7th UCAAT

User Conference on Advanced Automated Testing

ETSI

TESTING EMBEDDED CONNECTIVITY COMPONENTS

IN A MEDICAL REGULATED CONTEXT

Presented by Ron Jaegers

PHILIPS

Innovation Services

Contents

About Me

- Ron Jaegers (ron.jaegers@philips.com)

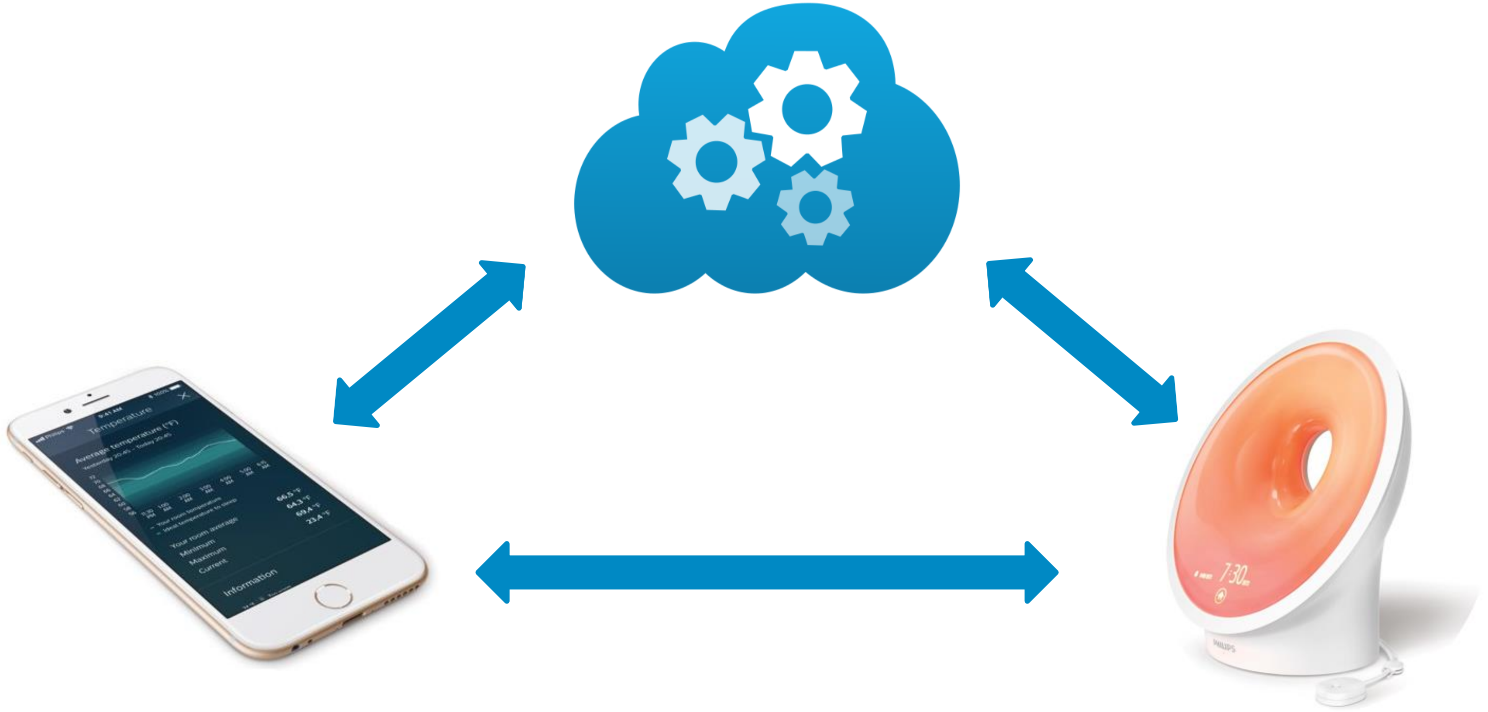
- Embedded Development Engineer @ Philips Innovation Services
- Father of three
- French car tinkerer
- Open-source enthusiast

PROJECT CONTEXT

Connectivity Platform

- Electronic Systems & IoT *by Philips Innovation Services* | Connectivity Center of Expertise
- Connectivity Platform
  - WiFi and BLE Connectivity Nodes
  - Mobile phone libraries

