



Multi-Platform SCADA GUI Regression Testing at CERN

Paul Burkimsher

Manuel Gonzalez-Berges

Stefan Klikovits

**ICE Group, EN Department,
CERN, Geneva, Switzerland**

a.k.a.

Practical Experience With Sexy Software



- **Hands-on, practical talk**
- **Why are you interested?**
 - You can laugh at other people's mistakes
 - You may be about to make the same journey (with same mistakes!) as me 😊
- **What will you gain from this talk?**
 - A desire to actually read our paper (?)
 - A chance to get your Testing System right, 1st time

- **Beware the siren song**
 - Truthful(!) salesmen
 - But what remains unsaid?
- **Testing Cross-platform (vs Multi-platform)**
 - Messy
- **Virtual machines**
 - Appropriate

Layout Of This Talk

- **Introduction:** *GUI Quality Assurance*
- **Testing what?** *JCOP Framework*
- **How?** *Automatic; Squish*
- **Orchestration:** *Continuous Integration tool.*

- **Lessons:** **“In the light of experience...”**
 - What went badly 1st time?
 - How did we respond?

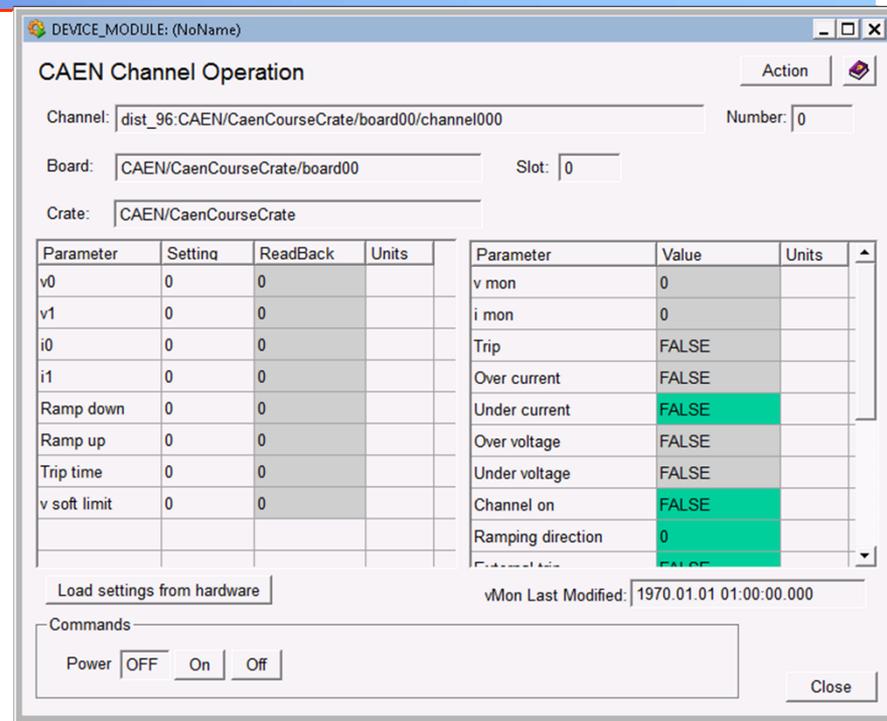


GUI Quality Assurance (QA)



- **Lots of people test their software** (some even document it...)
 - Unit testing; Black box, white box
 - ~ Easy(?) Straightforward
- **Testing SCADA (control) systems**
 - Many Graphical User Interface (GUI) panels
 - Need testing too
 - Don't want to do it manually
 - Not so easy. Especially when there is control logic behind the buttons etc

- **JCOP Framework**
 - **A toolkit for end users to build their own control applications**
 - **Long-lifetime project (-10..+15 years)**
 - **Staff rotation**
 - **Natural turnover**
 - **Use Experiment staff/students**
 - **New developers -> unintended side-effects**



