

# SOFTWARE SUSTAINABILITY WITHIN RESEARCH INFRASTRUCTURES

Carsten Thiel – CESSDA ERIC

*EURISE Network Workshop – Utrecht, NL, 12-13 March 2019*





# RESEARCH SOFTWARE

# RESEARCH SOFTWARE

- Facilitates modern research
- Access to and use of data

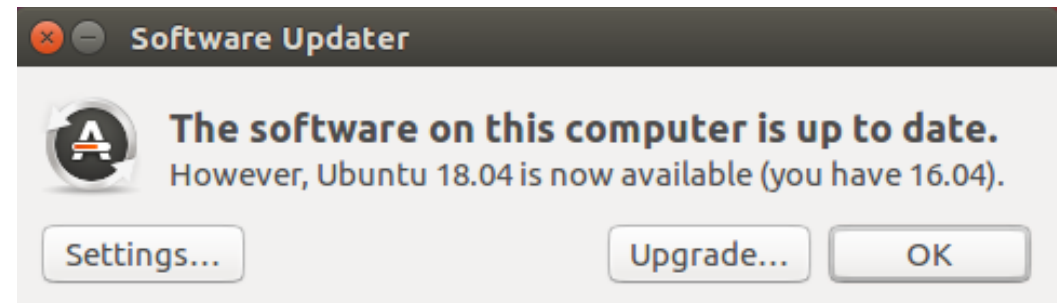
“[...] software that is developed within academia and used for the purposes of research: to generate, process and analyse results. This includes a broad range of software, from highly developed packages with significant user bases to short [...] programs written by researchers for their own use.”

Hettrick, Simon (2016): “Research Software Sustainability: Report on a Knowledge Exchange Workshop”

- Research projects produce demonstrators/prototypes!

# CHALLENGES

- Projects produce prototypes
- People turnover
- Varying degrees of development experience
- Constantly evolving IT landscape
- Legacy code (in obscure languages)



```
program hello
  print *, "Hello World!"
end program hello
```

```
package main
import "fmt"
func main() {
    fmt.Println("hello world")
}
```

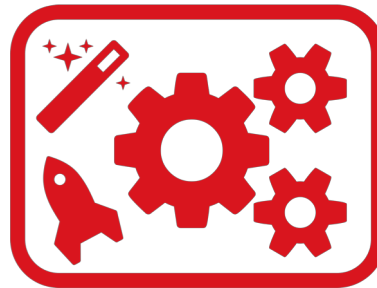


# INFRASTRUCTURES' VIEW

# BRIDGING A GAP



- Developers
- Researchers/Scholars
- Projects

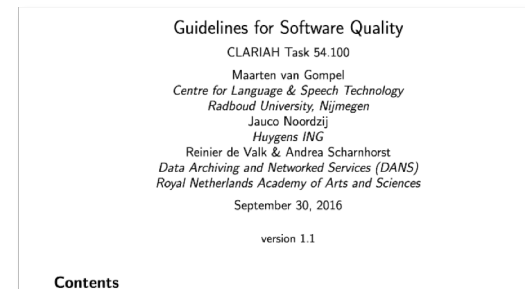
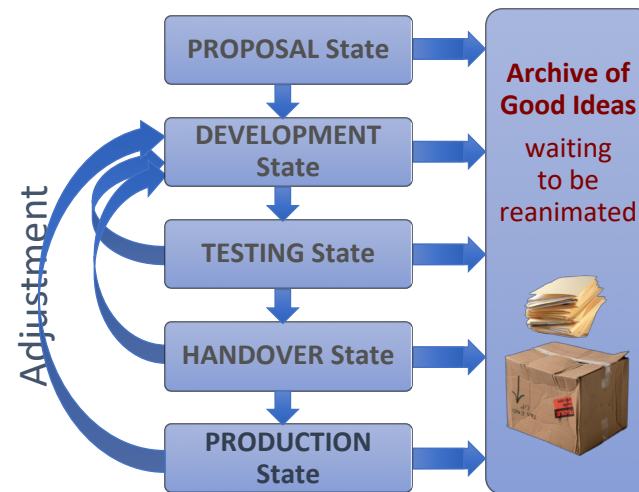


