FOSDEM'22
Testing Requirements Documents

Kristoffer Nordström

Saturday, 05 February 2022

info@sltoo.dev

Welcome

- Motivation
- Theory
- Practical Aspects
- Demonstration
- Conclusion

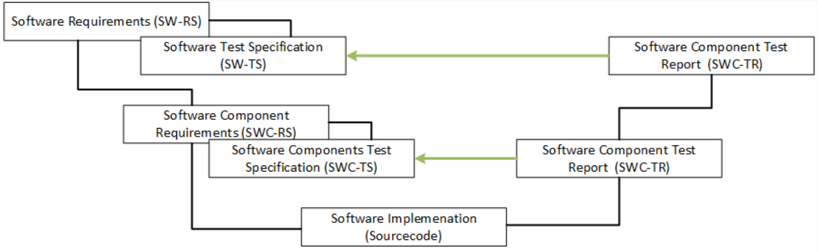
Motivation

- Automated traceability matrix
- *Consistent* and *up-to-date* documents
- Store *meta-information* with code

Theory

Requirements and Traceability

- Requirements across system hierarchies
  - Implies the need for traceability
- *Traceability* from and to specification items
- Directions
  - Forward (*Impact*)
  - Backwards



Software Requirements (SW-RS)
Software Test Specification (SW-TS)
Software Component Test Report (SWC-TR)
Software Component Requirements (SWC-RS)
Software Components Test Specification (SWC-TS)
Software Component Test Report (SWC-TR)
Software Implemenation (Sourcecode)

Traceability

- *Requirement A*
    - Red button to shut down system
- *Implementation a* says implemented A
    - Traceability can be automated
    - Machine-readable
- What if *A* changes?
    - *A* knows nothing of *a*

Traceability

- *Requirement A*
  - Green button with large friendly letters: don't panic
- *Implementation a* says implemented `A`
  - Traceability can be automated
  - Machine-readable
- What if *A* changes?
  - *A* knows nothing of *a*
  - Traceability isn't given anymore

Proposed Solution

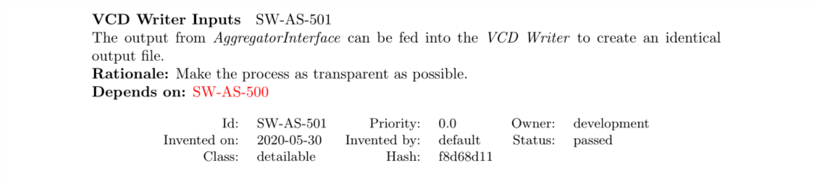
- *Requirement A-1.0*
  - Red button to shut down system
- *Implementation a* says implemented *A-1.0*
  - *A-1.0* (red button) changes to *A-2.0* (green button)
  - Use hashes instead of semantic versioning
  - Calculated automatically

Example Requirement

```
Name: VCD Writer Inputs
Topic: ReqsDocument
Description: The output from ...
Rationale: Make the process as ...
Status: external
Owner: development
Effort estimation: 1
Invented on: 2020-05-30
Invented by: default
Type: requirement
```

Hash is calculated over *Name*, *Description* and *Verification Method*

```
$ sha256sum ${Name} \
    ${Description} \
    ${VerifMethod}
```



VCD Writer Inputs SW-AS-501

The output from *AggregatorInterface* can be fed into the *VCD Writer* to create an identical output file.

Rationale: Make the process as transparent as possible.

Depends on: SW-AS-500

Id:	SW-AS-501	Priority:	0.0	Owner:	development
Invented on:	2020-05-30	Invented by:	default	Status:	passed
Class:	detailable	Hash:	f8d68d11		

Example from pymergevcd's architecture specification

Testing the Example Requirement

- Requirement ID: `SW-AS-501`
- Hash: `F8D68D11`

Test Code

```
def test_read_write_engines(record_property, dummy_vcd_file):
    """Write-back from read file, equal output"""
    record_property('req', 'SW-AS-501-f8d68d11')
    record_property('req', 'SW-AS-500-4c1a395a')
    ...
    assert filecmp.cmp(dummy_vcd_file, ofile)
```

xUnit Output

```
<testcase   line="20" name="test_read_write_engines" time="2.830">
  <properties>
    <property name="req" value="SW-AS-501-f8d68d11"/>
    <property name="req" value="SW-AS-500-4c1a395a"/>
  </properties>
</testcase>
```