(Re-)Testing is crucial

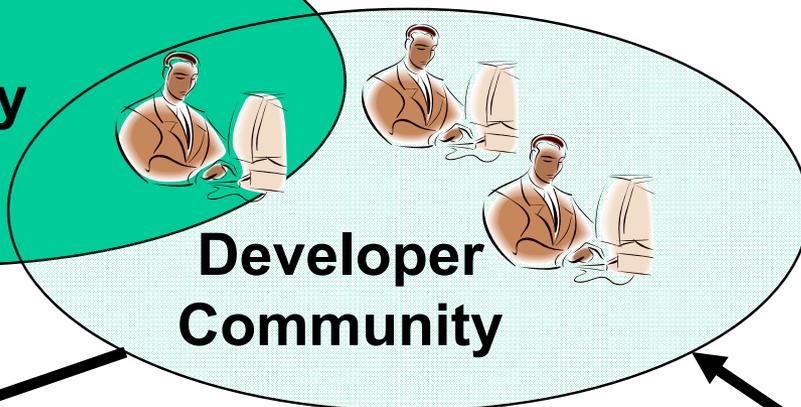
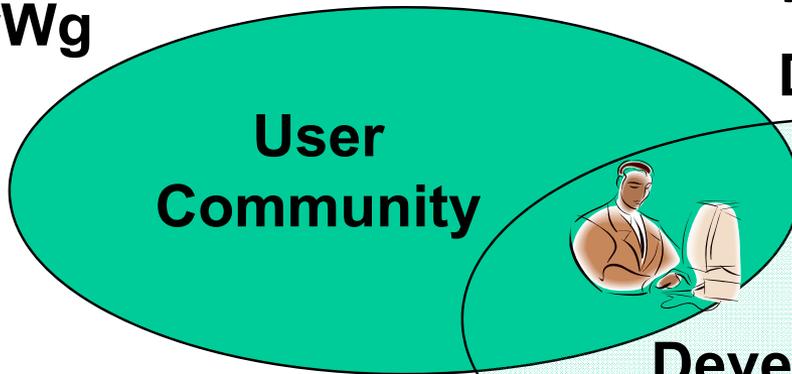


JCOP Fw Development Process

Many Components;

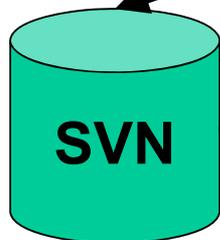
Different Developers over time

FwWg

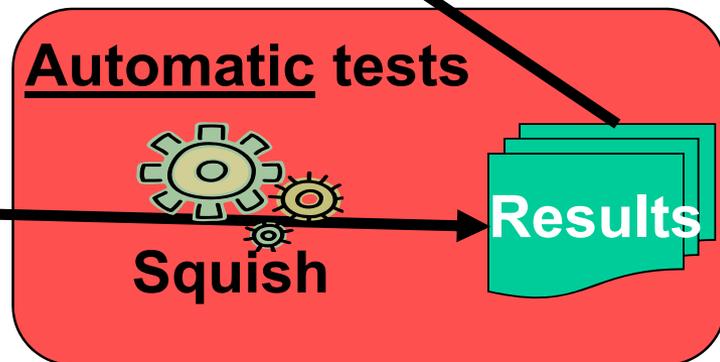
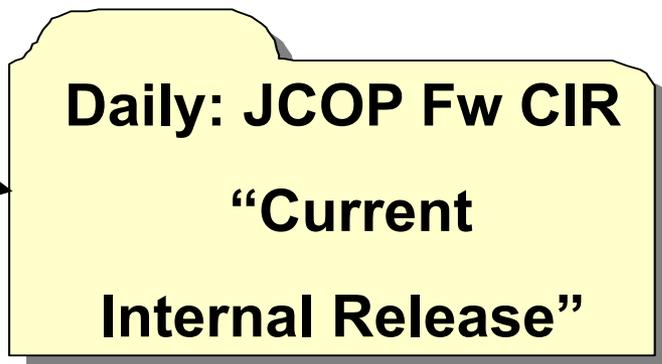


