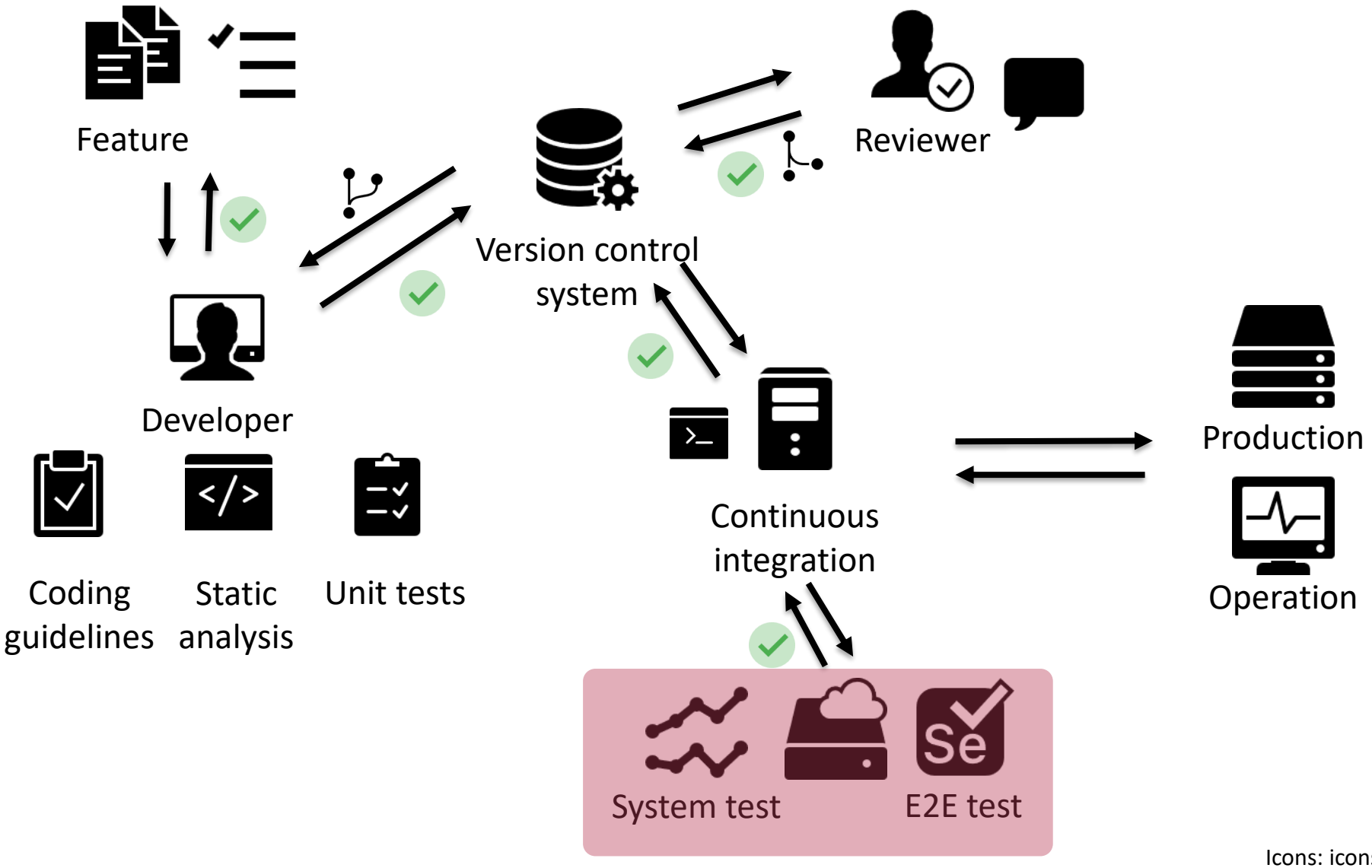
Test levels
Test automation

Zoltán Micskei, István Majzik

Department of Measurement and Information Systems

Overview



Icons: icons8.com

TEST LEVELS

Characteristics of tests in different levels

Recommendations from *How Google Tests Software*:

