54, 14.55, 14.56, 14.57, 14.58, 14.59, 14.60, 14.61, 14.62, 14.63, 14.64, 14.65, 14.66, 14.67, 14.68, 14.69, 14.70, 14.71, 14.72, 14.73, 14.74, 14.75, 14.76, 14.77, 14.78, 14.79, 14.80, 14.81, 14.82, 14.83, 14.84, 14.85, 14.86, 14.87, 14.88, 14.89, 14.90, 14.91, 14.92, 14.93, 14.94, 14.95, 14.96, 14.97, 14.98, 14.99, 15.00, 15.01, 15.02, 15.03, 15.04, 15.05, 15.06, 15.07, 15.08, 15.09, 15.10, 15.11, 15.12, 15.13, 15.14, 15.15, 15.16, 15.17, 15.18, 15.19, 15.20, 15.21, 15.22, 15.23, 15.24, 15.25, 15.26, 15.27, 15.28, 15.29, 15.30, 15.31, 15.32, 15.33, 15.34, 15.35, 15.36, 15.37, 15.38, 15.39, 15.40, 15.41, 15.42, 15.43, 15.44, 15.45, 15.46, 15.47, 15.48, 15.49, 15.50, 15.51, 15.52, 15.53, 15.54, 15.55, 15.56, 15.57, 15.58, 15.59, 15.60, 15.61, 15.62, 15.63, 15.64, 15.65, 15.66, 15.67, 15.68, 15.69, 15.70, 15.71, 15.72, 15.73, 15.74, 15.75, 15.76, 15.77, 15.78, 15.79, 15.80, 15.81, 15.82, 15.83, 15.84, 15.85, 15.86, 15.87, 15.88, 15.89, 15.90, 15.91, 15.92, 15.93, 15.94, 15.95, 15.96, 15.97, 15.98, 15.99, 16.00, 16.01, 16.02, 16.03, 16.04, 16.05, 16.06, 16.07, 16.08, 16.09, 16.10, 16.11, 16.12, 16.13, 16.14, 16.15, 16.16, 16.17, 16.18, 16.19, 16.20, 16.21, 16.22, 16.23, 16.24, 16.25, 16.26, 16.27, 16.28, 16.29, 16.30, 16.31, 16.32, 16.33, 16.34, 16.35, 16.36, 16.37, 16.38, 16.39, 16.40, 16.41, 16.42, 16.43, 16.44, 16.45, 16.46, 16.47, 16.48, 16.49, 16.50, 16.51, 16.52, 16.53, 16.54, 16.55, 16.56, 16.57, 16.58, 16.59, 16.60, 16.61, 16.62, 16.63, 16.64, 16.65, 16.66, 16.67, 16.68, 16.69, 16.70, 16.71, 16.72, 16.73, 16.74, 16.75, 16.76, 16.77, 16.78, 16.79, 16.80, 16.81, 16.82, 16.83, 16.84, 16.85, 16.86, 16.87, 16.88, 16.89, 16.90, 16.91, 16.92, 16.93, 16.94, 16.95, 16.96, 16.97, 16.98, 16.99, 17.00, 17.01, 17.02, 17.03, 17.04, 17.05, 17.06, 17.07, 17.08, 17.09, 17.10, 17.11, 17.12, 17.13, 17.14, 17.15, 17.16, 17.17, 17.18, 17.19, 17.20, 17.21, 17.22, 17.23, 17.24, 17.25, 17.26, 17.27, 17.28, 17.29, 17.30, 17.31, 17.32, 17.33, 17.34, 17.35, 17.36, 17.37, 17.38, 17.39, 17.40, 17.41, 17.42, 17.43, 17.44, 17.45, 17.46, 17.47, 17.48, 17.49, 17.50, 17.51, 17.52, 17.53, 17.54, 17.55, 17.56, 17.57, 17.58, 17.59, 17.60, 17.61, 17.62, 17.63, 17.64, 17.65, 17.66, 17.67, 17.68, 17.69, 17.70, 17.71, 17.72, 17.73, 17.74, 17.75, 17.76, 17.77, 17.78, 17.79, 17.80, 17.81, 17.82, 17.83, 17.84, 17.85, 17.86, 17.87, 17.88, 17.89, 17.90, 17.91, 17.92, 17.93, 17.94, 17.95, 17.96, 17.97, 17.98, 17.99, 18.00, 18.01, 18.02, 18.03, 18.04, 18.05, 18.06, 18.07, 18.08, 18.09, 18.10, 18.11, 18.12, 18.13, 18.14, 18.15, 18.16, 18.17, 18.18, 18.19, 18.20, 18.21, 18.22, 18.23, 18.24, 18.25, 18.26, 18.27, 18.28, 18.29, 18.30, 18.31, 18.32, 18.33, 18.34, 18.35, 18.36, 18.37, 18.38, 18.39, 18.40, 18.41, 18.42, 18.43, 18.44, 18.45, 18.46, 18.47, 18.48, 18.49, 18.50, 18.51, 18.52, 18.53, 18.54, 18.55, 18.56, 18.57, 18.58, 18.59, 18.60, 18.61, 18.62, 18.63, 18.64, 18.65, 18.66, 18.67, 18.68, 18.69, 18.70, 18.71, 18.72, 18.73, 