Morning after:
Emailed test report

Commit new code & bug fixes to

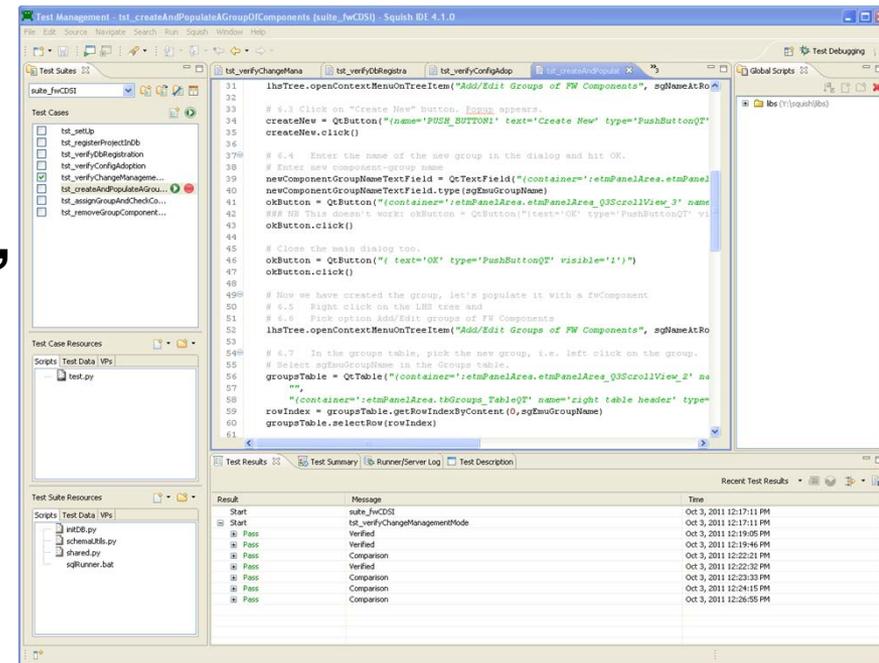


Subversion repository

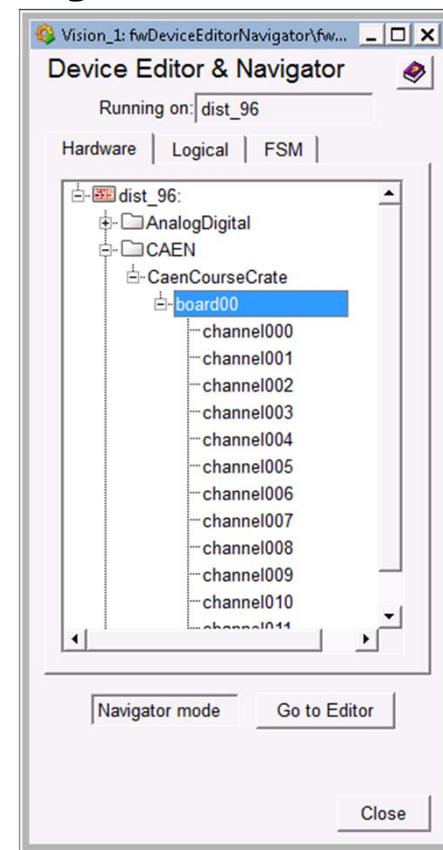


How Do We GUI-Test JCOP-Fw?

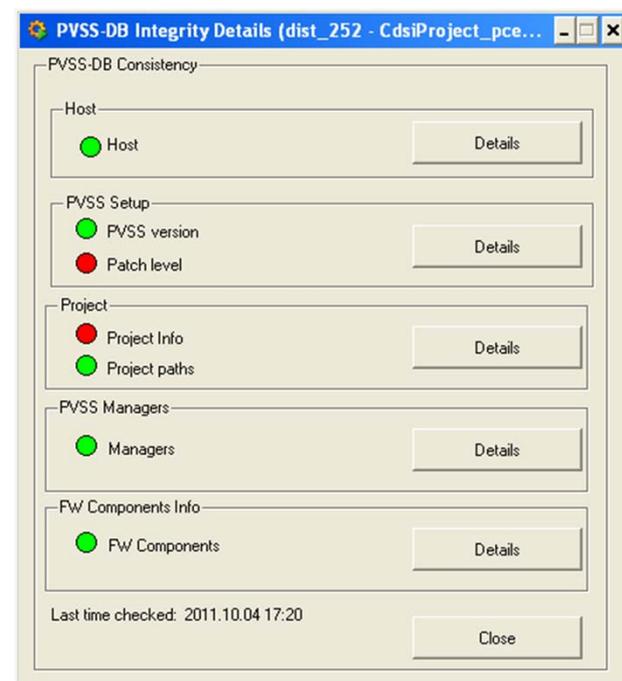
- **Commercial tool:**
 - (Rational Robot)
 - Squish (Qt) from Froglogic GmbH
- **Very powerful**
 - Intuitive
 - “Record my keystrokes”
 - “Record my mouseclicks”
 - Generates source code
 - Replay 😊
 - Regression testing done!