Traceability Matrix

Table A.1: Traceability Matrix Table

Req. ID	Status	UT
SW-AS-1	passed	passed
SW-AS-100	open	open
SW-AS-101	open	open
SW-AS-102	open	open
SW-AS-200	failed	failed
SW-AS-201	passed	passed
SW-AS-202	passed	passed
SW-AS-300	passed	passed
SW-AS-301	open	open
SW-AS-302	open	open
SW-AS-500	passed	passed
SW-AS-501	passed	passed

Integrating Requirements into CI/CD

- Integration for every output document
- Match *open* and/or *failed* issues
    - Left as an excercise for the reader
- Example for *failed* issues

```
$ bash -ec 'test "$(grep -c failed \
    arch/artifacts/tracematrix.tex)" -eq "0"'
```

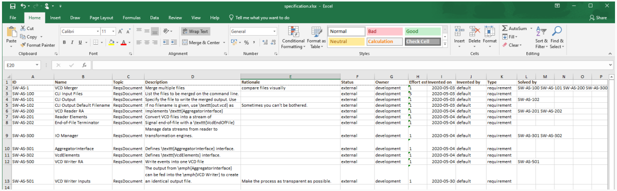
sltoo in Practice

sl. . . what?

- *sltoo* (currently) a fork of rmtoo
  - text-file based req. tracking tool
- Tracking requirements in text files with git
  - might be ideal for engineering department
- Defining system behaviour is a team effort
- Solution not for everyone / command-line unfamiliar UI

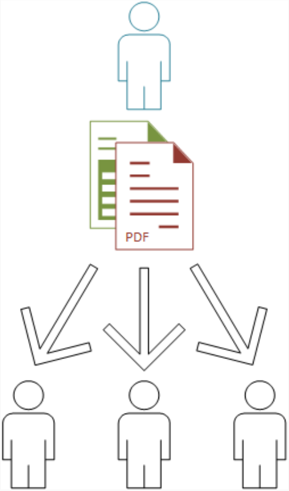
Excel Workflow (I)

- Defining system collaborative effort
- Familiarity / Ease-of-use
- Consistency of Documents
  - The *Truth* is always in your repository
  - Templating for branding
- Works if all you've got is Office and E-Mail
- Getting Started: Edit example Excel-Sheet

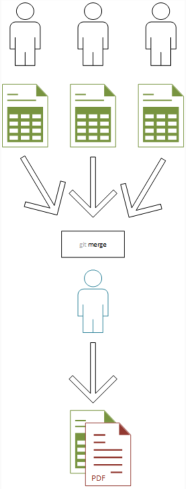


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	ID	Name	Topic	Description	Rationale	Status	Owner	Effort est	Invented on	Invented by	Type	Solved by				
2	SW-AS-1	VCD Merger	ReqsDocument	Merge multiple files	compare files visually	external	development	1	2020-05-03	default	requirement	SW-AS-100	SW-AS-101	SW-AS-200	SW-AS-300	
3	SW-AS-100	CLI Input Files	ReqsDocument	List the files to be merged on the command line.		external	development	5	2020-05-03	default	requirement					
4	SW-AS-101	CLI Output	ReqsDocument	Specify the file to write the merged output. Use		external	development	1	2020-05-03	default	requirement	SW-AS-102				
5	SW-AS-102	CLI Output Default Filename	ReqsDocument	If no filename is given, use \texttt{out.vcd} as	Sometimes you can't be bothered.	external	development	1	2020-05-03	default	requirement					
6	SW-AS-200	VCD Reader RA	ReqsDocument	Implements \texttt{AggregatorInterface}		external	development	1	2020-05-04	default	requirement	SW-AS-201	SW-AS-202			
7	SW-AS-201	Reader Elements	ReqsDocument	Convert VCD files into a stream of		external	development	1	2020-05-04	default	requirement					
8	SW-AS-202	End-of-File Terminator	ReqsDocument	Signal end-of-file with a \texttt{VcdEndOfFile}		external	development	1	2020-05-04	default	requirement					
9	SW-AS-300	IO Manager	ReqsDocument	Manage data streams from reader to transformation engines.		external	development	1	2020-05-04	default	requirement	SW-AS-301	SW-AS-302			
10	SW-AS-301	AggregatorInterface	ReqsDocument	Defines \texttt{AggregatorInterface} interface.		external	development	1	2020-05-04	default	requirement					
11	SW-AS-302	VcdElements	ReqsDocument	Defines \texttt{VcdElements} interface.		external	development	1	2020-05-04	default	requirement					
12	SW-AS-500	VCD Writer RA	ReqsDocument	Write events into one VCD file		external	development	1	2020-05-04	default	requirement	SW-AS-501				
13	SW-AS-501	VCD Writer Inputs	ReqsDocument	The output from \emph{AggregatorInterface} can be fed into the \emph{VCD Writer} to create an identical output file.	Make the process as transparent as possible.	external	development	1	2020-05-30	default	requirement					
14																