	Small	Medium	Large
Execution time	< 100 ms	< 1 sec	As fast as poss.
Time limit (kill)	1 minute	5 minutes	1 hour

Resource	Small	Medium	Large
Network (socket)	Mocked	only localhost	Yes
Database	Mocked	Yes	Yes
File access	Mocked	Yes	Yes
System call	No	Not recommended	Yes
Multiple threads	Not recommended	Yes	Yes
Sleep	No	Yes	Yes
System properties	No	Yes	Yes

Integration testing

Testing the **interactions** of modules

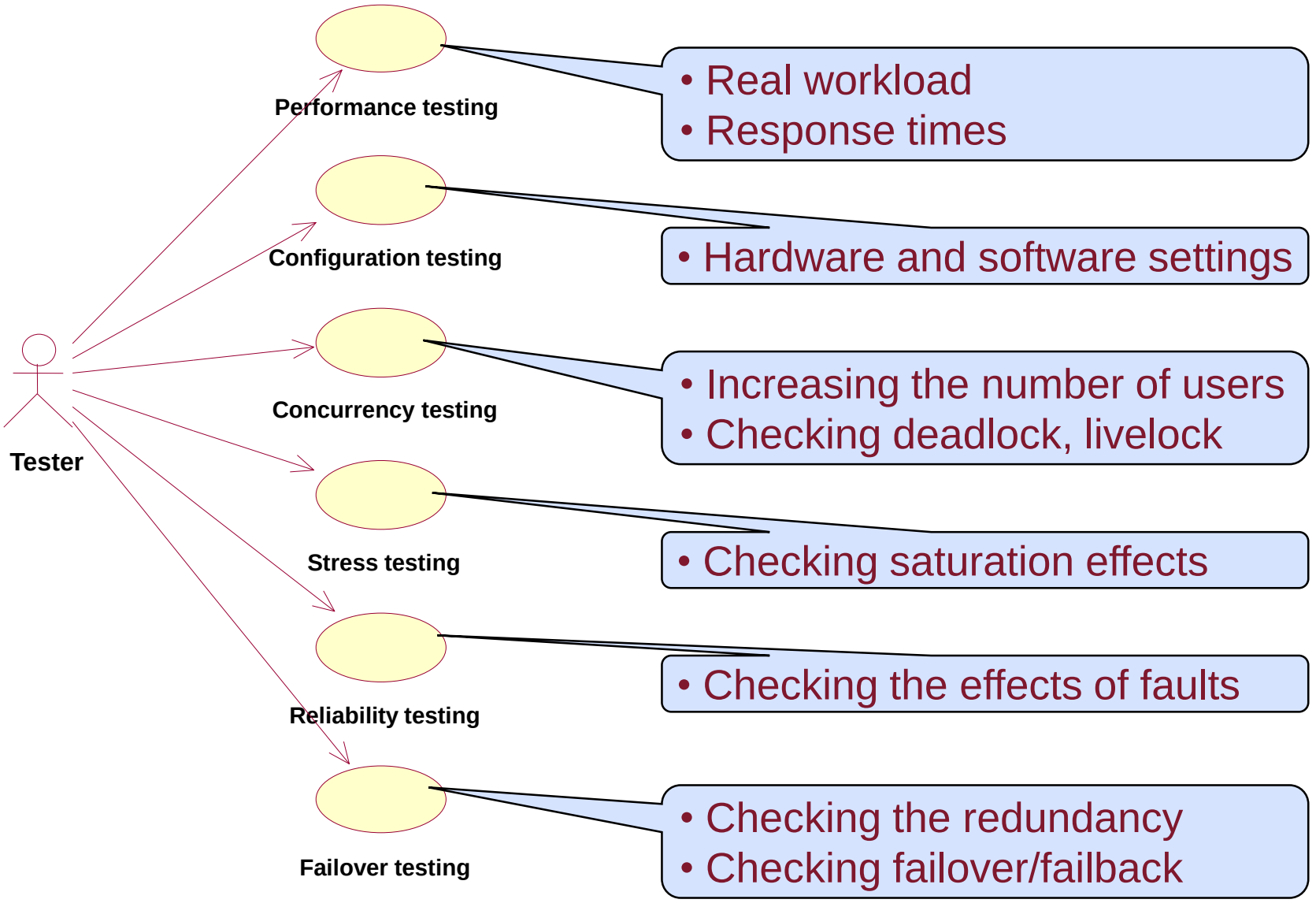
- Motivation
  - The system-level interaction of modules may be incorrect despite the fact that all modules are correct
- Methods
  - Functional testing: Testing scenarios
    - Sometimes the scenarios are part of the specification
  - (Structure based testing at module level)
- Approaches
  - “Big bang” testing: integration of all modules
  - Incremental testing: stepwise integration of modules

