
HOW DOES ROS CARE ABOUT QUALITY?

A preliminary study of ROS Quality Assurance (QA) practices and the nature of ROS bugs



rosin-project.eu

CO-AUTHORS OF THE DELIVERABLE

- Yvonne Dittrich, ITU (brabrand@itu.dk)
- Claus Brabrand, ITU (brabrand@itu.dk)
- Gijs van der Hoorn, TUD (g.a.vanderhoorn@tudelft.nl)
- Jon Azpiazu, TEC (jon.azpiazu@tecnalia.com)
- Iñigo Martinez, TEC (inigo.martinez@tecnalia.com)
- Nicolas Limpert, FHA (nicolas.limpert@alumni.fh-aachen.de)
- Jonathan Hechtbauer, IPA (jonathan.hechtbauer@ipa.fraunhofer.de)
- Jon Tjerngren, ABB (jon.tjerngren@se.abb.com)
- Andrzej Wąsowski, ITU (wasowski@itu.dk)
- Adam Alami, ITU (adaa@itu.dk)

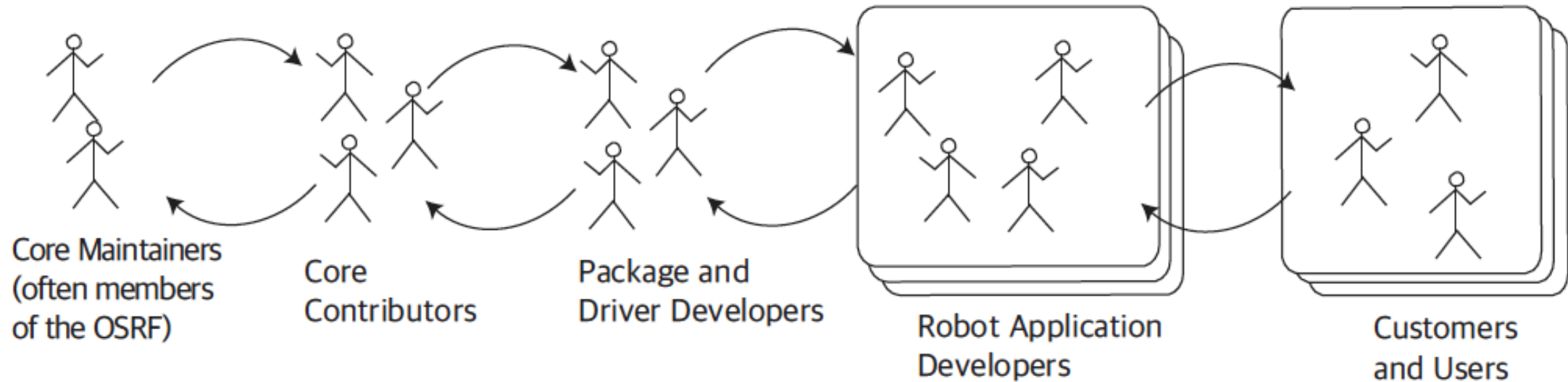
-
- 2 slides about Software Engineering
 - The studies
 - Bug Analysis
 - The ROS communities view
 - What to do with the results
 - First Steps

Software Qualities

ISO 25010

ROS as Software Ecosystem

Dittrich, Y. (2014). Software engineering beyond the project—Sustaining software ecosystems. *Information and Software Technology*, 56(11), 1436-1456.



