

**CodeThink** Providing  
Genius.

# Testing and Open Source in automotive: beyond automation and frameworks

Event: ALS 2017

Schedule: June 1st 2017

License: [CC BY-SA](#)

© Codethink Ltd

Agustín Benito Bethencourt

Principal Consultant - FOSS

Codethink Ltd



# Who are Codethink?

- Provide software engineering & consultancy services.
- Expert in Linux and Open Source software.
- Focus on automotive industry and embedded devices.
- UK Headquarters, serving clients in EU, US and Asia.
- Independent and unbiased.
- Membership: AGL (2015), OIN (2010), CIP (founder member), GENIVI (2012)...

# Why Codethink?

Open Source company.

+

Automotive experience.

+

Passionate about building &  
delivering complex Linux systems.

# The speaker: [@toscalix](#)

- Principal Consultant - FOSS at [Codethink Ltd](#)
- Experienced working in the open and managing development & delivery of complex Linux based systems.
- Contributor in several FOSS communities (LTSP, KDE, openSUSE...).
- More about myself at <http://www.toscalix.com>

# Talking points

1. From back to front.
2. Responsiveness vs. coverage.
3. Treat test(ing) like code(ing).
4. Apply Open Source “quality pillars”.
5. Takeaways.

# 1. From back to front

- Testing at deployment and validation stages (post integration) first.
- Acceptance / system testing first.
- Killing elephants with a nail is hard. Users first, then developers.

## 2. Responsiveness vs coverage

- Shorten feedback loops first, then increase test coverage.
- Bill Gates' **second rule** of any technology applied to business (testing), is very often true.
- Focus on trust first: transparency.



### 3. Treat test(ing) like cod(ing)

- Good code takes time and knowledge, so good tests.
- Good code is self readable, so good tests.
- Good code will eventually die, so good tests.

## 4. Apply Open Source “quality pillars”.

- Transparency.
- Code review.
- Dogfooding.
- Adapt release process to maximise beta testing.

Details in this [blog post](#).

# 5. Takeaways

- Once you gain efficiency and flexibility, testing becomes a differentiation factor.
- There is no control without the source code and the knowledge associated with it: work with upstream.
- If you consume, develop and deliver Open Source software...become an Open Source company.

Thanks.

Questions?

**CodeThink** Providing  
Genius.