System testing

Testing on the basis of the system specification

- Characteristics:
  - Performed after hardware-software integration
  - Testing functional specification + testing extra-functional properties
- Testing aspects:
  - Data integrity
  - User profile (workload)
  - Checking application conditions of the system (resource usage, saturation)
  - Testing fault handling

Types of system tests



Validation testing

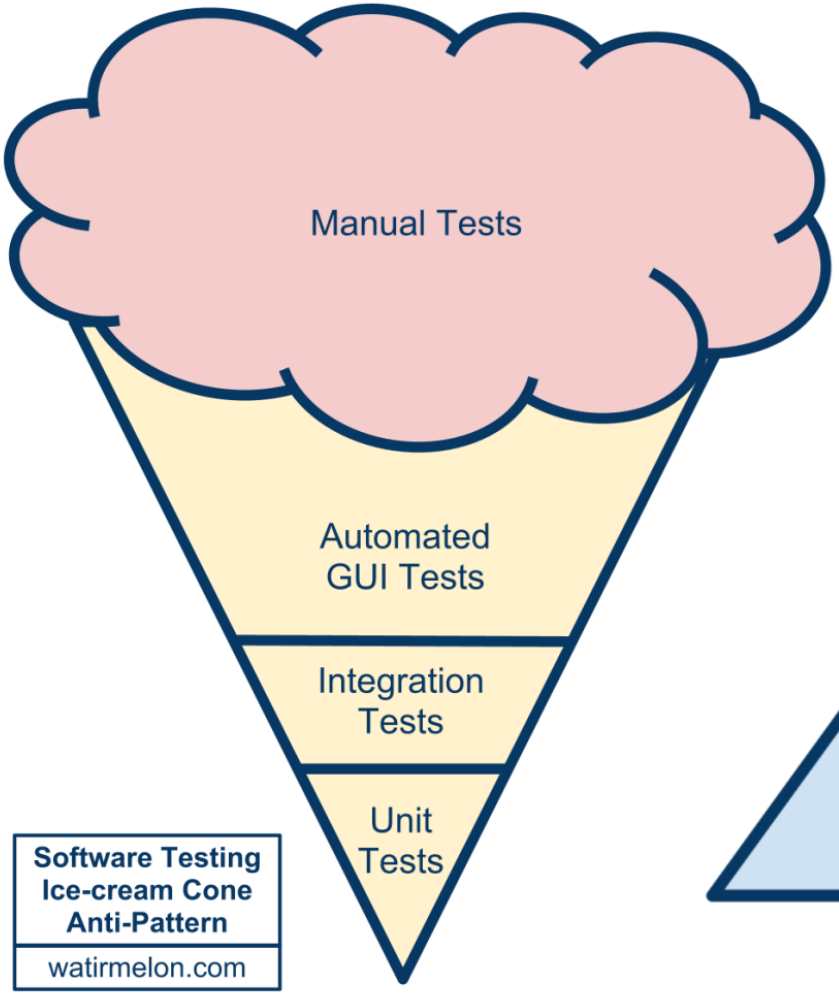
- Goal: Testing in real environment
  - User requirements are taken into account
  - Non-specified expectations come to light
  - Reaction to unexpected inputs/conditions is checked
  - Events of low probability may appear
- Timing aspects
  - Constraints and conditions of the real environment
  - Real-time testing and monitoring is needed
- Environment simulation
  - If given situations cannot be tested in a real environment (e.g., protection systems)
  - Simulators shall be validated somehow