Connectivity Platform Triangle



Connectivity Platform Customers

Medical Context

- **Key values:** re-useable, traceable, automated

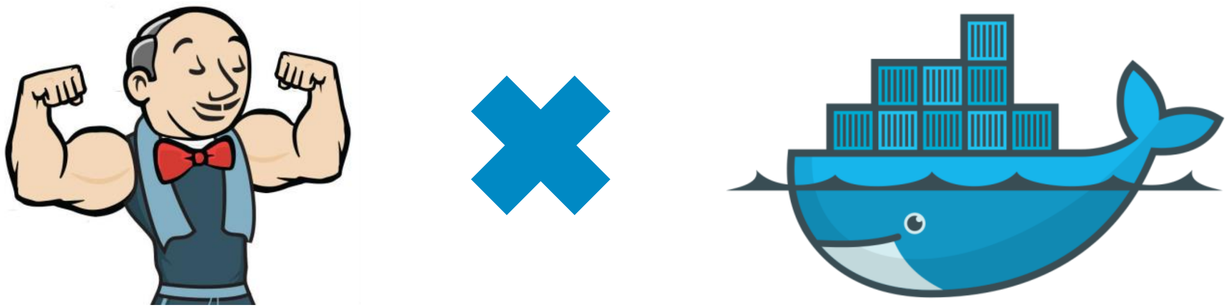
- No intended use…
- But not SOUP either…

COMPONENT LEVEL TESTS

Build and Test Setup

- **Key values**: scalable, fast, reproducible

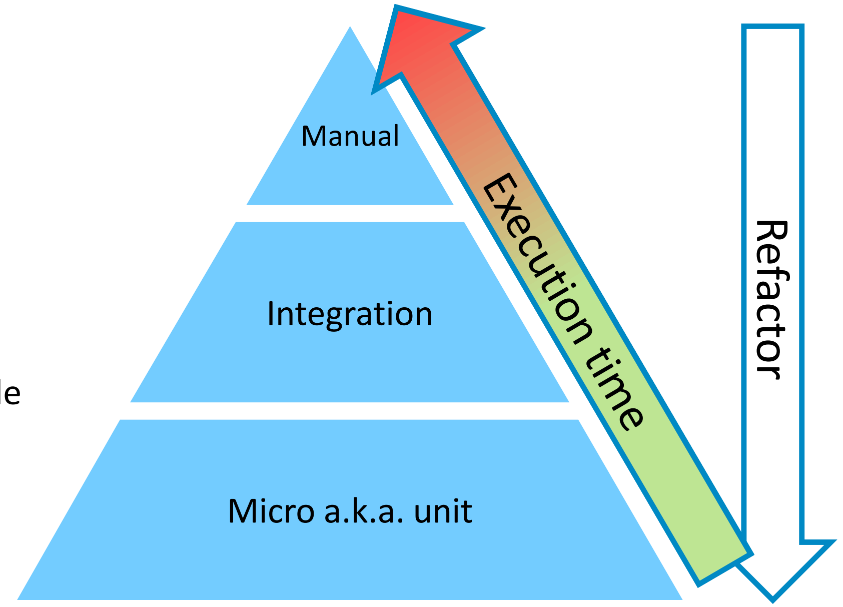
- Dockerized Jenkins master and slave architecture
- All configuration under source control
- Supporting multiple configuration permutations
- CI build + micro tests + smoke tests on five embedded platforms: 6 minutes!



Component Test Pyramid

- *Flexibility* **declines** from base to top
- *Reliability* **declines** from base to top
- *Execution time* **increases** from base to top