18.74, 18.75, 18.76, 18.77, 18.78, 18.79, 18.80, 18.81, 18.82, 18.83, 18.84, 18.85, 18.86, 18.87, 18.88, 18.89, 18.90, 18.91, 18.92, 18.93, 18.94, 18.95, 18.96, 18.97, 18.98, 18.99, 19.00, 19.01, 19.02, 19.03, 19.04, 19.05, 19.06, 19.07, 19.08, 19.09, 19.10, 19.11, 19.12, 19.13, 19.14, 19.15, 19.16, 19.17, 19.18, 19.19, 19.20, 19.21, 19.22, 19.23, 19.24, 19.25, 19.26, 19.27, 19.28, 19.29, 19.30, 19.31, 19.32, 19.33, 19.34, 19.35, 19.36, 19.37, 19.38, 19.39, 19.40, 19.41, 19.42, 19.43, 19.44, 19.45, 19.46, 19.47, 19.48, 19.49, 19.50, 19.51, 19.52, 19.53, 19.54, 19.55, 19.56, 19.57, 19.58, 19.59, 19.60, 19.61, 19.62, 19.63, 19.64, 19.65, 19.66, 19.67, 19.68, 19.69, 19.70, 19.71, 19.72, 19.73, 19.74, 19.75, 19.76, 19.77, 19.78, 19.79, 19.80, 19.81, 19.82, 19.83, 19.84, 19.85, 19.86, 19.87, 19.88, 19.89, 19.90, 19.91, 19.92, 19.93, 19.94, 19.95, 19.96, 19.97, 19.98, 19.99, 20.00, 20.01, 20.02, 20.03, 20.04, 20.05, 20.06, 20.07, 20.08, 20.09, 20.10, 20.11, 20.12, 20.13, 20.14, 20.15, 20.16, 20.17, 20.18, 20.19, 20.20, 20.21, 20.22, 20.23, 20.24, 20.25, 20.26, 20.27, 20.28, 20.29, 20.30, 20.31, 20.32, 20.33, 20.34, 20.35, 20.36, 20.37, 20.38, 20.39, 20.40, 20.41, 20.42, 20.43, 20.44, 20.45, 20.46, 20.47, 20.48, 20.49, 20.50, 20.51, 20.52, 20.53, 20.54, 20.55, 20.56, 20.57, 20.58, 20.59, 20.60, 20.61, 20.62, 20.63, 20.64, 20.65, 20.66, 20.67, 20.68, 20.69, 20.70, 20.71, 20.72, 20.73, 20.74, 20.75, 20.76, 20.77, 20.78, 20.79, 20.80, 20.81, 20.82, 20.83, 20.84, 20.85, 20.86, 20.87, 20.88, 20.89, 20.90, 20.91, 20.92, 20.93, 20.94, 20.95, 20.96, 20.97, 20.98, 20.99, 21.00, 21.01, 21.02, 21.03, 21.04, 21.05, 21.06, 21.07, 21.08, 21.09, 21.10, 21.11, 21.12, 21.13, 21.14, 21.15, 21.16, 21.17, 21.18, 21.19, 21.20, 21.21, 21.22, 21.23, 21.24, 21.25, 21.26, 21.27, 21.28, 21.29, 21.30, 21.31, 21.32, 21.33, 21.34, 21.35, 21.36, 21.37, 21.38, 21.39, 21.40, 21.41, 21.42, 21.43, 21.44, 21.45, 21.46, 21.47, 21.48, 21.49, 21.50, 21.51, 21.52, 21.53, 21.54, 21.55, 21.56, 21.57, 21.58, 21.59, 21.60, 21.61, 21.62, 21.63, 21.64, 21.65, 21.66, 21.67, 21.68, 21.69, 21.70, 21.71, 21.72, 21.73, 21.74, 21.75, 21.76, 21.77, 21.78, 21.79, 21.80, 21.81, 21.82, 21.83, 21.84, 21.85, 21.86, 21.87, 21.88, 21.89, 21.90, 21.91, 21.92, 21.93, 21.94, 21.95, 21.96, 21.97, 21.98, 21.99, 22.00, 22.01, 22.02, 22.03, 22.04, 22.05, 22.06, 22.07, 22.08, 22.09, 22.10, 22.11, 22.12, 22.13, 22.14, 22.15, 22.16, 22.17, 22.18, 22.19, 22.20, 22.21, 22.22, 22.23, 22.24, 22.25, 22.26, 22.27, 22.28, 22.29, 22.30, 22.31, 22.32, 22.33, 22.34, 22.35, 22.36, 22.37, 22.38, 22.39, 22.40, 22.41, 22.42, 22.43, 22.44, 22.45, 22.46, 22.47, 22.48, 22.49, 22.50, 22.51, 22.52, 22.53, 22.54, 22.55, 22.56, 22.57, 22.58, 22.59, 22.60, 22.61, 22.62, 22.63, 22.64, 22.65, 22.66, 22.67, 22.68, 22.69, 22.70, 22.71, 22.72, 22.73, 22.74, 22.75, 22.76, 22.77, 22.78, 22.79, 22.80, 22.81, 22.82, 22.83, 22.84, 22.85, 22.86, 22.87, 22.88, 22.89, 22.90, 22.91, 22.92, 22.93, 2