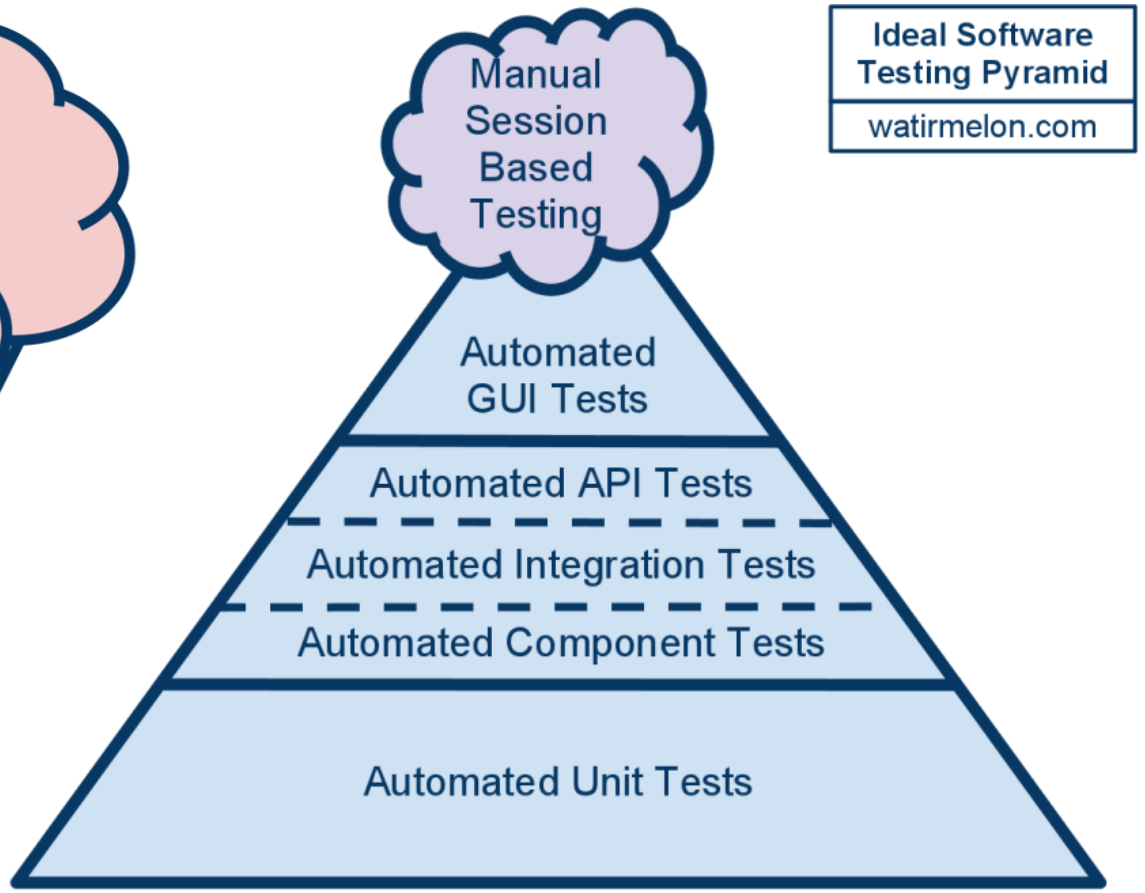
TEST AUTOMATION

Test automation?

- Automating test execution and/or evaluation
  - Manual could be slow/error-prone
- Manual or automated?
  - **Depends on lot of factors!**
  - Hard to automate
    - E.g. GUI, touch screen, printing…
  - ROI of automation
    - Cost, frequency of testing, lifetime of tests…
  - Accuracy
    - False positives

WHAT: Test pyramid





Source: Alister Scott

See also: Mike Cohn, Martin Fowler…

HOW: Test automation approaches

Capture/replay	• Easy to setup • Hard to maintain
Structured Scripting	• Script library (common actions) • Test logic and code not separated
Data-driven	• Test inputs/outputs extracted to external source (file, DB…)
Keyword-driven	• Tests: business/domain keywords • Automation code behind keyword
Model-based	• Test design is also automated