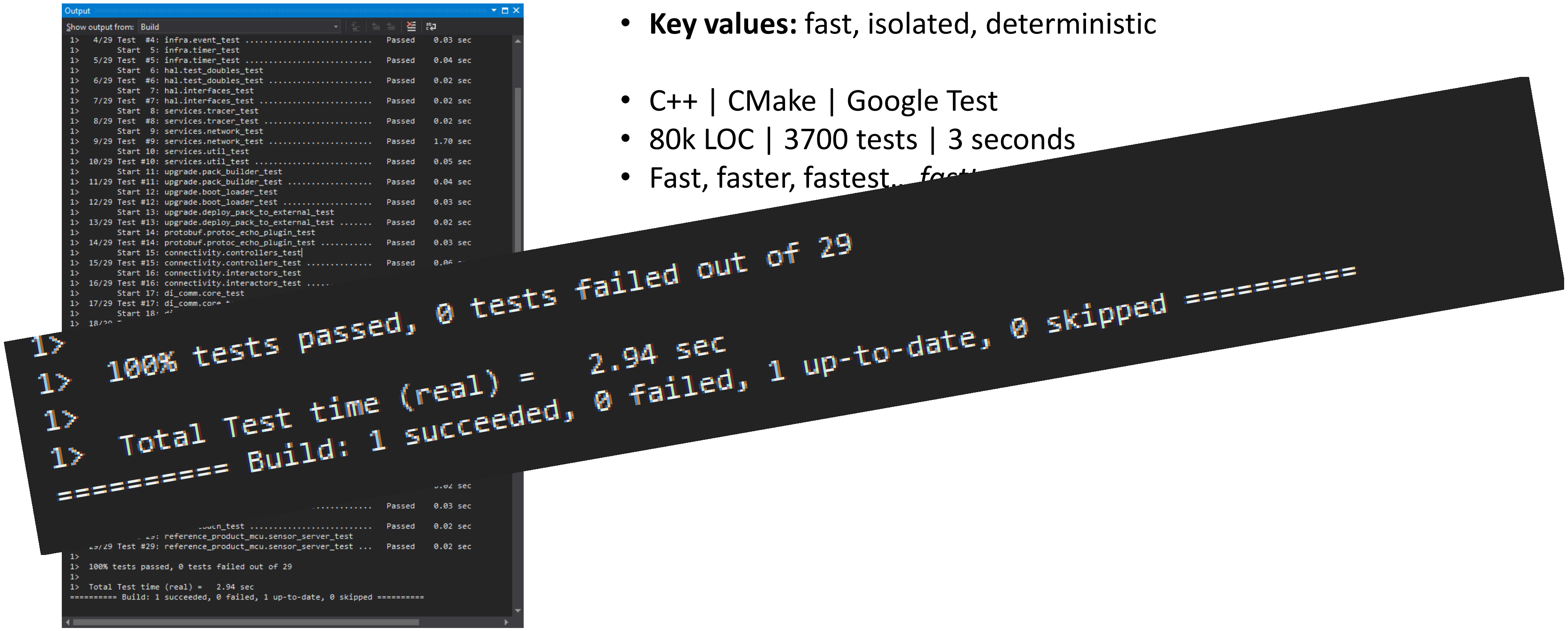
- Strive to push down tests by refactoring where possible



Enabler: EmbeddedInfraLib

- C++, heap-less, STL-like, library for embedded devices
- Features:
  - Hardware Abstraction Layer (HAL)
  - Asynchronous event mechanism
  - Containers and streams
  - Network layer
- Published to GitHub: https://github.com/philips-software/embeddedinfralib