- A library call like `sqrt(2)` would be easy to check
- But Fw tool is used to
 - Declare hardware in a new control system
 - Define alarms on values
 - Replay & verify get same definitions as yesterday.



- A library call like `sqrt(2)` would be easy to check
- But Fw tool is used to
 - Declare hardware in a new control system
 - Define alarms on values
 - Replay & verify get same definitions as yesterday.
- User presses a button, e.g. to connect to the database
 - Verify that the LED is green



- **Tools to verify screen conditions:**
 - Fields contain correct values, are correct colour,

[-] Start	tst_verifyChangeManagementMode
[+] Pass	Verified
[-] Pass	Verified
Detail	True expression
Detail	suite_FwCDSI:5.6:tst_verifyChangeManagementMode == Central
[+] Pass	Comparison

- **...cannot verify application-specific things:**
 - WinCC-OA (PVSS) from Siemens (ETM)
 - Export definitions from WinCC
 - Write our own comparison tool (Totem)

- **Squish tests the JCOP-Framework,**
 - which uses WinCC-OA,
 - which is built on Qt,
 - which uses Windows or Linux



- **Any changes can break the pre-recorded tests**

- Squish tests the JCOP-Framework,

Lesson 1

- Simply recording keystrokes and clicks is not scalable to many tests

- We had to seriously re-factor the code into libraries to make it maintainable



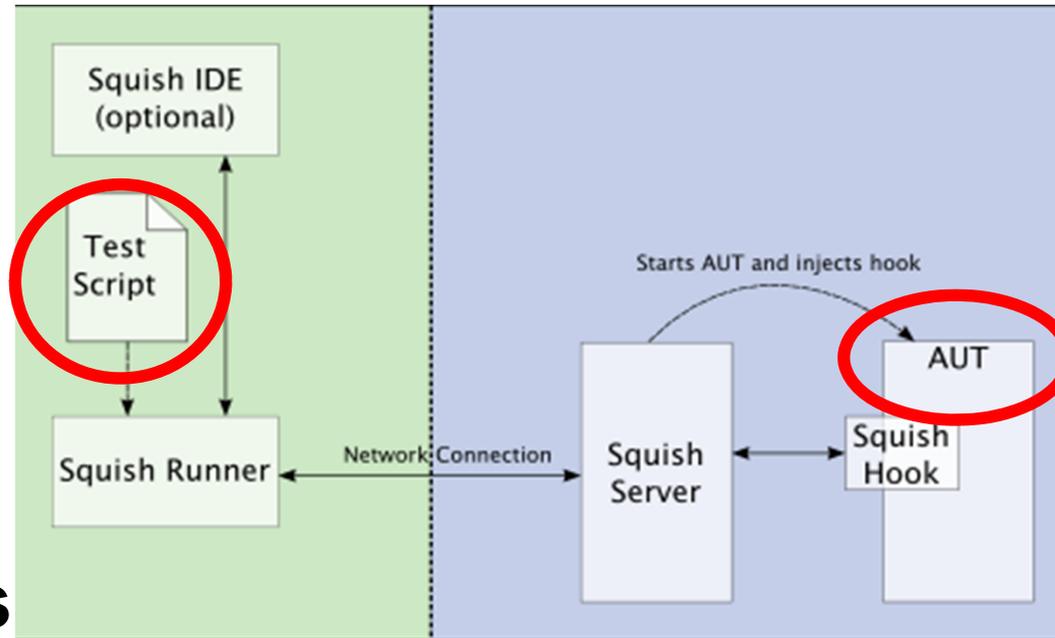
- Any changes can break the pre-recorded tests

- **Squish development environment is pretty...
...but that's only half the story.**
 - **Overnight-runs started from a command line script**
- **Difficulties with the fundamentals**
 - **Bash to trigger the production runs:**
 - **Windows and Linux.**
 - **Seemed like a good idea at the time.**
 - **Incompatibilities**
 - **Cygwin-Bash (on Windows) vs Linux-Bash**
 - **One script for 2 platforms => Messy scripts**