See: ISTQB syllabus

HOW: Steps in automated tests

Setup	• Get/compile latest version • Different hardware, platforms, OS… • Virtual machines: hosted or cloud
Execution	• Simple script / xUnit / custom framework • Detailed logging
Analysis	• Evaluating tests • Not trivial in integration/system level
Reporting	• 1000s of tests → too much information • Summary reports, analysis
Cleanup	• Resetting to a known, clean state • Goal: tests do not interfere with each other
Help	• Need to document tests code also • Test code and frameworks are part of the application

WHEN: Test execution strategies

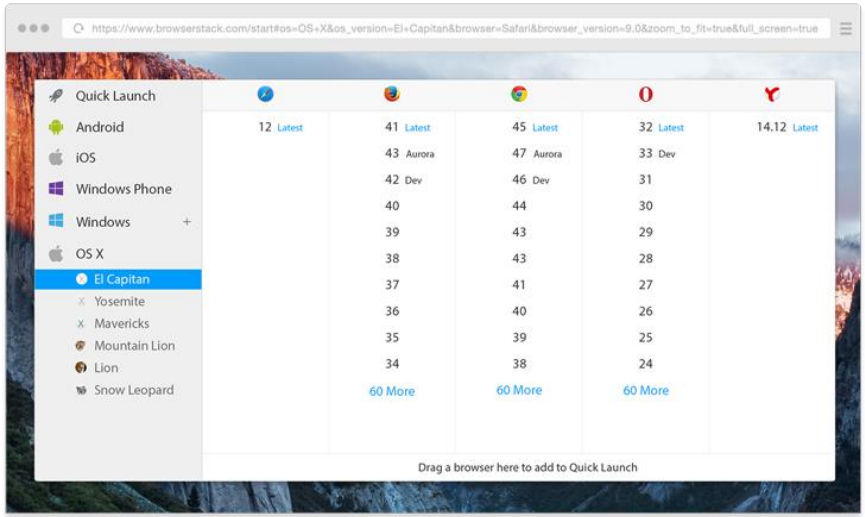
- Full (every tests)
  - At least before each release
- Smoke tests
  - Small test suite checking basic functionality
  - Quick feedback but limited accuracy
  - Many names, e.g. build verification test (BVT)
- Regression testing
  - Selective re-testing (test selection)
  - Test priorization

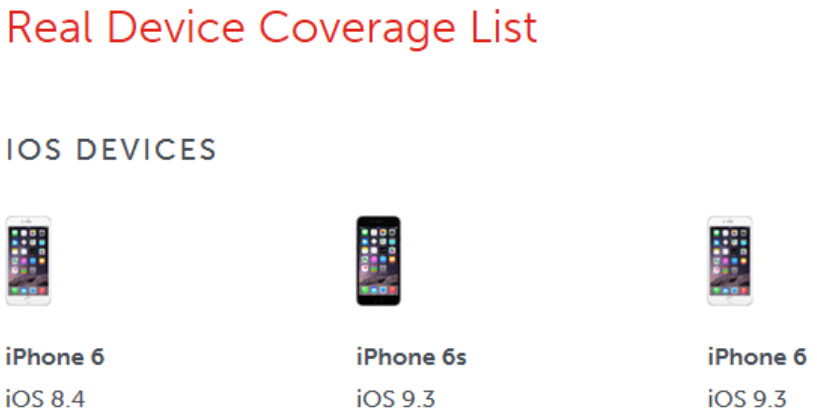
WHEN: Complete build and test workflow

- First steps
  - Pre-build, compile & build
  - Smoke tests
- Further steps (depends on build type)
  - Integration, system, E2E tests
  - Non-functional: performance, security (fuzzing)…
  - Static analysis
  - Manual testing…