Agenda:



Group of persons/
stakeholder

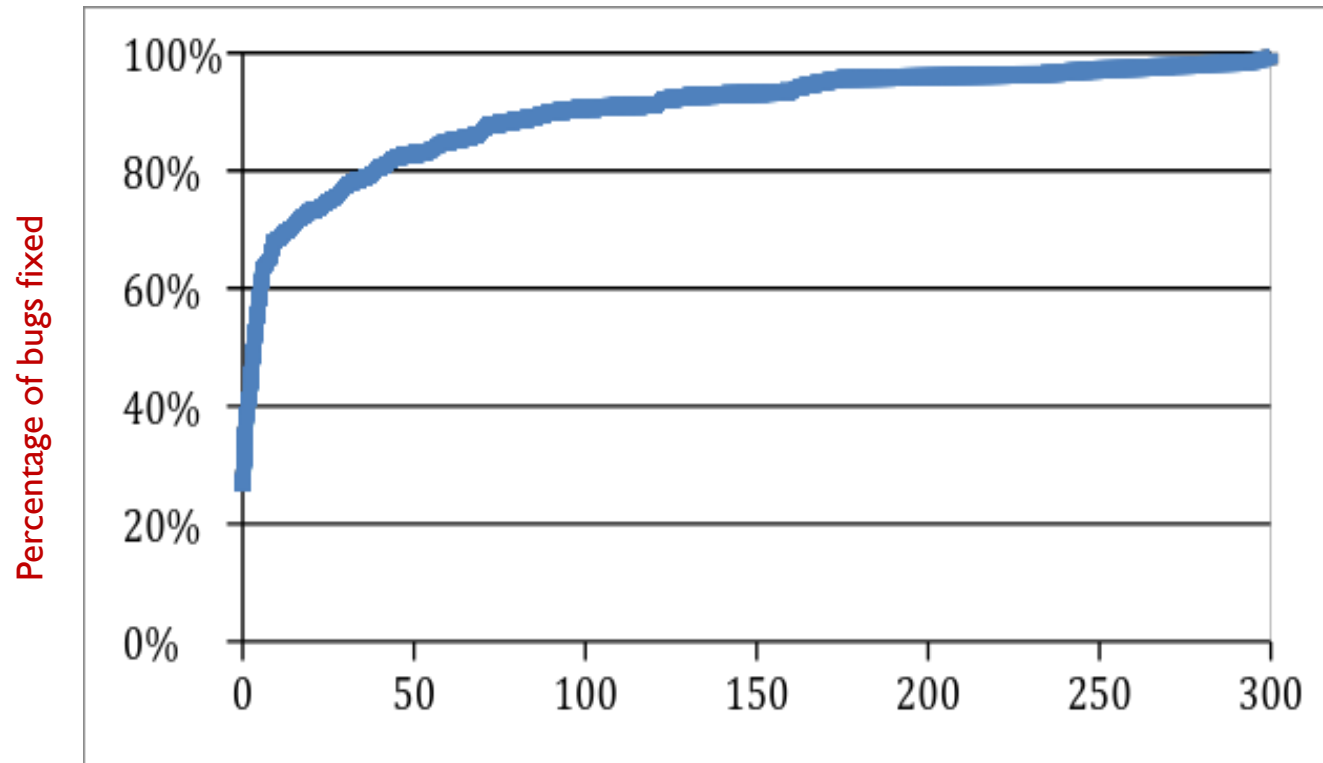


(Sub)
Organisation

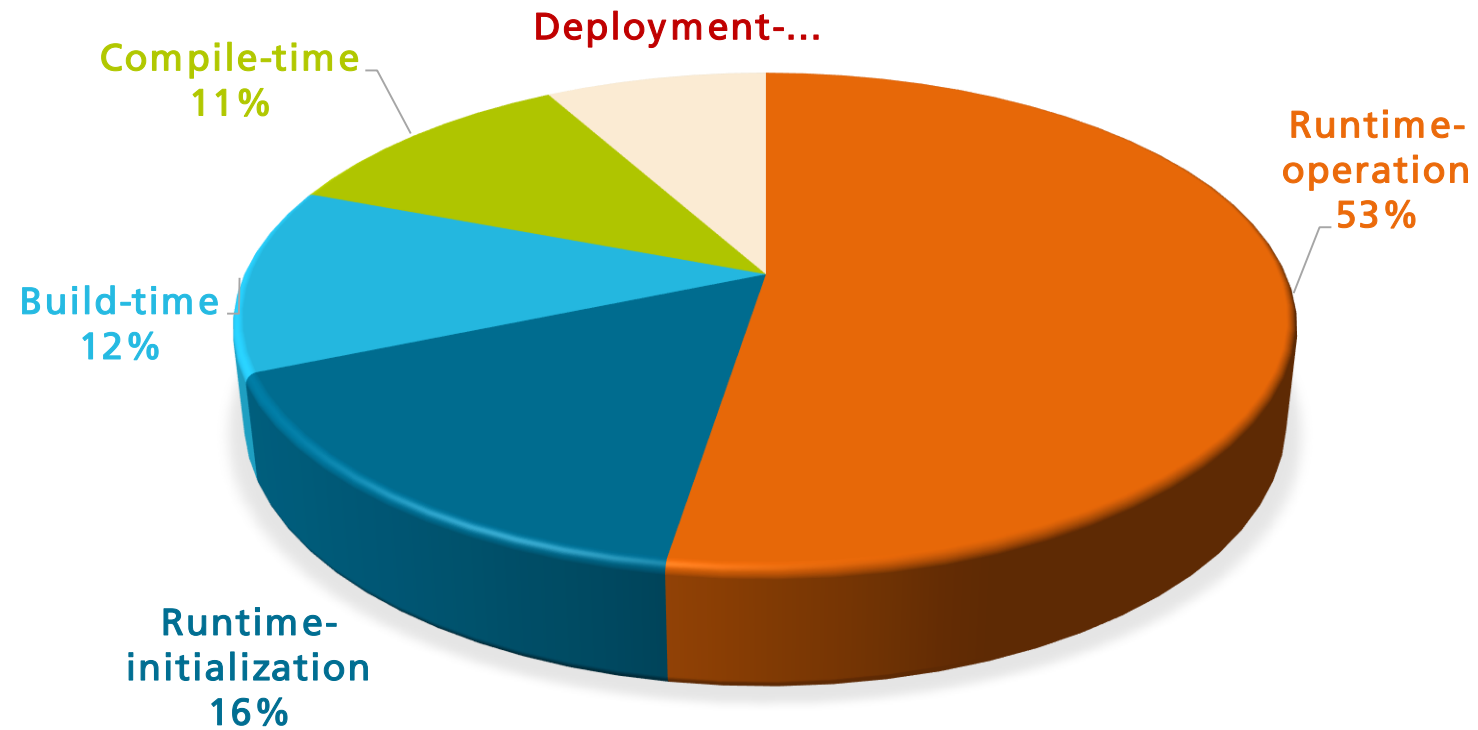
The Preliminary Study

- Qualitative:
 - Interviews with community members
 - Analysis of ROS Wiki and online documentation
- Quantitative:
 - The analysis of 177 reported bugs of ROS-Industrial code repositories

Bug Analysis: How long does it take to fix a bug?