- Squish has a client-server mode of operation



**Unique
(clean!)
test
scripts
on
Windows**



**Target
application
(AUT) on
Linux.**



- **Many Linux X-Windows**
 - Display back to the client (on Microsoft Windows)
- **Remote file access**
 - Returning results across the link
- **In practice ☹ we suffered networking timeouts**
 - All our displays (and open files) would then collapse
- **Despite trying various workarounds (e.g. VNC), we had to change tack**

- Many Linux X-Windows

– D

- Re
- F
- In p
- A
- Des

had to change tack

Lesson 2

Cross-platform gets messy

- Messy scripts.
- Messy timeouts.

se
, we

Profound Reflection



- **Abandon Bash**
 - **Use Python**
 - **Better compatibility across platforms**
 - **Consistent with use of Python within Squish**
- **Returned to running the Squish test scripts on the target platform being tested**
- **And...**

- **Orchestrate many different platform instances (Windows XP, W7, Linux SLCn, n+1 etc)**
- **Looked at tools to**
 - **Distribute the testing**
 - **Centralise the results.**
- **Bamboo (from Atlassian) ?**
 - **but their model is different to ours.**

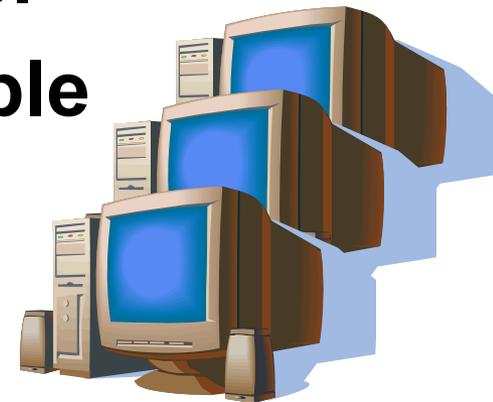
The screenshot shows the Hudson web interface. At the top, there's a search bar and a link to 'ENABLE AUTO REFRESH'. Below that, a navigation menu on the left includes 'New Job', 'Manage Hudson', 'People', 'Build History', 'Edit View', and 'Delete View'. The main content area displays a table of jobs with columns for status (S), weather icon (W), job name, last success, and last failure. The jobs listed are 'BuildEmu', 'paul_cvpc8', 'win7_382', and 'winXP_382'. Below the table, there are links for 'Icon: S M L' and a legend for RSS feeds: 'for all', 'for failures', and 'for just latest builds'.

S	W	Job ↓	Last Success	Last Failure
		BuildEmu	34 min (#214)	4 mo 13 days (#1)
		paul_cvpc8	11 days (#42)	12 days (#41)
		win7_382	1 day 13 hr (#167)	13 hr (#168)
		winXP_382	14 hr (#290)	N/A

- Hudson is a **Continuous Integration (CI)** tool that is
 - Flexible
 - Open Source (free licensing)
- Works well



- **CERN Virtual Machine (VM) service.**
- **Dedicated machines make life simple**
 - Different O/Ses
 - Different software versions
- **VMs work like the real thing 😊**
 - **Functionality tests, not performance tests!**
 - **Speed/timing issues on different platforms (real or virtual) !**
- **CERN is happy too. (VM Server reallocation)**



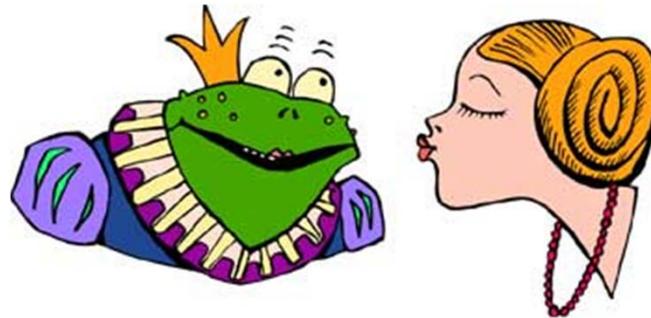
“Experience” Is Unfinished

- **Real life is ongoing** (fortunately).
- **Setting up on Linux again right now.** 