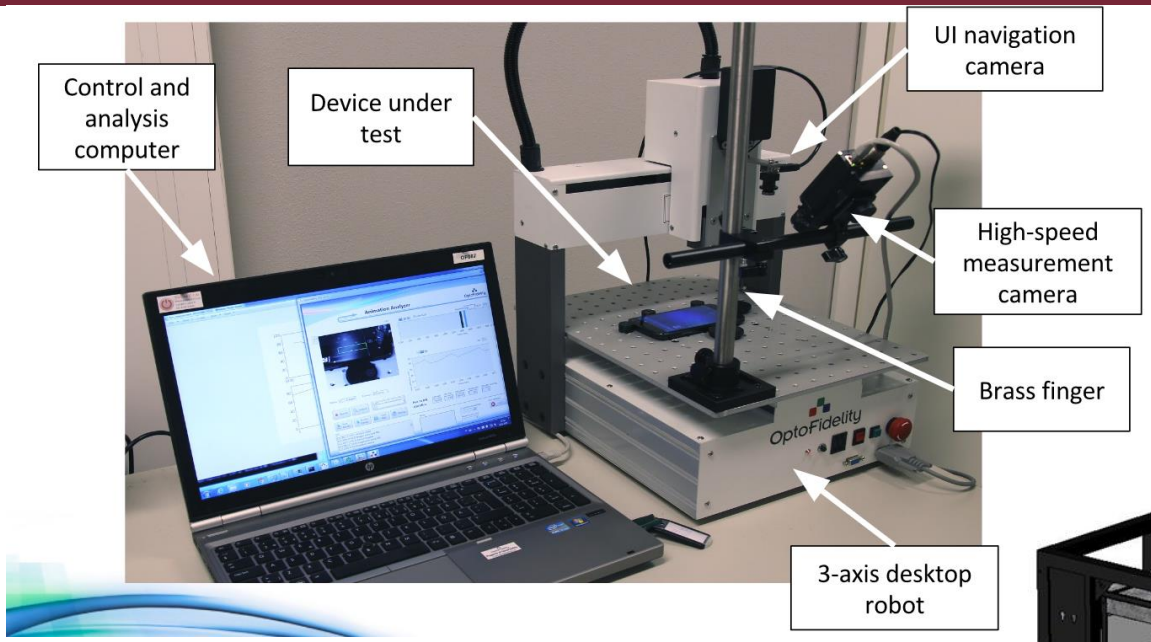
WHERE: Test execution platforms

- Web: browsers on different platforms
- Mobile: emulated or physical devices
- Many solutions
  - Hosted: Selenium, Robot framework…
  - Cloud: Browserstack, SauceLabs…

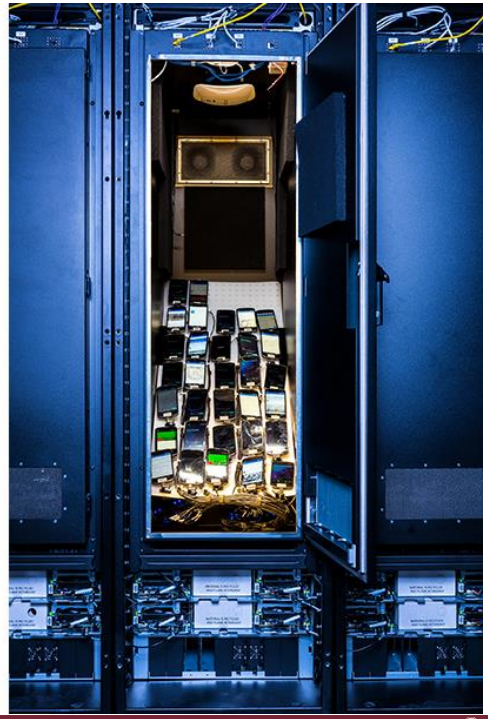




WHERE: Test labs (web and mobile)