Architecture: OpenQA client: hardware (LAVA)



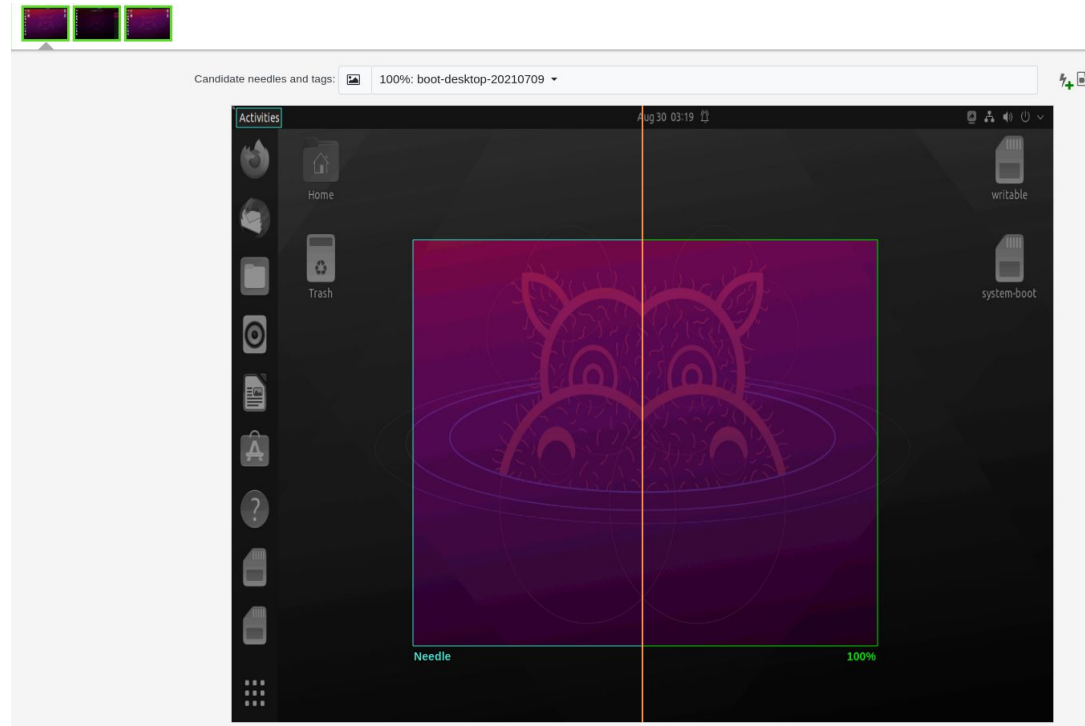
LAVA and OpenQA

- Testing on hardware rather than emulators
- Utilise the existing KernelCI tests
- OpenQA testing to check for regressions (e.g in the graphics driver)
- Check out the [blog post](#) for details of the setup

It's up and running!

- <https://lava.qa.codethink.co.uk>
- <https://openqa.qa.codethink.co.uk/>

It's being used!



It's being used!

- Positive [feedback received](#):

On Mon, Aug 30, 2021 at 2:12 AM Sudip Mukherjee
<sudipm.mukherjee@gmail.com> wrote:

```
>  
> We were recently working on openqa based testing and is a very basic  
> testing for now.. Build the kernel for x86_64 and arm64, boot it on  
> qemu and rpi4 and test that the desktop environment is working. And,  
> it now tests mainline branch every night. So, last night it tested  
> "5.14.0-7d2a07b76933" and both tests were ok.
```

Thanks. The more the merrier, and if you do this every night, having a fairly low-latency "it stopped working" will be good.

Of course, if you can find some other slightly more oddball configuration that you would also like to test, that it would be even better.

Because while it's lovely to have more automated testing, if everybody only tests x86-64 and arm64, the less common cases get little to no testing.

No big deal, but I thought I'd just mention it in case you go "Yeah, I know XYZ is entirely irrelevant, but I happen to like it, so I could easily add that to the testing too".

Linus

Next steps....

We wanted to be able to test on representative hardware, not just dev-boards....

QAD and testing on hardware



What is Q.A.D.?

- [QAD](#) is a brand new, open-source, Linux based service
- Runs on real rigs and production hardware
- Minimally invasive integration
- Can be thought of as an automatable test engineer



What can Q.A.D. do?

- Access and capture all displays attached to the system
- Send input events to the system
 - Touch events, on any screen in any location
 - Swipe events, on any screen, with configurable velocity
- Provide access to all of that via an HTTP(s) API



Q.A.D. as a remote access service

- Interact with rigs over the network
 - Permits rigs to be stored away from engineer desks
 - Allows engineers to share access to limited hardware resources
- With VPNs, engineers can also access rigs, or even vehicles, remotely to permit diagnosis and testing to happen more efficiently.
- With minimal dependencies and a low-invasiveness design, Q.A.D. can be integrated into development and even production builds of your software.



What Q.A.D. is not

- Q.A.D. only provides display capture and input event injection.
 - You should use SSH or similar for remote shell access
- Q.A.D. does not automate anything itself
 - Use of rig management and test orchestration frameworks will be necessary
- Not a silver bullet for your testing woes, but it will help