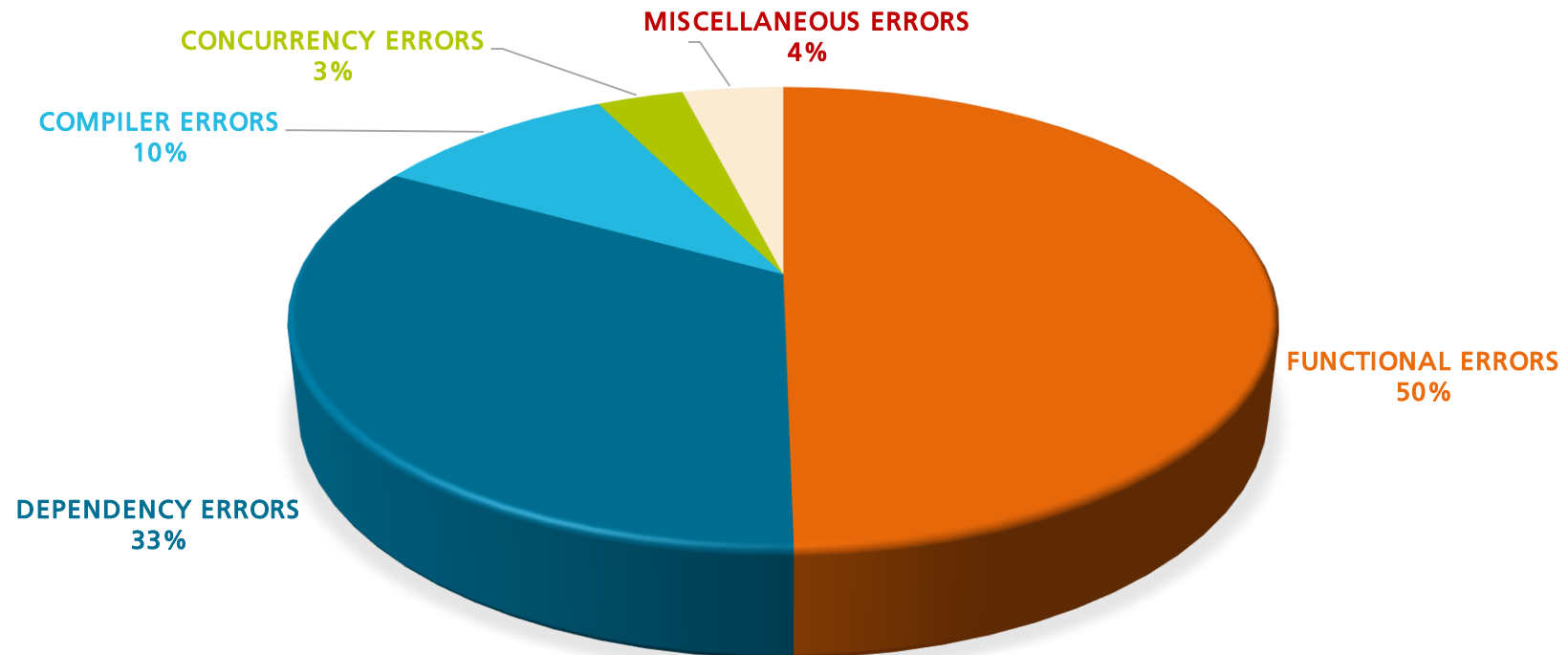


Bug Analysis: Detection Phase



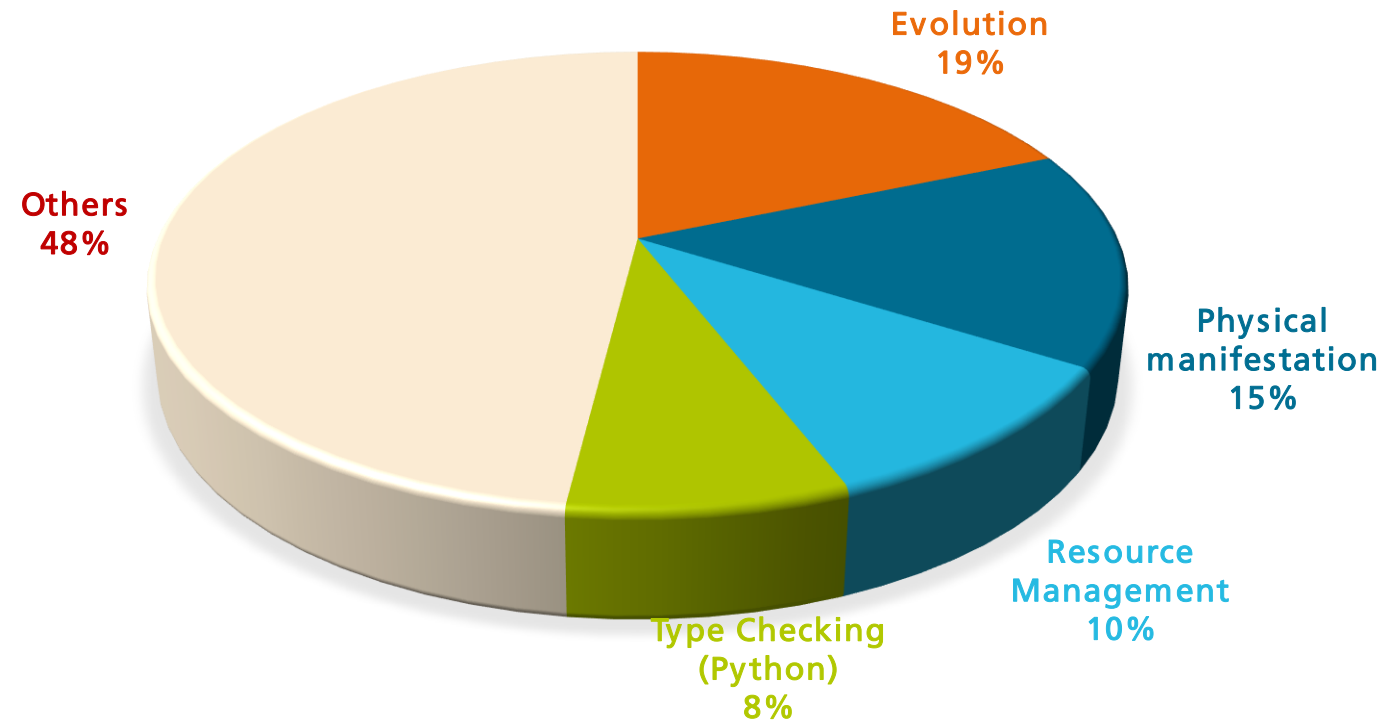
- One out of five bugs are detected during usage (either by users or at runtime).
- One out of seven bugs are reported by users.

Bug Analysis: Functional Classification



Bug Analysis:

- One out of five bugs is due to evolution



The ROS Community View

Dittrich, Y. (2016). What does it mean to use a method?
Towards a practice theory for software engineering.
Information and Software Technology, 70, 220-231.

- Many good QA and QC practices are in place
 - We have formulated them in form of method patterns.
 - We plan to publish them to provide orientation to community members.