- Data Centers
- Libraries/Archives
- Horizontal Infrastructures



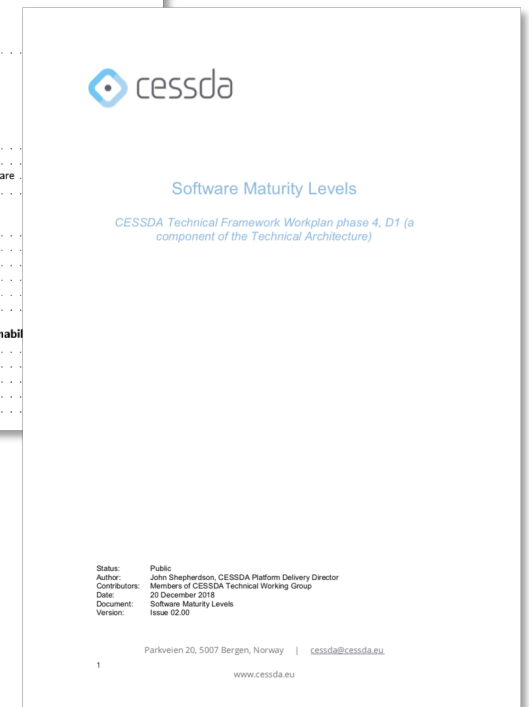
# STANDARDS AND REQUIREMENTS

- Guidelines & Best Practices
- Documentation
- Quality Requirements



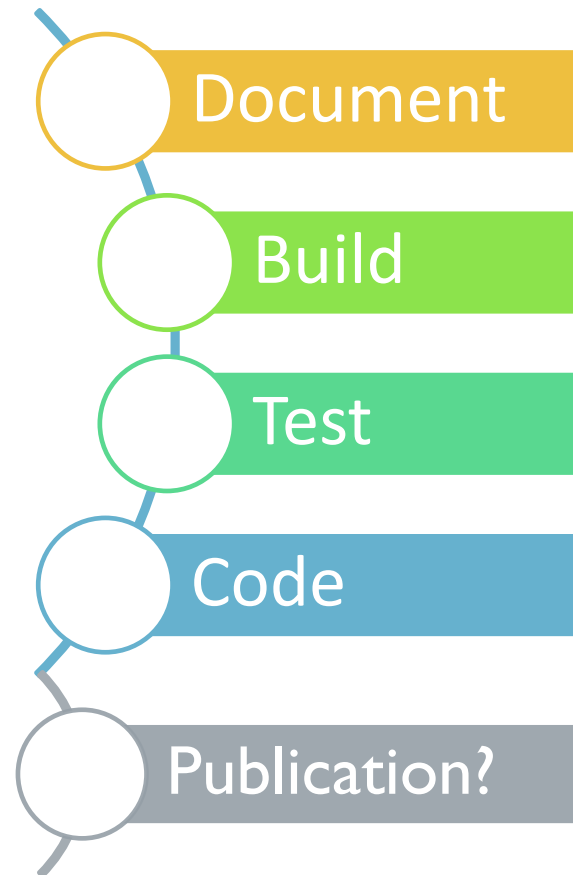
Contents

- 1 Motivation
  - 1.1 Structure of the Document
- 2 Context
- 3 Introduction - glossary of terms
- 4 Developer Guidelines: Minimal Requirements
  - 4.1 Configuration 1: Actively Supported End User Software
  - 4.2 Configuration 2: Unsupported End User Software
  - 4.3 Configuration 3: Actively Supported Experimental Software
  - 4.4 Configuration 4: Unsupported Experimental Software
- 5 Quality Assessment Criteria - Usability
  - 5.1 Understandability
  - 5.2 Documentation
  - 5.3 Learnability
  - 5.4 Buildability
  - 5.5 Instability
  - 5.6 Performance
- 6 Quality Assessment Criteria - Sustainability and Maintainability
  - 6.1 Identity
  - 6.2 Copyright & Licensing
  - 6.3 Accessibility
  - 6.4 Community
  - 6.5 Testability