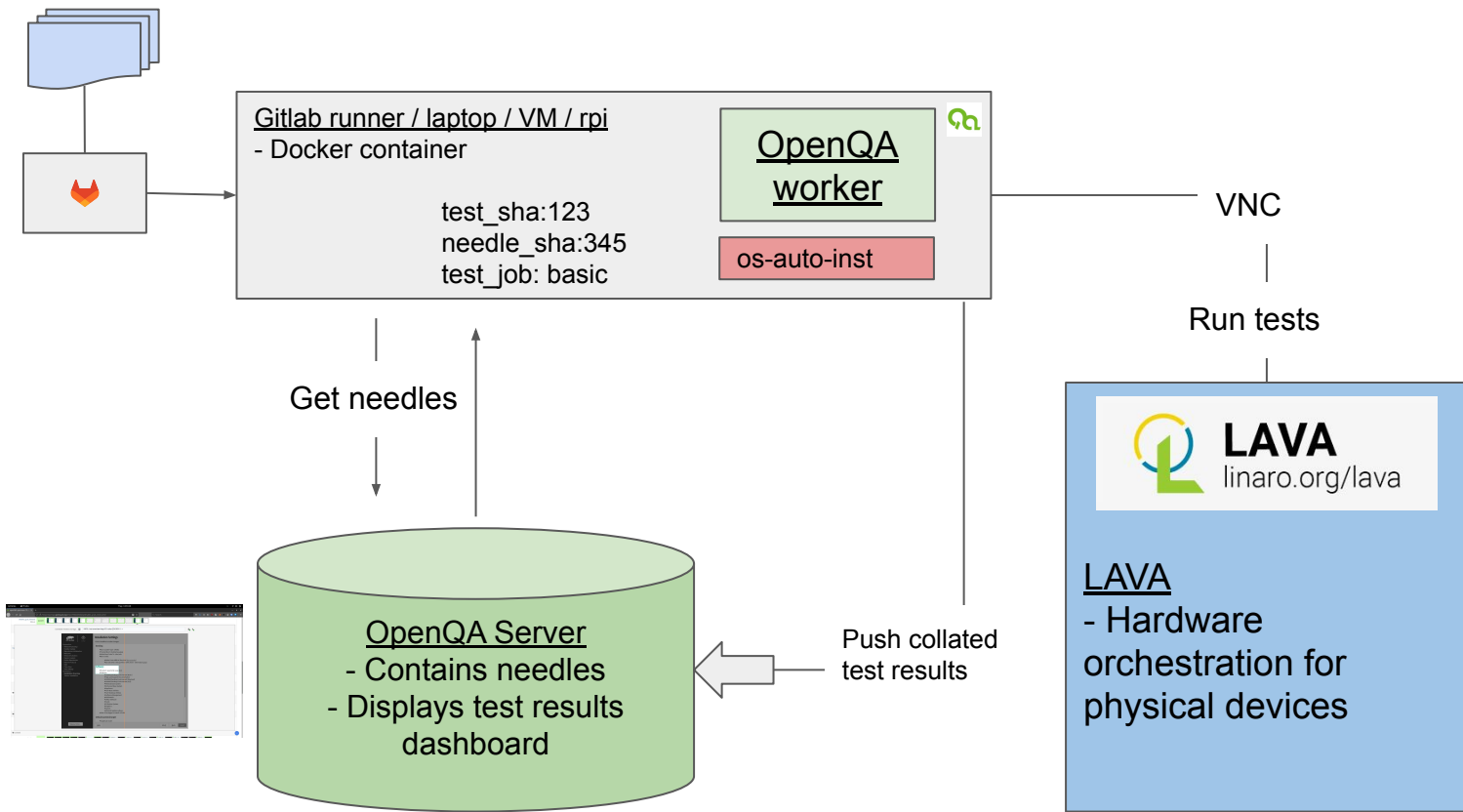


Q.A.D. in automation

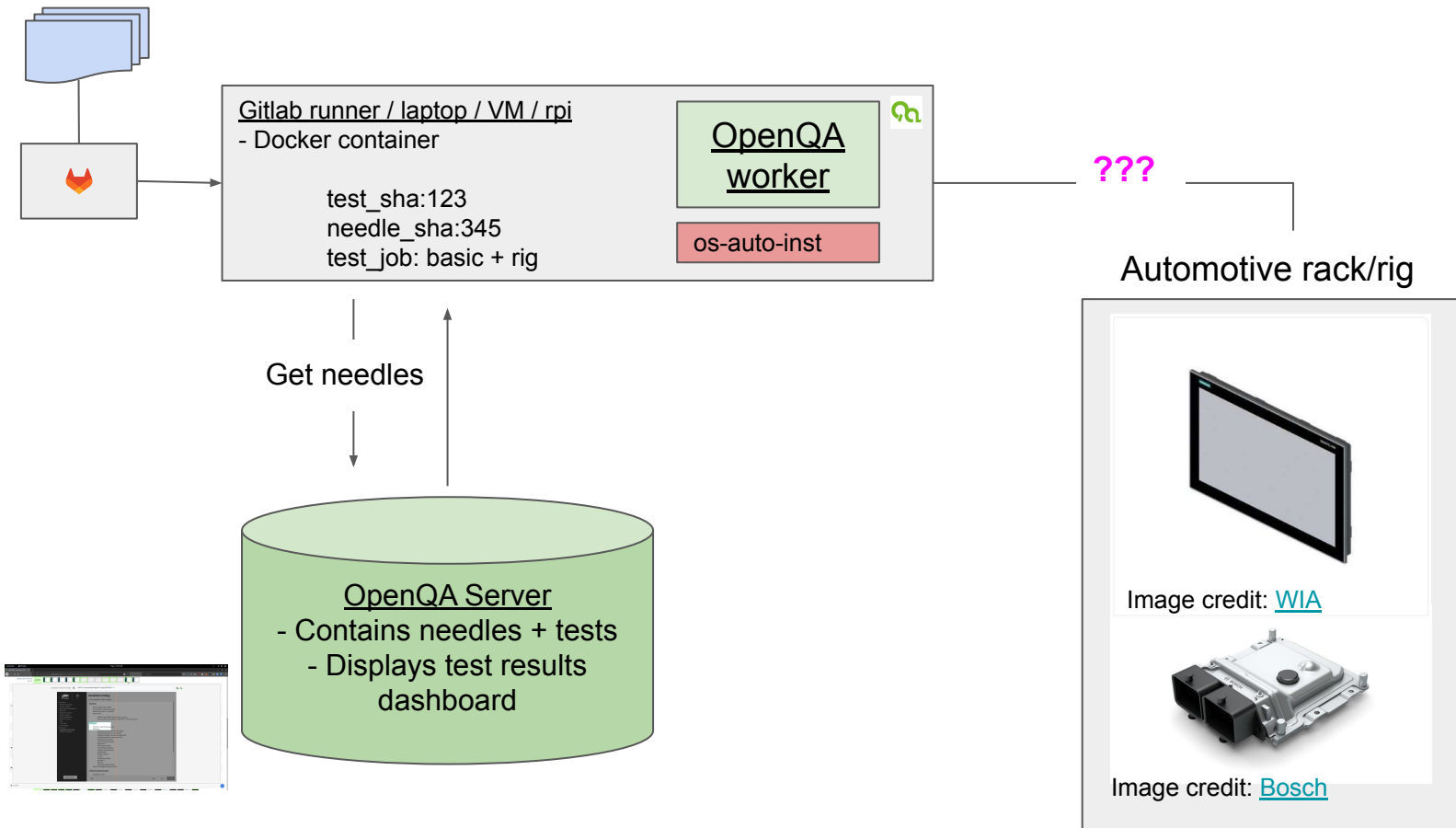
- Simple HTTP(s)/JSON API for input events
- Simple screenshot-over-HTTP(s) API
- Effectively one of OpenQA's testing methodologies encapsulated
- Integration with OpenQA provided in Q.A.D. repository



