



# Building and testing an automotive platform - how Automotive Grade Linux is built and tested

*Embedded Linux Conference Europe 2016*

*Jan-Simon Möller*

*Release Manager, [AGL](#) , [The Linux Foundation](#)*

*( [jsmoeller@linuxfoundation.org](mailto:jsmoeller@linuxfoundation.org), DL9PF @IRC and elsewhere )*

# AGL - what ?

- Automotive Grade Linux is a Linux Distribution
- It is based on the Yocto Project/Openembedded
- Platform for multiple device profiles (IVI, telematics, ... )

# AGL - what ?

- Open Source and Code First
- Multiple Architectures:
  - x86 (e.g. Intel Minnowboard)
  - ARM 32 (e.g. Renesas Porter, TI Vayu, RaspberryPI)
  - ARM 64 (e.g. Qualcomm dragonboard 410c, rpi3?)

# This talk ...

- why ? ... we do this
- what ? ... tools we use
- how ? ... we combine them
- what ? ... we want to achieve



Why ?  
... we do this

# Why ?

AGL development ...

- AGL development is done in a distributed way
- Developers around the globe contribute
- Code review

# Why ?

- Does it build ?
- Does it work ?
  - on board/arch A ?
  - on board/arch B ?
  - on board/arch C ?
  - ...
  - on board/arch <n> ?



# Why ?

- This must be a common problem!
  - just see how many talks during ELCE we have ;)
  - multiple solutions – good!
  - different use-cases
- Here is what we use! (Well, we think it does the trick – ideas/feedback welcome! Curious to hear your ideas !)





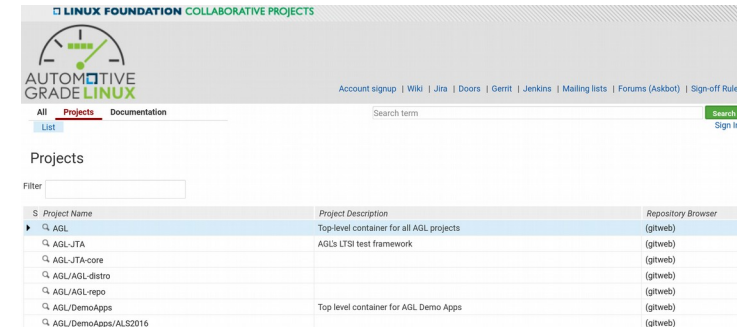
What ?  
... tools we use

# What ? ... tools we use

- SCM + Review: Gerrit (sorry, Greg)
- CI Builds: Jenkins
- Tests on HW: a) AGL-JTA (Fuego)  
b) LAVA
- Data-Postproc ... ???

# SCM/Code-Review - Gerrit

- <https://git.automotivelinux.org>
- AGL-related projects in AGL/\*
- if we are upstream → /src/\*
- to try out code → /staging/\*



- we use "repo" to pull down the git repositories

# SCM/Code-Review - Gerrit

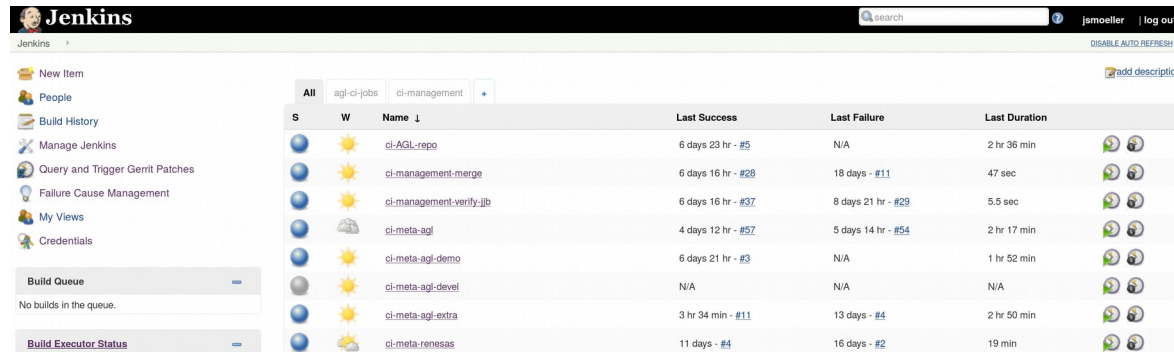
- All code that goes into AGL/\* needs to work on all **reference** and **community** platforms

→ Test matrix:

- Renesas Porter
- Intel Minnowboard
- Qemux86-64 (emulator)
- DragonBoard
- TI Vayu
- NXP Wandboard, Sabre
- RPI 2/3
- ...