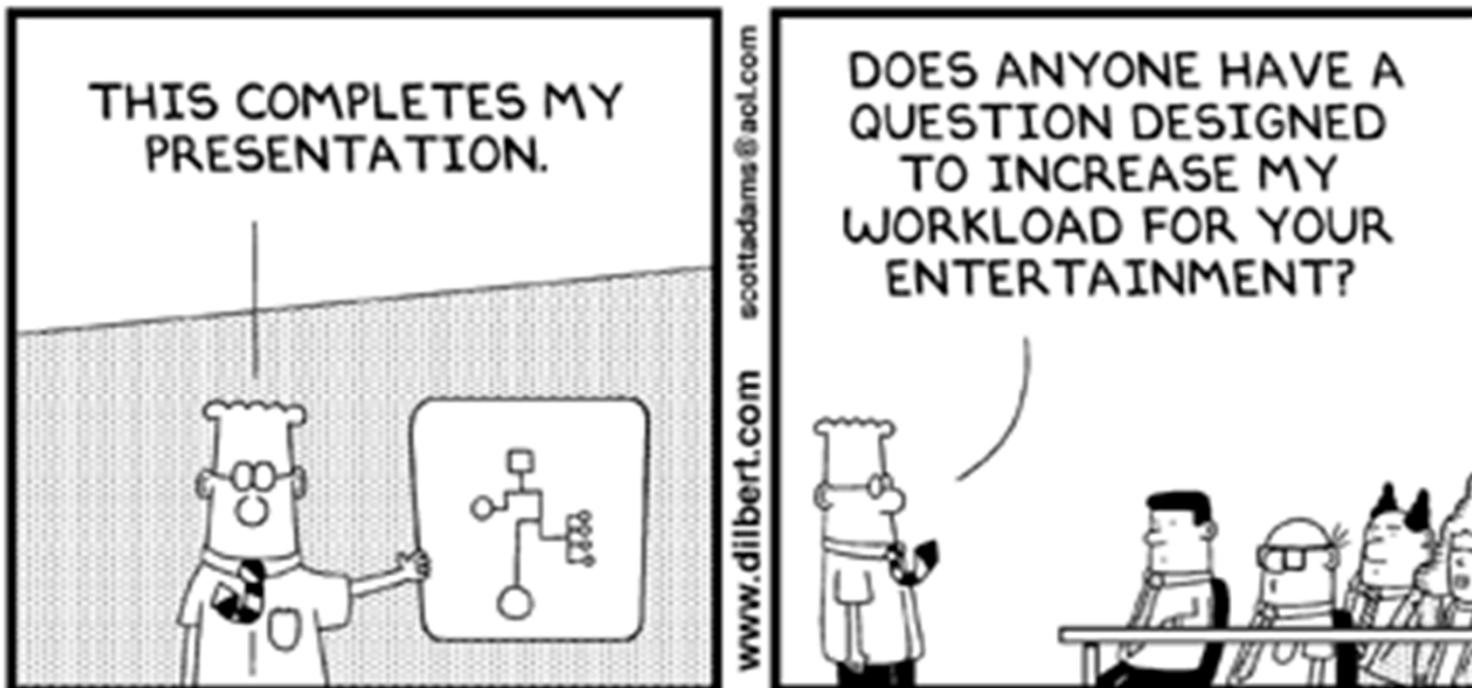
- **Sexiness doesn't scale** (Ask your girlfriend/boyfriend...)
 - Investment in good old-fashioned coding (libraries)
- **Cross-platform can be tricky**
- **Virtual machines are great**

- **We've made a system to test Qt GUIs, designed for multi-platform**
 - **Scalable**
 - # tests
 - # machines
 - **General**
 - **Applicable to testing other Qt GUIs (e.g. Unicos)**
 - **Useful (!)**
 - **Reduced manual effort (time!) to release each new version of the Fw**
 - **Steady trickle of errors found (1..2 per month). (Pays off!)**

- **As for Froglogic's Squish:**
 - We love it, but (like with a husband or a wife) you've got to be prepared to work within (and on) the relationship



Questions?



© Scott Adams, Inc./Dist. by UFS, Inc.