# PRIORITIES

Scaling & weighting



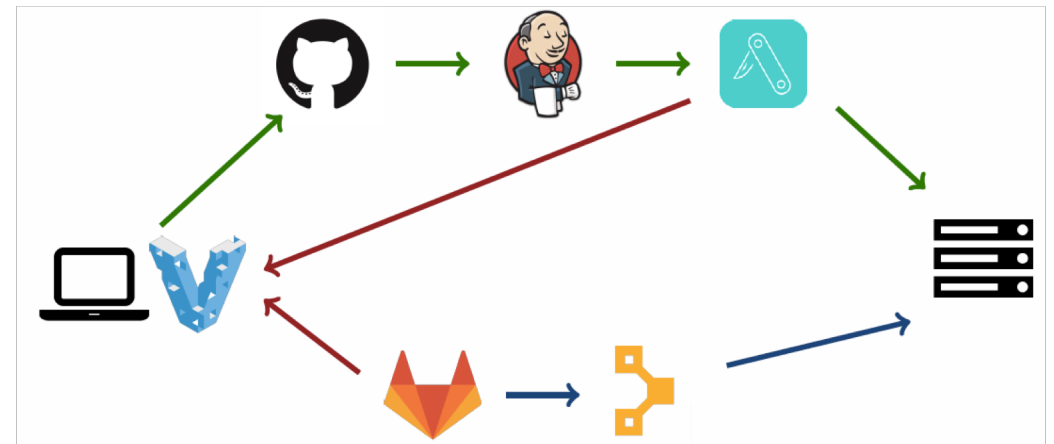




# SOFTWARE ENGINEERING

# INDUSTRY STANDARDS

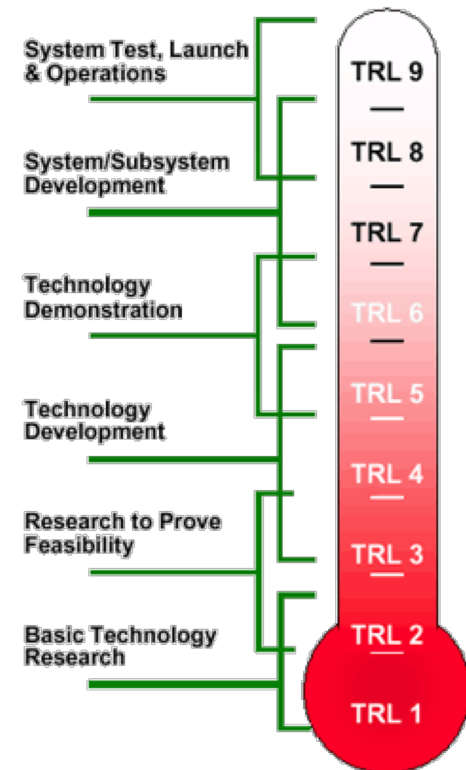
- Automation
- Testing
- Continuous Everything



# TECHNOLOGY READINESS

EOSC – The European Science Cloud will require Technology Readiness Level 8!

- We need to ensure the software quality of our services
  - We need to be able to provide evidence for our technical maturity
  - We need to be operate the services in a reliable manner!
- Training! – Software Carpentry & Code Refinery



# RESEARCH SOFTWARE ENGINEERING

- National Initiatives (registered organisations in some countries)
  - Researchers doing software engineering
  - Software development as Research Output
- 
- UK Software Sustainability Institute
  - Netherlands eScience Center

Conferences in June in Potsdam (DE) and September in Birmingham (UK).



# OPERATIONALISATION

## Operation

### Software vs Service

- Running good software the right way

### Operation is (also) about management

- Service Lifecycle Management
- Security Management
- Business Models
- ...

## Various standards exist:

- ITIL
- FitSM
- REFEDS
- SIRTIFI
- ...



# THIS WORKSHOP



# THE EURISE NETWORK

## **European Research Infrastructure Software Engineers' Network**

- an umbrella where research infrastructures meet research software engineers
- formed by the three SSH Infrastructures CESSDA, CLARIN & DARIAH

First workshop in Berlin in October 2017 – <https://dhd-blog.org/?p=8685>

Named in spring 2018

# THE EURISE NETWORK

## Goals

- » Re-Usability of Research Software
- » Infrastructure Sustainability
- » Software Quality
- » Training & Education

❖ Technical Reference ⇒ Michelle's talk tomorrow!

The screenshot shows the 'EURISE Network Technical Reference' page. The header includes 'Technical Reference', 'EURISE Network', and version '0.2'. A search bar is present. The left sidebar lists 'CONTENTS' with links to Developer Guidelines, Operational Guidelines, Policy Recommendations, Software Quality Checklist, Glossary, and Bibliography. The main content area is titled 'EURISE Network Technical Reference' and contains several paragraphs of text, including a description of the document's purpose, its origin as part of the 'DESIR - DARIAH ERIC Sustainability Refined' project, and recommendations for its use. It also mentions the license (Creative Commons Attribution 4.0 International) and provides a 'Next' button. The footer includes copyright information (© Copyright 2018, CC-BY 4.0) and mentions it was built with Sphinx.



# PROGRAMME

	09:00 - 09:15	Coffee
	09:15 - 09:45	<b>EURISE Technical Reference</b> <i>Michelle Weidling (SUB Göttingen)</i>
	09:45 - 10:15	<b>Implementing a Service Management System in a Federated Multi-Supply Environment</b> <i>Yannick Legré (EGI)</i>
15:30 - 16:00		<b>Welcome &amp; Introduction</b> <i>Carsten Thiel (CESSDA)</i>
16:00 - 16:30		<b>CESSDA's Software Maturity Model</b> <i>John Shepherdson (CESSDA)</i>
16:30 - 17:00		<b>CLARIN's first steps on the long path to software sustainability</b> <i>Dieter van Uytvanck (CLARIN)</i>
17:00 - 17:30		<b>Business Models for Digital Research Infrastructures using the Example of DARIAH</b> <i>Frank Fischer (DARIAH)</i>
18:00 - 21:00		<b>Dinner:</b> De Utrechter Stadsbrasserie
	10:15 - 10:45	<b>Infrastructure Security</b> <i>David Kelsey (UKRI STFC)</i>
	10:45 - 11:00	Coffee break
	11:00 - 11:30	<b>A FAIR Software Route - the Dutch Way</b> <i>Mustapha Mokrane (DANS) / Carlos Martinez Ortiz (NLeSC)</i>
	11:30 - 12:00	<b>Sustainability beyond guidelines – How our perception of research infrastructures shapes our perception of the sustainability issue?</b> <i>Andrea Scharnhorst / Francesca Morselli (DANS)</i>
	12:00 - 13:00	<b>Breakout Session:</b> Discussion
	13:00 - 14:00	Lunch
	14:00 - 15:00	<b>Final Discussion</b> and Closing of the Workshop

WELCOME!

Looking forward to a fruitful workshop!