- Challenges are partly role specific
- Quality Assurance related wiki contents is distributed and partly outdated.

Limitation:

The results presented here are based on few interviews

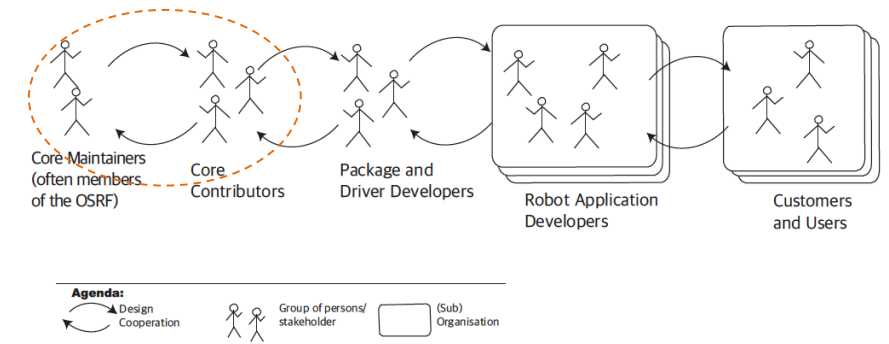
The ROS Community View: Core development

Challenges

- Lack of overview for new contributors
- Maintainers have heterogeneous quality criteria
- High maintenance effort and few resources
- Lack of maintainers
- Unmaintained packages
- Some errors only show up after extended use

Remedies

- Clarifying code quality and QA standards
- Onboarding of new core developers and maintainers
- Improving CI and Build Farm with Static Analysis and Linters