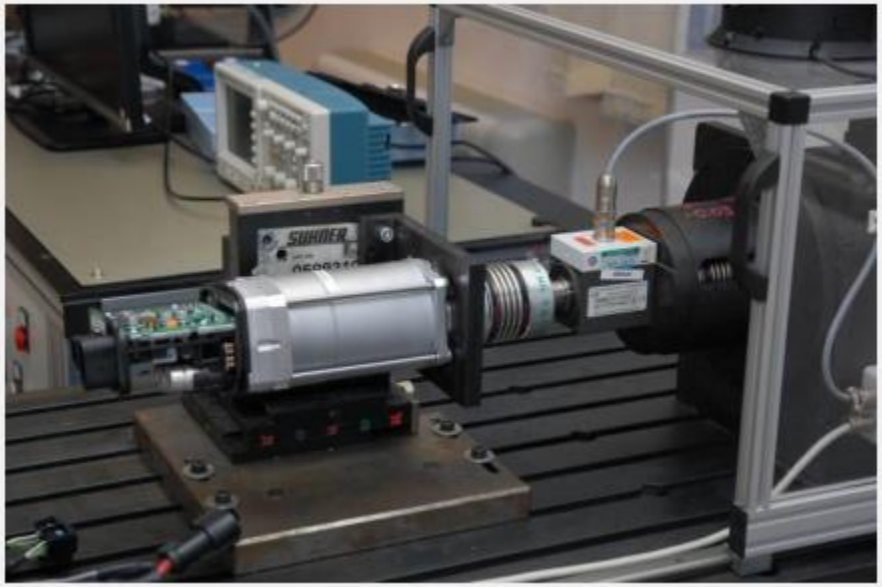


Robot Assisted Test Automation (GTAC 2015)



Mobile device lab

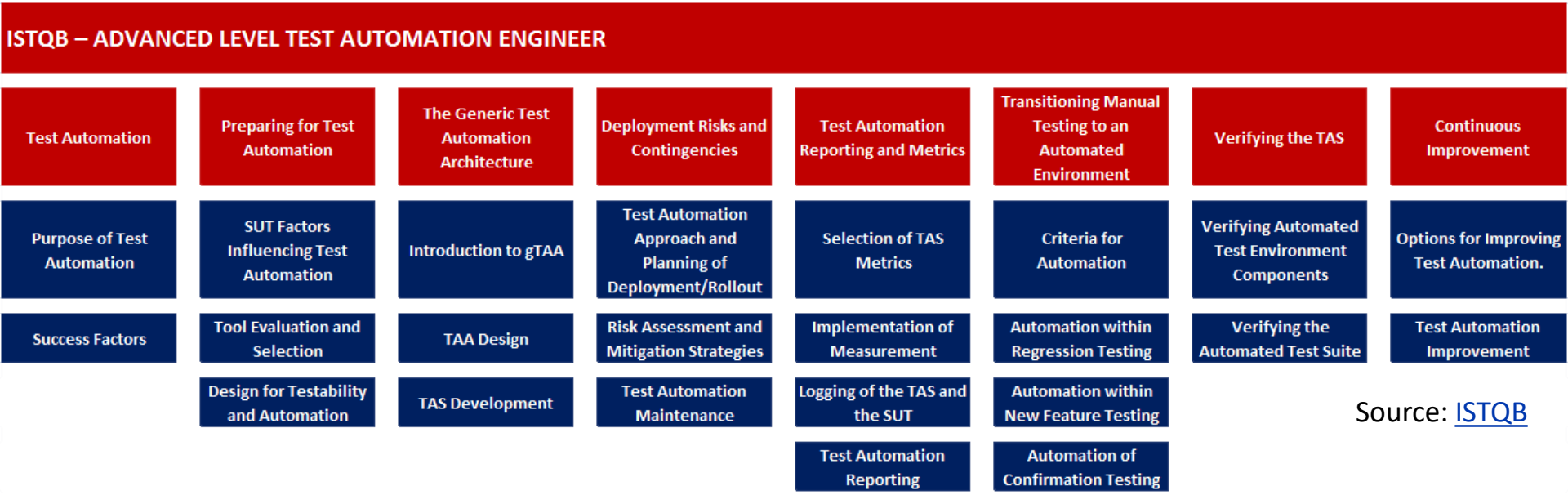
WHERE: Test labs (critical systems)



Functional test challenges in safety critical EPAS systems, ThyssenKrupp Presta (Test&Tea 2015)

Video and radar test, Bosch (Test & Tea 2015)

MORE: ISTQB Test Automation Engineer



Source: ISTQB

MORE: test automation conferences