



Shared automated testing in GENIVI

on QEMU and automotive hardware

Stephen Lawrence (Renesas), GENIVI BIT Lead | 2 May 2019



Topic today

- Japanese proverb: さるもきからおちる (Saru mo ki kara ochiru)
- Even monkeys fall out of trees
- GENIVI exists for the industry to reduce costs and accelerate time to market by collaborating on non-differentiating technology
- Putting the two together
 - Shared testing to support **development**
 - Automated where possible and
 - on QEMU and automotive hardware

Industry Trends

Shared testing upstream

- Shared testing of important software
 - Successful example is [Kernel CI](#)
 - Community based, open source distributed test automation system focused on upstream Linux kernel development.
 - Goals
 - Build every configuration for each architecture.
 - Boot these configurations.
 - Execute tests on these configurations.
- Collaboration on test cases, interoperability and mechanisms
 - [Automated Testing Summit](#)

Automotive needs



- Companies have advanced internal testing setups
 - proprietary s/w which can't be shared and
 - OSS s/w which is ← here shared testing is possible
- OSS included in company-internal testing anyway so why share?
 - Pooling resources creates ability to test wider variation of versions and configuration than is normally done in a production or internal platform project (remember Kernel CI)
 - We use component Foo v4. Should we take v5?
 - v5 may not be tested internally (yet), but in wider community it might be
 - Investigating upstream components for integration
- Conclusion: ability to look upstream for test results is a stronger basis for development

GENIVI Contribution



What's already in place

- Distributed CI of “systems” using [GENIVI GoCD instance](#)
 - Builds GDP and Baseline
 - Central server, with remote build agents
 - Ability for companies to contribute agents to expand capacity
- [GENIVI CI Policy](#) encourages use of GitHub-integrated tools such as Travis-CI where teams select their own tooling for components
- GENIVI source hosted in [GitHub](#)
 - Integrated with GoCD to sanity build test pull requests for GDP and baseline
- [GDP](#)
 - Sanity tests using ptest
- [Yocto Baseline \(meta-ivi\)](#)
 - Meta-ivi-test image contains component unit tests
- Components
 - Largely tested in individual companies seemingly

GENIVI s/w scope is expanding

- Multi-OS strategy
 - Domain Interaction evolving into Multi-OS
 - Android SiG, safety OSs
- Meaning multiple development environments
- Collective opportunity
 - Possibility to collaborate and integrate with existing testing infrastructures for supported development platforms such as Apertis, WebOS and Android.
 - Open dialog and flexibility
 - Favour working together towards greater combined solutions, than repetition
 - Example
 - Discussions about testing of [Adaptive Autosar ARA::COM and Franca IDL tooling](#)
 - Mix of internal models which can't be shared and hopefully some that can.

New automated test initiative

- We would like your help
- Working mode: make a start, be flexible and open to collaboration with other orgs
- [LAVA](#) based test system to be connected to Genivi GoCD CI (and other CI as needed)
 - Distributed system in wide use.
 - Example, embedded industrial [Civil Infrastructure Platform \(CiP\)](#)
 - LAVA Master (control server)
 - LAVA Worker (execute tests on boards) for QEMU and automotive hardware
- Will use it for CIAT test of future GENIVI code emerging from Multi-OS

How can we all contribute?

- Renesas will contribute a LAVA Worker to provide an R-Car board farm.
- Renesas will also contribute to the setup of the LAVA Master (control server) and LAVA Worker for QEMU.
- Integrate your next GENIVI collaboration into the GENIVI CIAT
 - Easier done from the start, than later of course
- Propose to us other integration opportunities with your existing testing infrastructure
 - Internal (externalise test result, what can be shared upstream?)
 - Related alliances
- Contribute LAVA Worker board farm
- Contribute GoCD agents to increase build capacity
- Contribute test cases or help with integration

Thank you!

Visit GENIVI:

<http://www.genivi.org>

<http://projects.genivi.org>

Contact us:

help@genivi.org