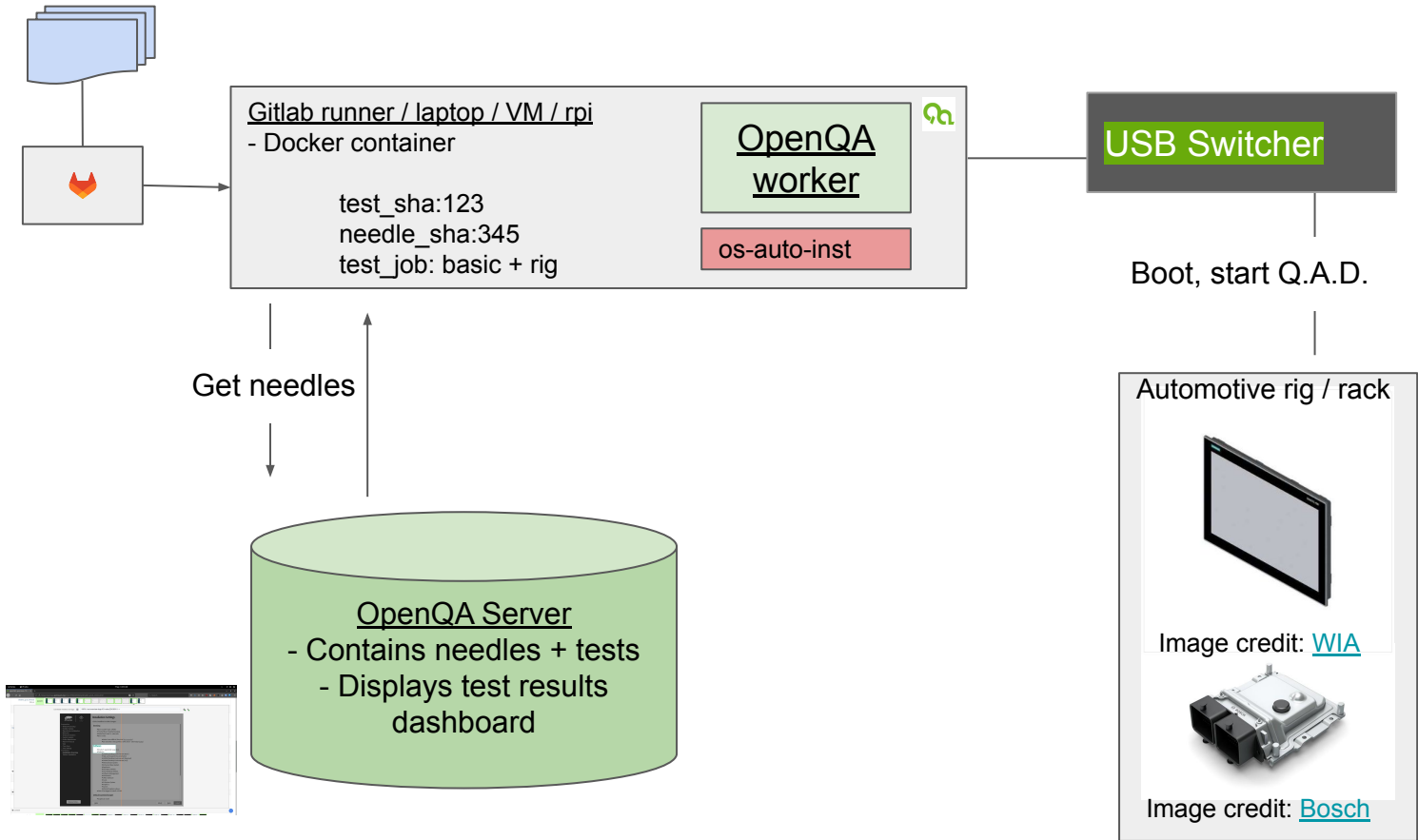
Architecture: OpenQA client: hardware (LAVA)