# CI Builds - Jenkins

- <https://jenkins-new.automotivelinux.org>
  - Standard Jenkins
    - + gerrit-trigger plugin (to poll git.automotivelinux.org)
    - + openstack cloud plugin (to start jenkins slaves/minions)
      - + slaves run off identical base-images
  - CI-jobs created with Jenkins-Job-Builder (yaml)



The screenshot shows the Jenkins web interface. At the top, there's a header with the Jenkins logo, a search bar, and user information (jmoeller | log out). Below the header, there's a sidebar with navigation links: New Item, People, Build History, Manage Jenkins, Query and Trigger Gerrit Patches, Failure Cause Management, My Views, and Credentials. The main content area displays a table of jobs under the 'agf-ci-jobs' tab. The table has columns for Status (S), Weather icon (W), Name (L), Last Success, Last Failure, Last Duration, and icons for build logs and other actions. The jobs listed are: ci-AGL-repo, ci-management-merge, ci-management-verify-jib, ci-meta-agf, ci-meta-agf-demo, ci-meta-agf-devel, ci-meta-agf-extra, and ci-meta.renasas.

S	W	Name L	Last Success	Last Failure	Last Duration	
●	☀	ci-AGL-repo	6 days 23 hr - #5	N/A	2 hr 36 min	🔍 📄
●	☀	ci-management-merge	6 days 16 hr - #28	18 days - #11	47 sec	🔍 📄
●	☀	ci-management-verify-jib	6 days 16 hr - #37	8 days 21 hr - #29	5.5 sec	🔍 📄
●	☁	ci-meta-agf	4 days 12 hr - #57	5 days 14 hr - #54	2 hr 17 min	🔍 📄
●	☀	ci-meta-agf-demo	6 days 21 hr - #3	N/A	1 hr 52 min	🔍 📄
●	☀	ci-meta-agf-devel	N/A	N/A	N/A	🔍 📄
●	☀	ci-meta-agf-extra	3 hr 34 min - #11	13 days - #4	2 hr 50 min	🔍 📄
●	☁	ci-meta.renasas	11 days - #4	16 days - #2	19 min	🔍 📄

# CI Builds - Jenkins

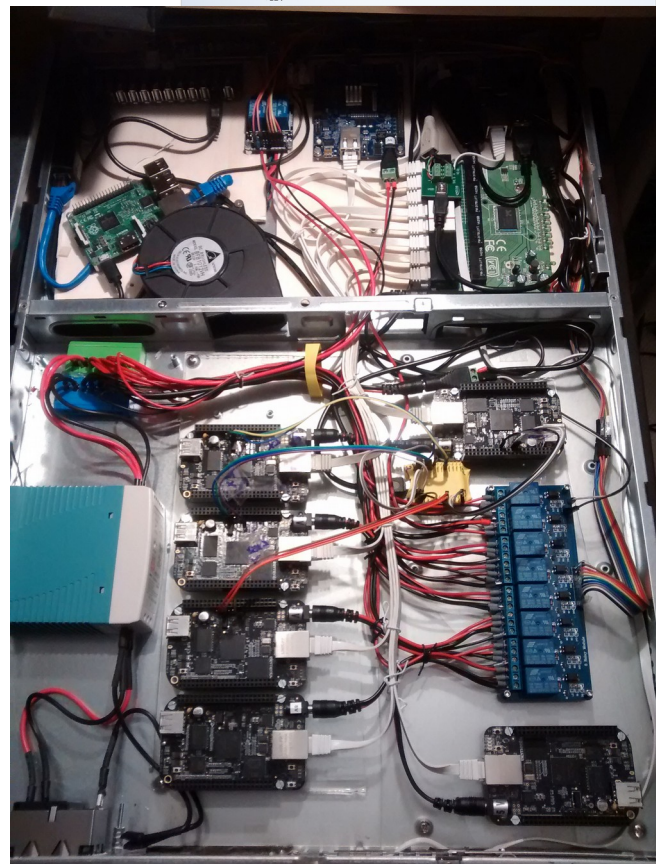
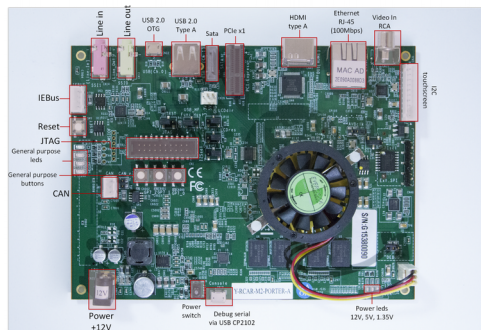
- A successful build will vote "Verified +1"
  - A failed build will vote "Verified -1"
- in Gerrit
- You need to define the success/failure criteria
    - Starts with "it builds" (yay!)
    - Ends with "it boots, runs, passes all tests, updates cleanly, communicates with X and shuts-down properly"