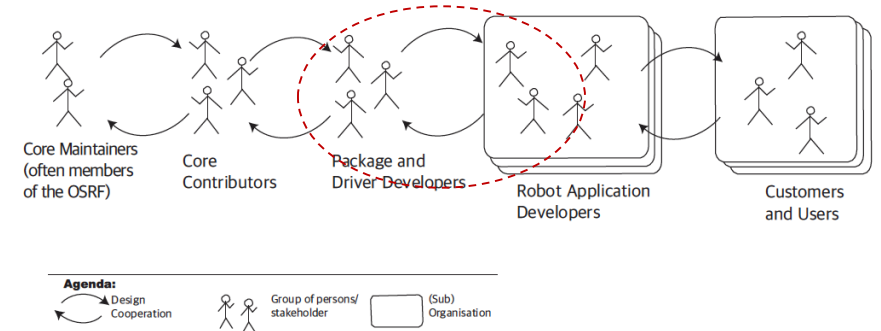
The ROS Community View: Package or Driver Development

Challenges

- Software hardware integration
- Quality of architectural design
- Lack of established corporate processes to quality assure open source contributions

Remedies

- Documentation
- Code Review
- Build farm and Continuous integration
- Continuous Testing



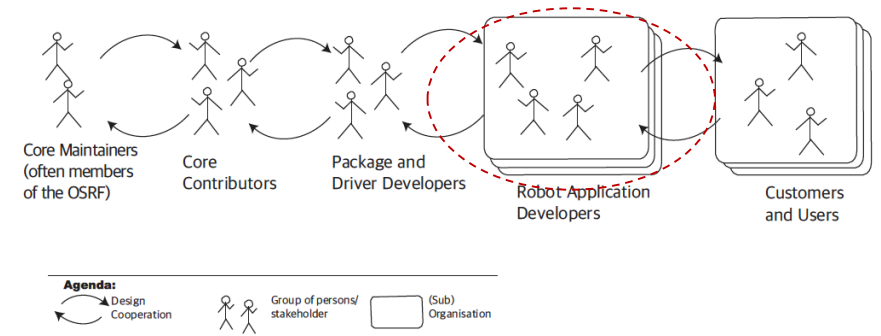
The ROS Community View: Application Develop

Challenges

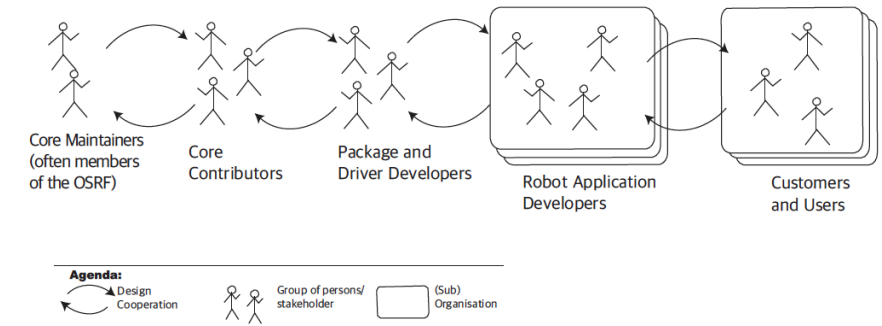
- Complexity
- Selecting the right module
- Interdisciplinary domain
- User interfaces

Remedies

- Documentation
- Quality indicators for Modules
- Continuous integration
- Testing support
- Debugging support



What to do with the results?



- Clarifying Quality Assurance and Quality Control processes in ROS and ROS-Industrial
- Making Quality Assurance and Quality Control practices easily available
- Making Quality of packages Visible: Support for non-core package and driver development and usage.
- Supporting contributors and maintainers to take care of ROS' quality
- Developing a ROS community quality culture

- Development and use of code scanning tools

First Steps

- Modernizing, Tailoring, and Scaling up the Continuous Integration Service
- Build Farm and ROS Wiki
- Industrial CI
- ROS Quality Hub (see also <https://quality.mozilla.org/>)

THANK YOU



rosin-project.eu