Micro Tests a.k.a. Unit Tests



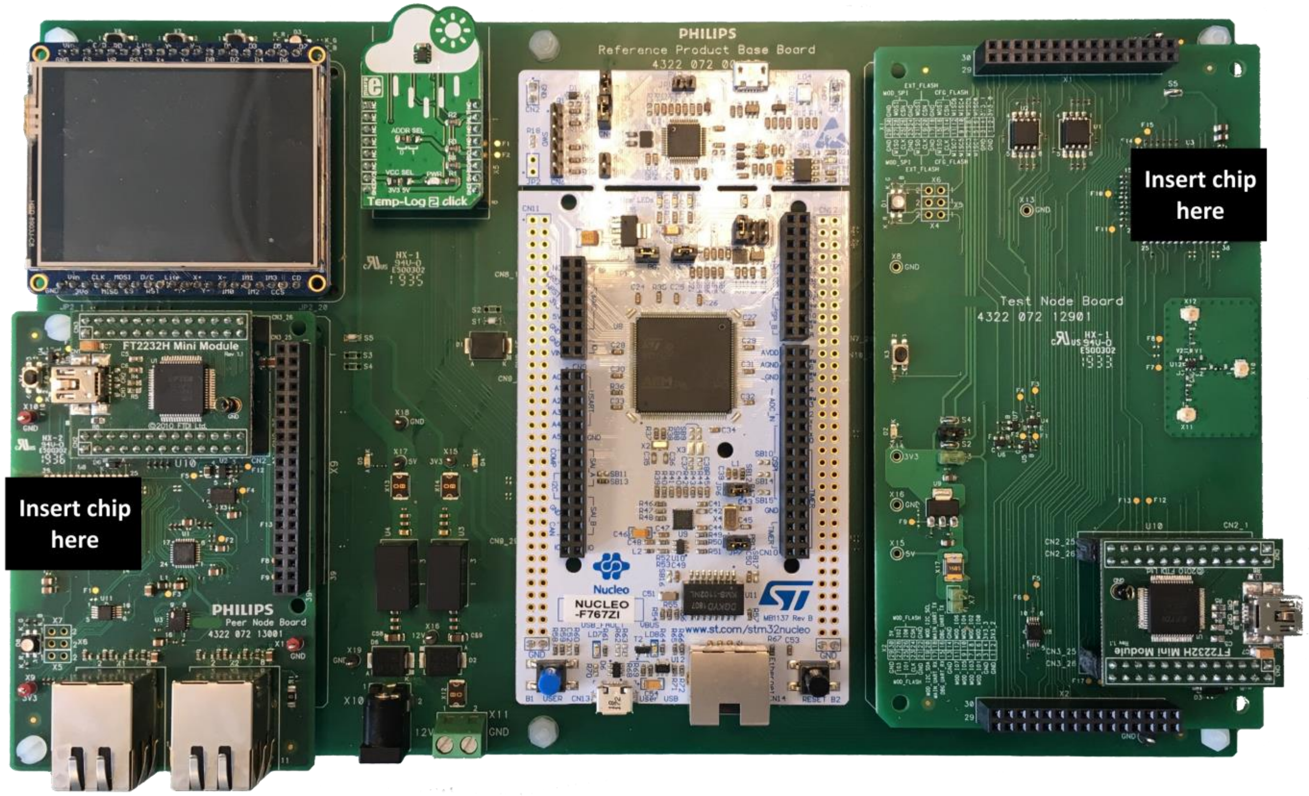
```
1>   4/29 Test  #4: infra.event_test ..........................   Passed    0.03 sec
1>        Start  5: infra.timer_test
1>   5/29 Test  #5: infra.timer_test ..........................   Passed    0.04 sec
1>        Start  6: hal.test_doubles_test
1>   6/29 Test  #6: hal.test_doubles_test .....................   Passed    0.02 sec
1>        Start  7: hal.interfaces_test
1>   7/29 Test  #7: hal.interfaces_test .......................   Passed    0.02 sec
1>        Start  8: services.tracer_test
1>   8/29 Test  #8: services.tracer_test ......................   Passed    0.02 sec
1>        Start  9: services.network_test
1>   9/29 Test  #9: services.network_test .....................   Passed    1.70 sec
1>        Start 10: services.util_test
1>  10/29 Test #10: services.util_test ........................   Passed    0.05 sec
1>        Start 11: upgrade.pack_builder_test
1>  11/29 Test #11: upgrade.pack_builder_test .................   Passed    0.04 sec
1>        Start 12: upgrade.boot_loader_test
1>  12/29 Test #12: upgrade.boot_loader_test ..................   Passed    0.03 sec
1>        Start 13: upgrade.deploy_pack_to_external_test
1>  13/29 Test #13: upgrade.deploy_pack_to_external_test .......   Passed    0.02 sec
1>        Start 14: protobuf.protoc_echo_plugin_test
1>  14/29 Test #14: protobuf.protoc_echo_plugin_test ..........   Passed    0.03 sec
1>        Start 15: connectivity.controllers_test
1>  15/29 Test #15: connectivity.controllers_test .............   Passed    0.06
1>        Start 16: connectivity.interactors_test
1>  16/29 Test #16: connectivity.interactors_test .....
1>        Start 17: di_comm.core_test
1>  17/29 Test #17: di_comm.core
1>        Start 18:
1>  18/
                                                                            sec
                                                ............   Passed    0.03 sec
                   _test .........................   Passed    0.02 sec
                 : reference_product_mcu.sensor_server_test
    /29 Test #29: reference_product_mcu.sensor_server_test ...   Passed    0.02 sec
1>
1>  100% tests passed, 0 tests failed out of 29
1>
1>  Total Test time (real) =   2.94 sec
========== Build: 1 succeeded, 0 failed, 1 up-to-date, 0 skipped ==========
```

```
1>
1>  100% tests passed, 0 tests failed out of 29
1>
1>  Total Test time (real) =   2.94 sec
========== Build: 1 succeeded, 0 failed, 1 up-to-date, 0 skipped ==========
```