Excel Workflow (II) – Distribution



Excel Workflow (III) – Merging



Document Baseline

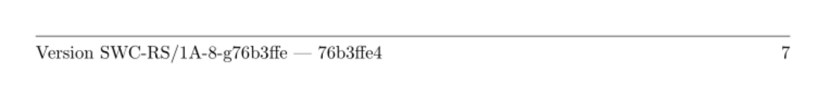
Every document has a its own version tag

```
$ git tag -a RS/1A
$ git describe $(git log -n 1 --format=%H -- docs/reqs)
```

The output from `git describe` will be used as document baseline

```
RS/1A - 0aec3ad0                # good
RS/1A-8-g76b3ffe - 76b3ffe4     # tainted
```

Example excerpt from page 7:



Version SWC-RS/1A-8-g76b3ffe — 76b3ffe4 7

Outlook

- Merge changes with *rmtoo*
- Transform *sltoo* into a GUI
  - Provide a GUI to show relations between items
  - Provide cross-documents editing capabilities
  - Update hashes in sources
- How much can AI help?

Demonstration

Demonstration Notes

- Traceability still in beta
  - Template Project
  - hopefully released by 5th of February
- Demo is in `FOSDEM22` branch
  - ESA DOORS dump
  - use your search engine of choice
- Set *Status* from *not done* to *external*
- Configure lengths not to be affected by large ESA requirements
  - see `Config.json`
- Images, tables and graphs are to be demonstrated, see the homepage for updates

Conclusion

Storytime Revisited

- Requirements shipped with code ✓
  - Including relational meta-information
  - Use commit message to document the *why*
- Traceability matrix automated ✓
- Continuously updated documentation ✓
  - Document Versioning (baselining) ✓

A familiar UI for all stakeholders included

Questions

Appendix

rmtoo – Introductions

An introduction presentation into *rmtoo* and with more details.

Traceability Rationale

- Traceability for the given requirements
- Bring code and documentation into same repository
- Integrate into build-system
  - Detect upstream changes to requirements
  - Quickly identify affected code-regions
- No silver bullet for verification

Results

The status *external* will yield the following results:

- *open*
    - No matching requirement ID
- *passed*
    - Matching requirement ID
    - All hashes match
    - Unit-tests passed
- *failed*
    - Matching requirement ID
    - Some/all hashes didn’t match, or
    - Unit-tests haven’t passed

Installation

- Traceability features are in the beta releases.
- Document to document traceability requires beta-4 at least

```
$ pip3 install sltoo>=25.1.0b4
$ wget https://kown7.github.io/pymergevcd/assets/template_p
```

Alternatives

- Sphinx-Needs
- Octane ALM
- Codebeamer
- Aligned elements
- See Wikipedia
- . . .

Future Developements

- Write Parser for *Test Reports* ✓
- Documents with the correct identifier automatically solve the specification
    - Document Formats:
        - docx (maybe with pandoc)
        - LaTeX✓
        - Text
        - CAD Files from HW/Mechanical
- GUI with multi-documents support (RS/TS/..)
    - Simplify design process

Final Thoughts

- Never test against your requirements
- Always write some form of test specification

Licensing