# Tests on HW

Tests on HW are hard !?

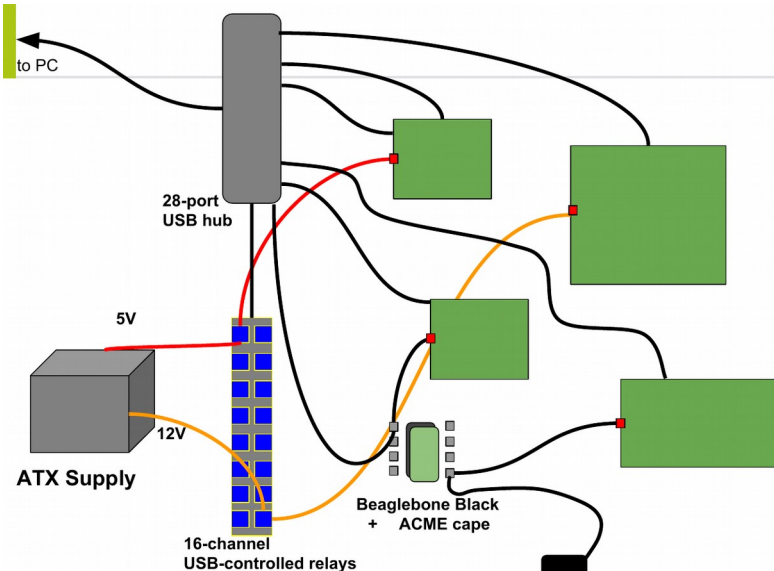
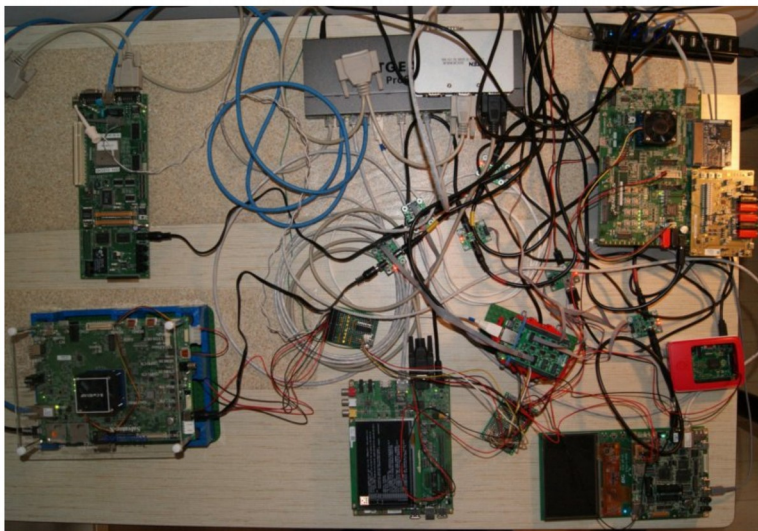
- You need the HW
- You need it on your desk/in your lab
- You need to deploy firmware/filesystems
- You need to reboot the board
- You need to initiate the test
- You need to collect the results
- You need to interpret the results

... rinse & repeat



# Lab setups ... (from ELCE slides)

Board Farm Bird View



Linaro





# Documenting our Lab setup

- WIP document for LAVA:

<http://bit.ly/lavasetup>

- Doc for AGL-JTA/Fuego:











<https://git.automotivelinux.org/gerrit/gitweb?p=AGL-JTA.git;a=tree;f=docs>

- Wiki page in AGL wiki:

- <https://wiki.automotivelinux.org/agl-testframework>

# Tests and frameworks ...


- AGL-JTA (modified/patched Fuego)
  - <https://git.automotivelinux.org/gerrit/gitweb?p=AGL-JTA.git>
  - [jta.automotivelinux.org](https://jta.automotivelinux.org) (Live instance - WIP)
  - runs tests on target boards and collects results
    - results end up right now in a git repo
- ☺ large set of Tests, postproc capabilities
- ☹ Installation, modification, board local (pwr/ssh)

-  [New Item](#)
-  [People](#)
-  [Build History](#)
-  [Edit View](#)
-  [Manage Jenkins](#)
-  [Query and Trigger Gerrit Patches](#)
-  [My Views](#)
-  [Scriptler](#)
-  [Exclusion administration](#)
-  [Credentials](#)

**Build Queue**


No builds in the queue.

**Build Executor Status**


 **master**

1 Idle


2 Idle

 **lager**


1 Idle

 **lager2**

1 Idle

 **porter**

1 Idle

 **qemu-test-arm**

1 Idle

# Test Automation Framework

0. History

AGL CIAT

Additional

Benchmarks

Functional









































all

batch runs

+

add description

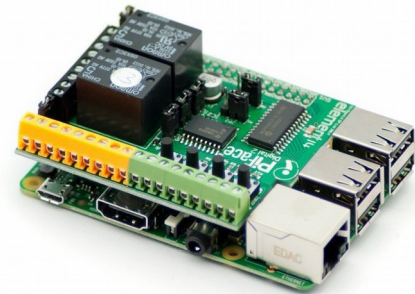
Latest tests runs

Job	Build	Time
 <a href="#">CIAT.java_done</a>	 #6	Oct 5, 2016 7:51:17 PM
 <a href="#">CIAT.upload_result</a>	 #13	Oct 5, 2016 7:51:17 PM
 <a href="#">CIAT.porter1</a>	 #8	Oct 5, 2016 7:51:06 PM
 <a href="#">CIAT.common1</a>	 #8	Oct 5, 2016 7:50:57 PM
 <a href="#">CIAT.porter2</a>	 #8	Oct 5, 2016 7:50:47 PM
 <a href="#">CIAT.renesas-porter</a>	 #8	Oct 5, 2016 7:50:47 PM
 <a href="#">CIAT.java_deploy</a>	 #8	Oct 5, 2016 7:40:04 PM
 <a href="#">CIAT.test</a>	 #8	Oct 5, 2016 7:40:04 PM
 <a href="#">CIAT.java_done</a>	 #5	Oct 5, 2016 4:21:34 PM
 <a href="#">CIAT.upload_result</a>	 #12	Oct 5, 2016 4:21:34 PM
 <a href="#">CIAT.porter1</a>	 #7	Oct 5, 2016 4:21:21 PM
 <a href="#">CIAT.common1</a>	 #7	Oct 5, 2016 4:21:12 PM
 <a href="#">CIAT.porter2</a>	 #7	Oct 5, 2016 4:21:02 PM
 <a href="#">CIAT.renesas-porter</a>	 #7	Oct 5, 2016 4:21:02 PM
 <a href="#">CIAT.java_deploy</a>	 #7	Oct 5, 2016 4:10:38 PM
 <a href="#">CIAT.test</a>	 #7	Oct 5, 2016 4:10:38 PM
 <a href="#">CIAT.java_done</a>	 #4	Oct 5, 2016 2:48:32 PM
 <a href="#">CIAT.upload_result</a>	 #11	Oct 5, 2016 2:48:32 PM
 <a href="#">CIAT.porter1</a>	 #6	Oct 5, 2016 2:48:21 PM
 <a href="#">CIAT.common1</a>	 #6	Oct 5, 2016 2:48:12 PM

Test Run statistics

# Tests and frameworks ...

- LAVA
  - <https://validation.linaro.org>
  - <https://porter.automotivelinux.org/scheduler/alljobs>
  - board farm management + test executor
  - grabs board from pool, pwr & boot & test
    - ☺ multiple boards per type, remote lab (WIP),
    - ☺ runs even on RPI2/3 with PiFACE ! (2 DUT)
    - ☹ first setup little hard, doc for satellite labs