- **Key values:** fast, isolated, deterministic

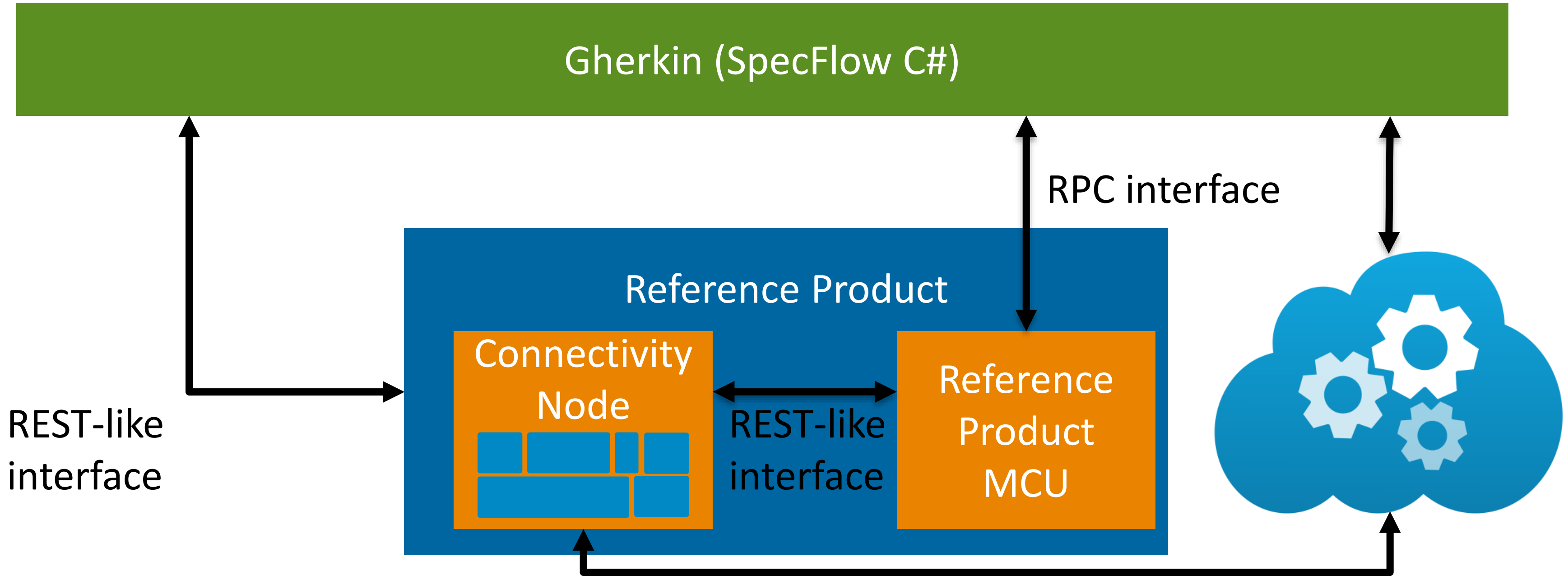
- C++ | CMake | Google Test
- 80k LOC | 3700 tests | 3 seconds
- Fast, faster, fastest… *fa*

Enabler: Reference Product



Integration Tests

- C# | CMake | Gherkin (SpecFlow)
- 200 integration tests, testing on feature level; tracing to requirements
- Runtime up to several hours



Manual tests

- Part of release process
- Hard to automate and destructive tests
- 2 for WiFi | 4 for BLE

PLATFORM LEVEL TESTS

Platform Level Tests

- Compatibility & interoperability
  - Between platform components
  - Between platform & phones
- Pen-testing

RELEASING A PLATFORM

Releasing a Platform

- Requirements & test-cases in ALM tooling
- Evidence exported from CI environment towards ALM tooling
- Documentation is:
  - (Automatically) generated
  - Reviewed
  - Signed-off & Published