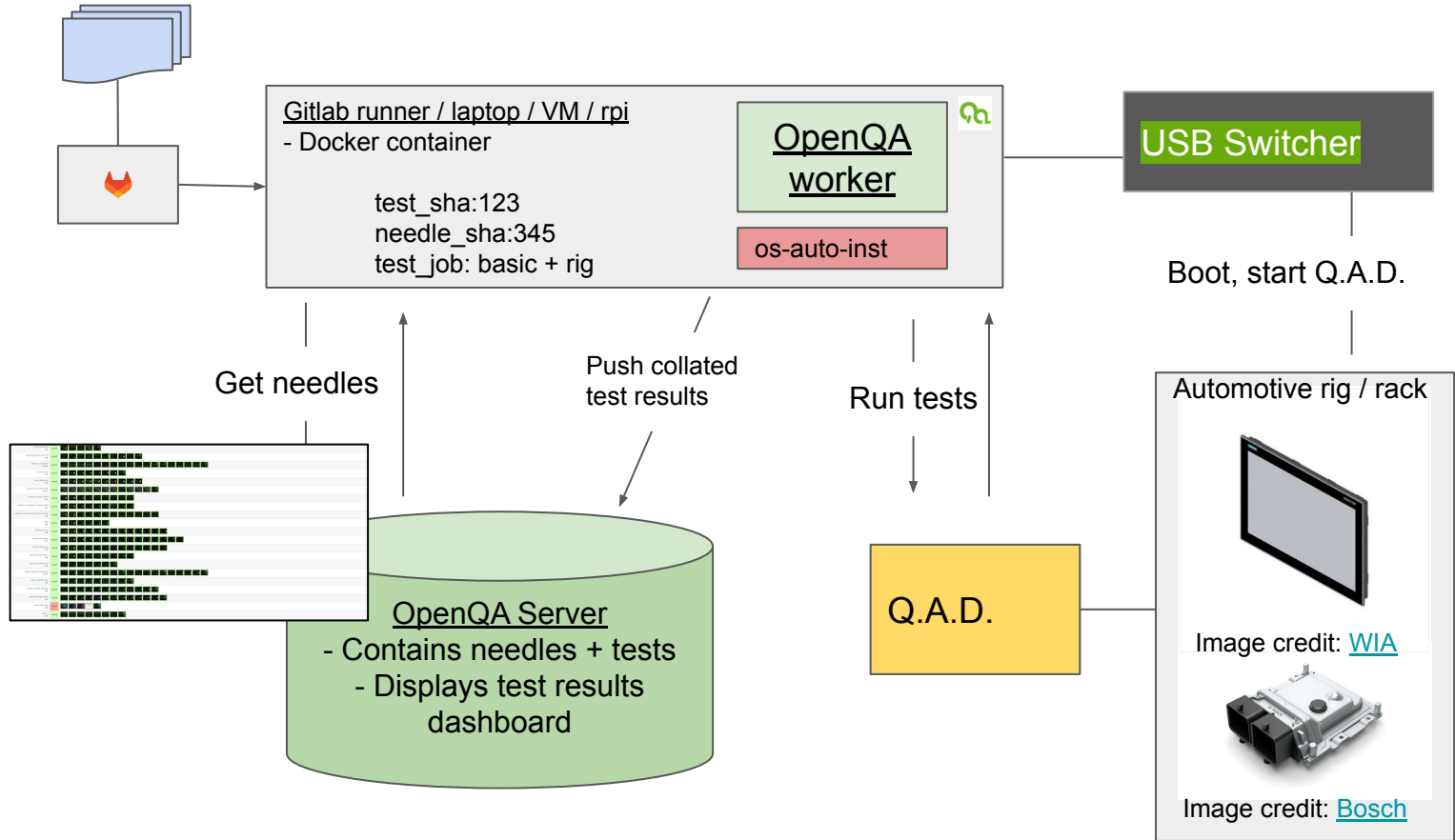
Architecture: OpenQA client: hardware

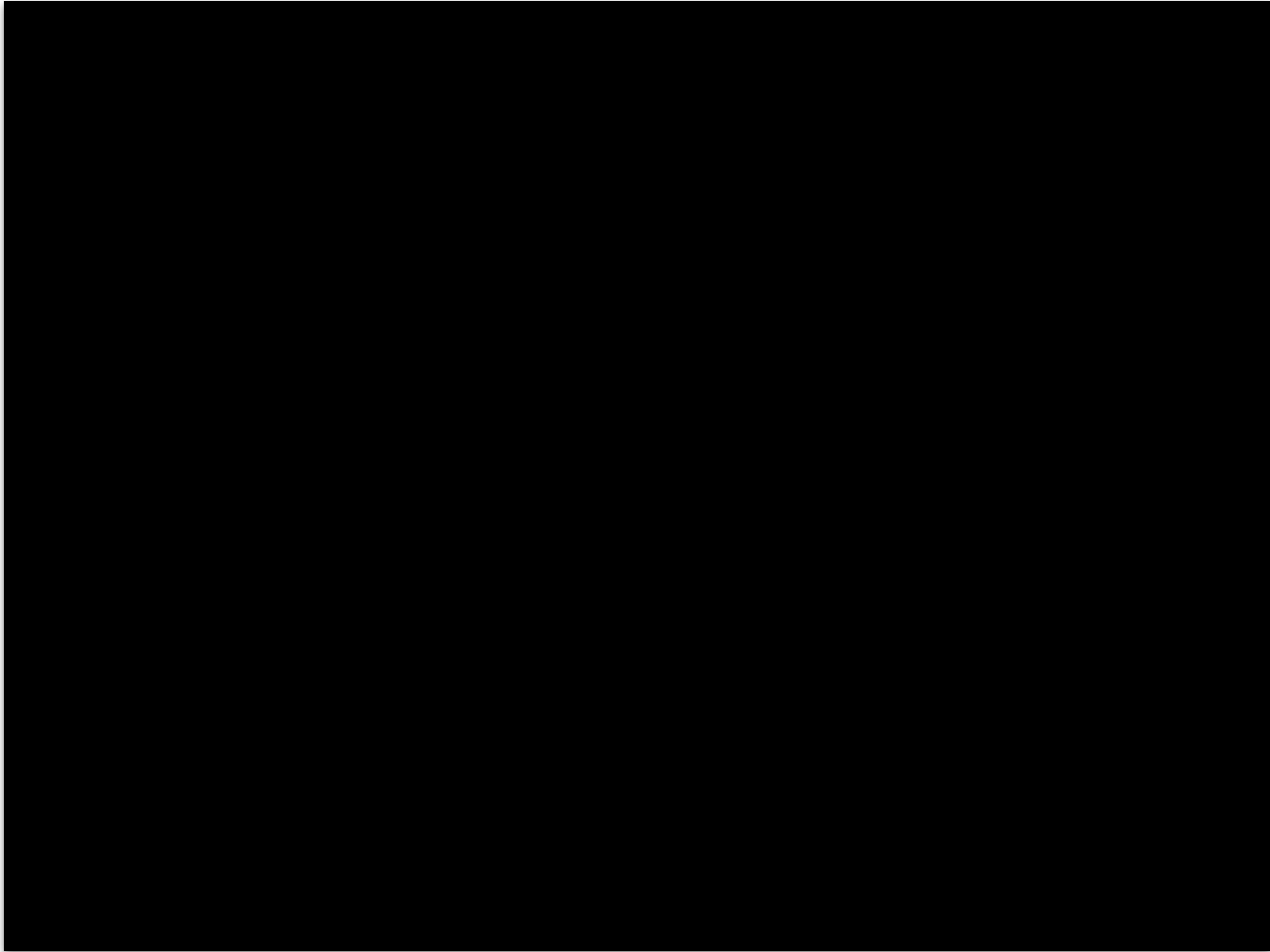


Architecture: OpenQA client: hardware








Architecture: OpenQA client: hardware





Building OpenQA needles with Q.A.D.

- Standard OpenQA workflows are supported
- We can build needles in the standard OpenQA manner
- But could be a better way

Details	Logs & Assets	Settings	Comments (0)	Next & previous results
Test	Result	References		
tests				
home 1m 2s  	passed			

Q.A.D. plus Web-UI

- In development web-ui to enable remote-rig support to be more pleasant
- Provides ability to see all screens attached to the device which Q.A.D. can capture

The screenshot displays the Q.A.D. plus Web-UI interface. On the left, a 'Needles' panel includes a 'Pick Needle Directory' button and a 'Filter by name' input field. Below these, a list shows 'agl-home' and 'agl-lvacc'. The main area on the right shows a device screen capture with a dark theme and various icons for 'DASHBOARD', 'HVAC', 'MEDIAPLAYER', 'MESSAGING', and 'MIXER'. Above the capture, a toolbar contains buttons for 'Refresh Screen', 'Download JSON to file', 'Download current screen', 