# All Jobs

All Jobs
Active Jobs
Queued Jobs
Pipeline Jobs
Healthcheck Jobs

Show
25
entries
Search

ID ↕	Status ↕	Priority ↕	Device ↕	Description ↕	Submitter ↕	Submit Time ↕	End Time ↕	Duration
163	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 5, 2016, 7:40 p.m.	Oct. 5, 2016, 7:51 p.m.	0:10:58.318669
162	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 5, 2016, 4:10 p.m.	Oct. 5, 2016, 4:21 p.m.	0:10:45.334469
161	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 5, 2016, 2:37 p.m.	Oct. 5, 2016, 2:48 p.m.	0:10:44.438036
160	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 5, 2016, 1:40 p.m.	Oct. 5, 2016, 1:51 p.m.	0:11:08.490490
159	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 5, 2016, 8:44 a.m.	Oct. 5, 2016, 8:55 a.m.	0:11:00.376715
158	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 9:32 p.m.	Oct. 4, 2016, 9:44 p.m.	0:10:56.294506
157	Canceled	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 9:11 p.m.	Oct. 4, 2016, 9:33 p.m.	0:18:15.838088
156	Canceled	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 8:50 p.m.	Oct. 4, 2016, 9:15 p.m.	0:24:15.917332
155	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 4:08 p.m.	Oct. 4, 2016, 4:19 p.m.	0:10:52.630930
154	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 1:13 p.m.	Oct. 4, 2016, 1:24 p.m.	0:10:46.719346
153	Complete	Medium	porter-01	lava-boot job for jta-agl	jta-agl	Oct. 4, 2016, 12:04 p.m.	Oct. 4, 2016, 12:15 p.m.	0:10:55.134659

# Documenting our Lab setup

- WIP document for LAVA:

<http://bit.ly/lavasetup>

- Doc for AGL-JTA/Fuego:

<https://git.automotivelinux.org/gerrit/gitweb?p=AGL-JTA.git;a=tree;f=docs>

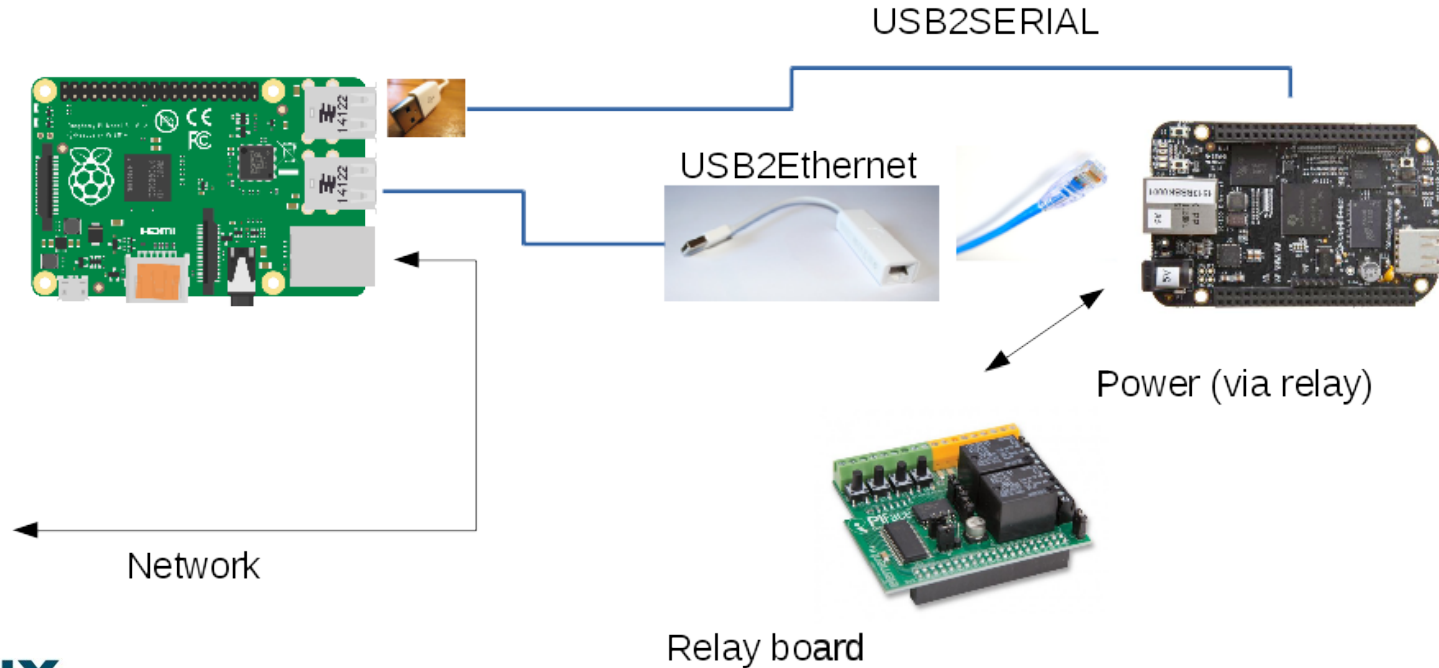
- Wiki page in AGL wiki:

- <https://wiki.automotivelinux.org/agl-testframework>

# BOM <= 100 €

HOST  
(Lava, tftp, ser2net)

DUT



# Data-Postproc ... ???

- Investigating
    - In fuego (AGL-JTA) ??
    - Other mechanism ??
    - What data to track at all ??
- You need to define your key indicators



# Feedback to Developers

- In our case – right in Gerrit:

**Code  
Review**  
(human)



**Verified**  
(CI complete)

**CI-Image-Build**  
("It builds")

**CI-Image-Boot-Test**  
("It boots on HW")

**CI-Image-LTSI-Test**  
("The tests pass")

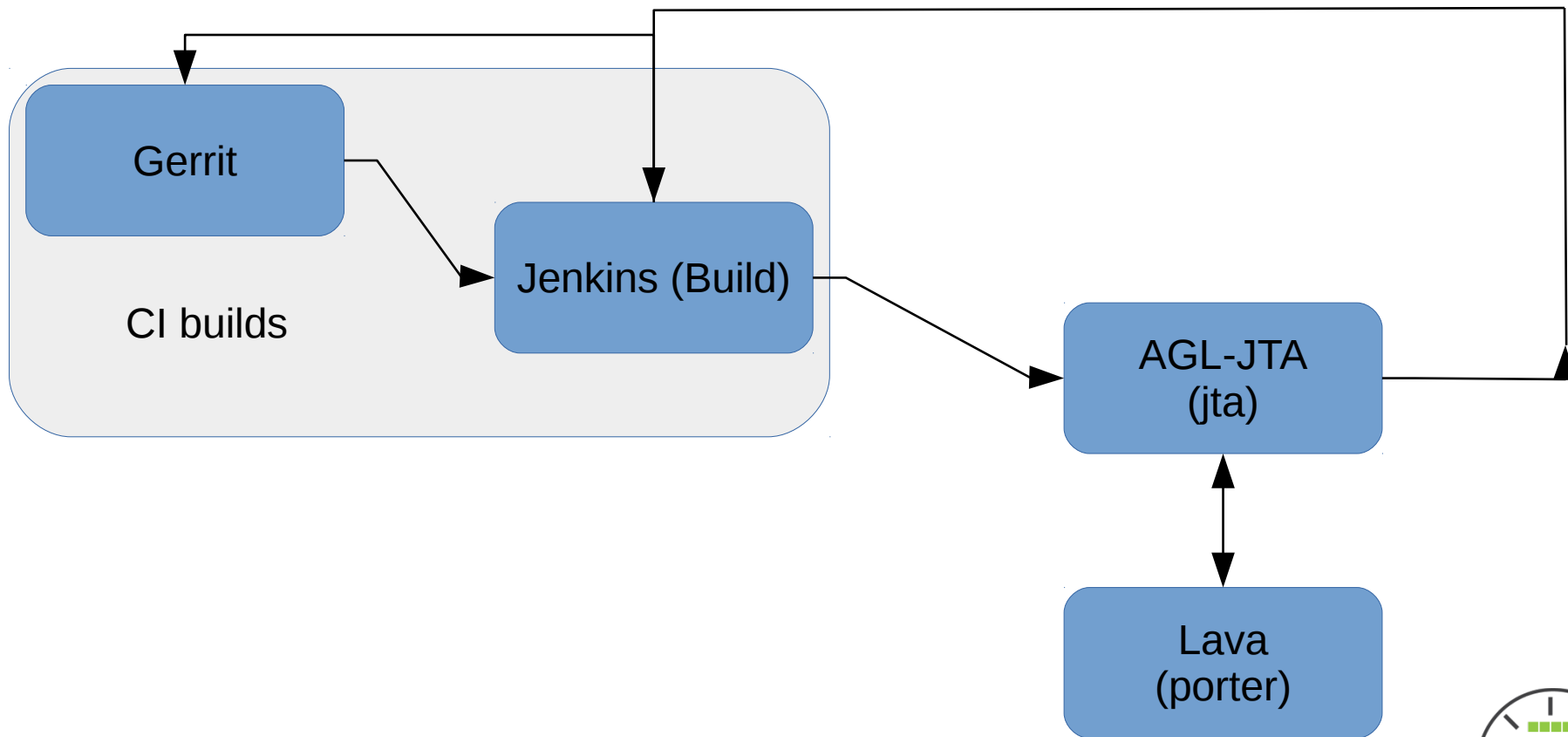
**CI-Image-UI-Test**  
("The UI tests pass")