'Toggle clickpoint', 'Toggle Image (needle)', 'Remove current area', 'Toggle needle areas', and 'Automatic Reload'. Below the capture, a 'Pick Tests Directory' button is visible. At the bottom, there are 'Update JSON' and 'Update PNG' buttons.

Needles in home test

```
{
  "area": {
    "xpos": 16,
    "ypos": 10,
    "width": 358,
    "height": 197,
    "type": "match",
    "match": 50
  },
  "xpos": 742,
  "ypos": 281,
  "width": 291,
  "height": 194,
  "type": "match",
  "match": 50,
  "click point": {
    "xpos": 9536,
    "ypos": 11936
  }
}
```

- Users can interact with the displays as though they are touchscreens
- The UI can help to streamline the creation of OpenQA needles
- Not 100% necessary, but certainly useful

Q.A.D. - Planned future work

- Ongoing open source project, with many plans for future improvements
- Web-UI to be made 'production ready'
- Support for other screen capture backends
- Support for other input devices and other modes of input

Q.A.D is under the MIT licence and we would love to see some of you engaging with us to help make it meet your use-cases.



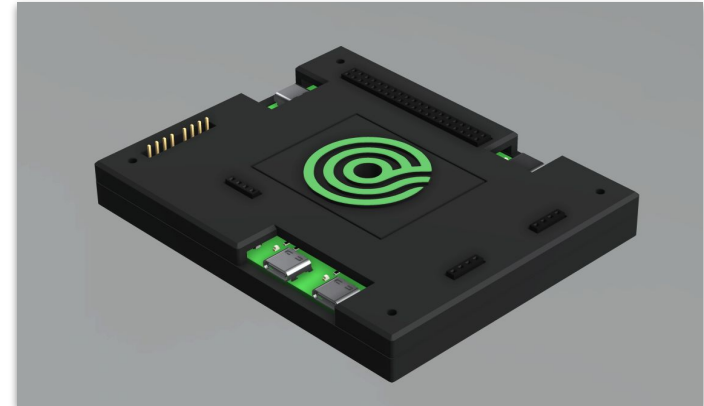
Taking test automation further...

- Write more tests to find more places to automate
- Provide a mechanism for automating CAN interactions to replace CANalyzer and the like...

```
[
{
  "send": "IGN_ON",
  "send_start": true,
  "send_period": 100,
  "signal_map": [
    {
      "send": "IGN_Stat",
      "default": "I_IGN_ON"
    },
    {
      "send": "RevGr_Engg",
      "default": "I_REV_OFF"
    }
  ]
},
{
  "send": "HVAC_SEND...",
  "receive": "HVAC_RECEIVE...",
  "send_start": true,
  "send_period": 1000,
  "signal_map": [
    {
      "send": "HVAC_FAN_ZONE_1...",
      "recv": "HVAC_FAN_ZONE_1_REC...",
      "default": 1.0
    },
    {
      "send": "FOO",
      "recv": "BAR",
      "default": 1.0
    }
  ],
  ....
}
```

USB Switcher - Further automation

- Permits creating automation flows involving USB devices
 - Writing new firmware to a rig via USB
 - Testing USB media
 - Validating screen projection by means of plugging in an iPhone or Android Device
- Can keep devices powered when not connected to keep phones alive



USB Switcher - Fully Open Testing

- Bidirectional USB-C switch with computer control
- Can connect two devices to one host
- Can connect two hosts to one device
- Open Source hardware, firmware, control software, and case.
- Suitable for use in automating more than just rig tests.



Testing in a Box



Testing in a Box

Current hardware testing

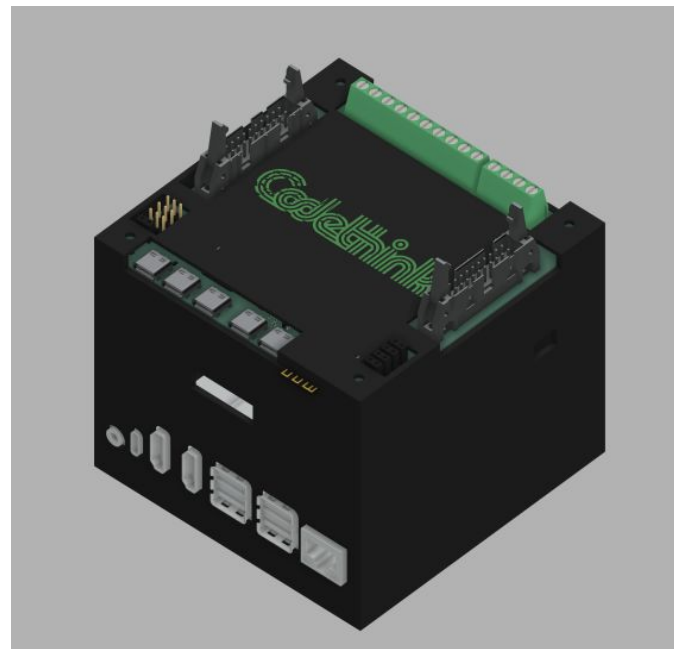
- Manual set up of infrastructure for hardware and software
- Takes time and effort to get up and running
- Messy set up (cables and peripherals all over the place)
- Access to client CI&T setup



The solution so far...

Hardware:

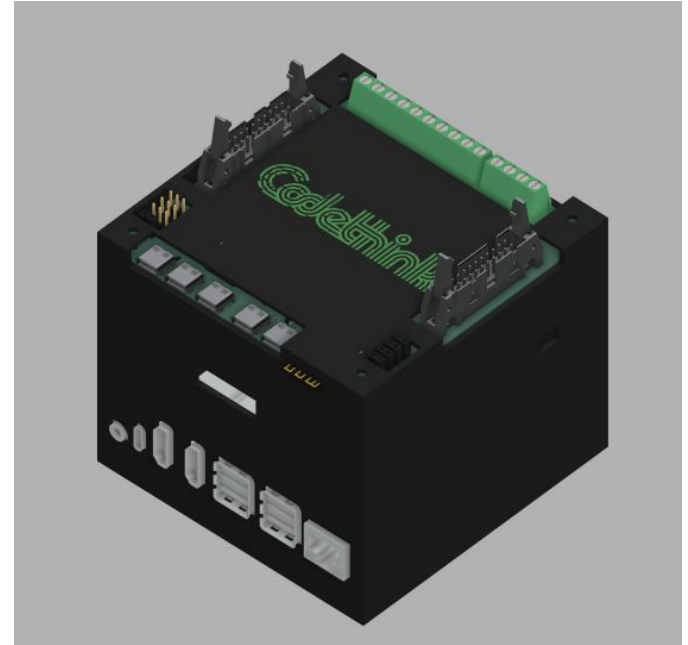
- Serial I/O
- CAN bus emulator
- USB Switch
- USB Hub
- Bluetooth/WiFi
- I2C
- SPI
- HID emulation
- Host PC



The solution so far...

Software (Running on the host):

- Gitlab CI
- Gitlab Runners
- QAD and OpenQA



Coming next

Future plans

- CI templates
- Version 2 for TIAB board
- Much more to come

It's open source so get involved!

Testing in a Box group:

<https://gitlab.com/CodethinkLabs/testing-in-a-box>



And finally...



Summary

Testing is expensive, and often a bottleneck with hidden costs:

- Automating that testing is hard
- With the right tooling it can be easier

With open-source tooling, it can be more cost-effective as well

Codethink is always happy to talk about this, wherever you may be on your testing journey.

Any questions?

Laurence Urhegyi <laurence.urhegyi@codethink.co.uk>

James Thomas <james.thomas@codethink.co.uk>



www.codethink.co.uk