	Size	CR	V	CIBT	CIB	CILT	CIUT
1		+1	✓				
1			✓				

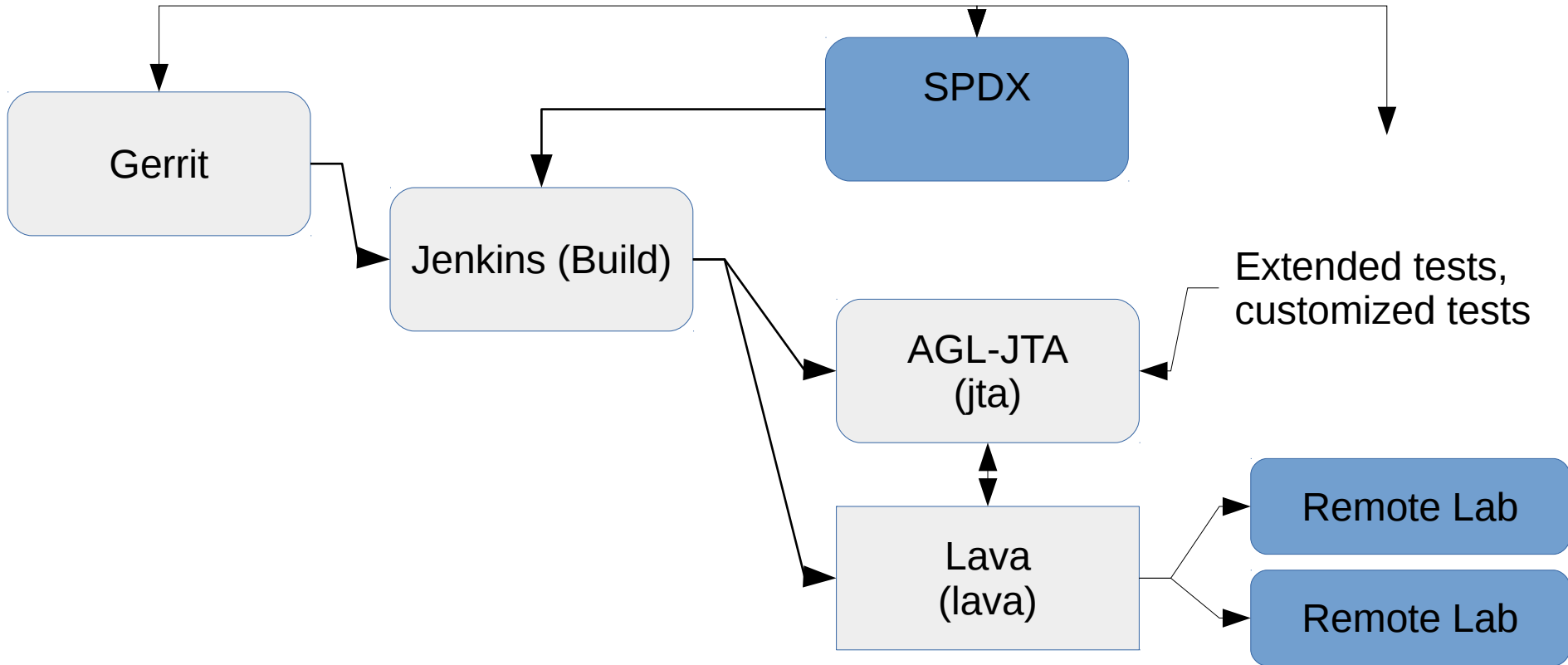


How ? ... we  
combine them

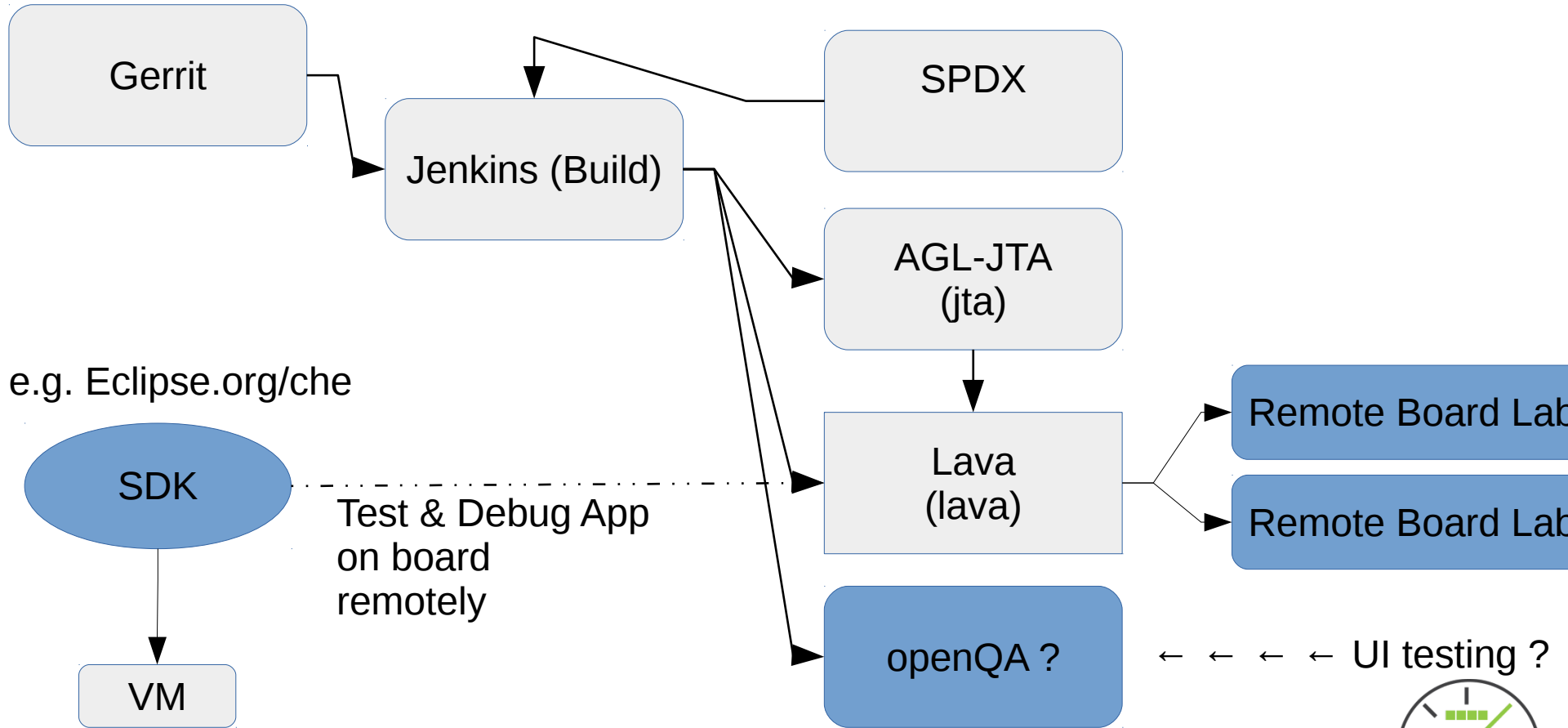
# Present



# Plan (short/mid)



# Plan (mid/long)





What ? ... we want  
to achieve

# What ? ... we want to achieve (Vision)

- Stable and tested platform to build-upon
  - wide range of devices
- Fast development through 'instant' feedback
  - developers work remotely, not all boards available
- Easy development through direct test on hw
  - remote testing capabilities in combination with SDK



Q/A ?!





# THANK YOU