Technical Software | Electronics | Industrial Mathematics | Remote Solutions

SIOUX
SOURCE OF YOUR TECHNOLOGY

Experiences of (Embedded) Test Automation

TESTNET

Bryan Bakker

TestNet thema avond 2014

bryan.bakker@sioux.eu
@Bryan_Bakker

SOURCE OF YOUR TECHNOLOGY

Contents

- Sioux
- Different in embedded testing
- Test automation aspects
- Case studies
- Summary

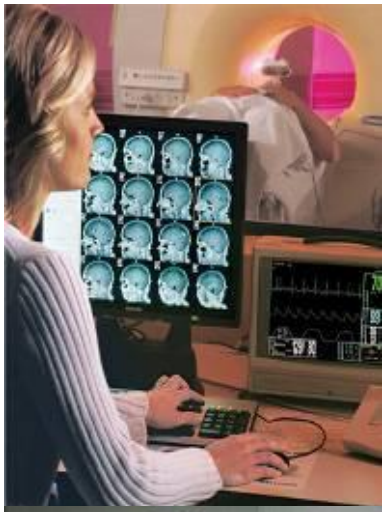
SOURCE OF YOUR TECHNOLOGY

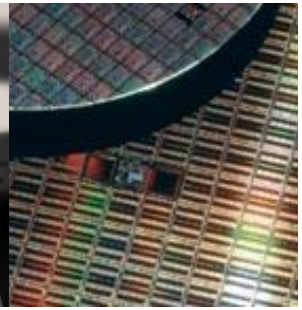
About Bryan Bakker

- Test Expert
- Certifications: ISTQB, TMap, Prince2
- Member of ISTQB Expert Level on Test Automation
- Tutor of several test related courses
- Domains: medical systems, professional security systems, semi-industry, electron microscopy
- Specialties: test automation, integration testing, design for testability, reliability testing

About Sioux

SOURCE OF YOUR TECHNOLOGY





SOURCE OF YOUR TECHNOLOGY

Examples of software failures

Toyota's troubles: A timeline

Since 2007, issues with brakes, floor mats and gas pedals have resulted in the recall of over 10 million vehicles and landed the Japanese carmaker in trouble.

FEB. 4 2010 — The automaker blames a software glitch for braking problems in its 2010 Prius. Link to story

Witnesses of the first of three Congressional hearings on Toyota's recall testify

Patriot Missile Failure

On February 25, 1991 during the Gulf W...
incomin
Accoun
Dhahra
since bo
was mul
particula
choppin
Patriot ba
choppin

Do not underestimate the impact of (defects in) software

Nancy G. Leveson,
Clark S. Turner, U

2007

MARCH 2007 —  Toyota receives reports about an accelerator pedal glitch in its Tundra model.

SEPT. 26 2007 —  Toyota recalls Lexus models to secure floor mats that could trap the gas pedal, causing the car to accelerate out of control. Link to story

2009

AUG. 28 2009 — A family driving a Lexus is killed after the gas pedal is caught under the floor mat.

SEPT. 29 2009 — The automaker issues a safety notice for 3.8 million vehicles due to the crash risk posed by the gas pedal becoming caught under the floor mat. Link to story

to criticize the equipment's manufacturer or anyone else. The mistakes that were

SIOUX
SOURCE OF YOUR TECHNOL

Yesterday

Satelliet in verkeerde baan door softwarefout

Door Marco van der Spek Datum: 9 sep 2014
Gepost in: Bugs

Twee Europese GPS-satellieten zijn in een verkeerde baan om de aarde geschoten, door fouten in de software van de Soyuz-draagraket. Onderzoek wijst uit dat fouten in de besturingssoftware van de verschillende trappen van de Russische Soyuz-raket debet zijn aan de mislukte lancering van twee Galileo-satellieten, de Europese tegenhanger van het Amerikaanse GPS-systeem. Dat melden bronnen aan de Russische site Izviestia, schrijft The Register.

Zowel de hardware als de software van de Soyuz-raket, veelgebruikt voor lancering van Europese kunstmanen, wordt gebouwd door het Russische technische instituut Pilyugin, kortweg NPTSAP. Alles werkte naar behoren, alleen kreeg de bovenste trap verkeerde instructies, waardoor de twee satellieten in een veel te lage en elliptische baan om de aarde raakten. ESA heeft de kunstmanen wel geactiveerd en onderzoekt of de loopbaan nog kan worden gecorrigeerd. Naast het Europese Galileo hebben ook de Russen (Glonass) en de Chinezen (BeiDou) een eigen alternatief GPS-systeem.

Bron: WebWereld

SOURCE OF YOUR TECHNOLOGY

Differences

- Not only software is developed, but also:
  - Mechanics
  - Electronics
  - Optics

    …. at the same time …
- Hardware behaves differently:
  - Wear
  - SW defect always in design <-> HW defect can be in manufacturing process
  - HW variations/imperfections (per batch, over time)

    → also caused by End of life and cost reductions

SIOUX
SOURCE OF YOUR TECHNOLOGY

Differences

- System test covers all disciplines
- Software testing is different from HW testing

- Integration is the most difficult part
- Defects often timing related
  - Hard to reproduce/analyze/fix/retest
- Safety
- Updates can be problematic
  - Automotive
  - Medical / Aircraft
  - Spacecraft
  - Mass products

SOURCE OF YOUR TECHNOLOGY

Access to system

- Access to SUT needed to:
  - Let SUT perform actions (test steps)
  - Retrieve information from SUT (test verification)
- “Default” approach:
  - Via User Interface
  - With standard tooling (record & playback)
- How to do when
  - No UI available for standard tooling?
  - UI automation is not desired?
    (e.g. maintainability)

SOURCE OF YOUR TECHNOLOGY

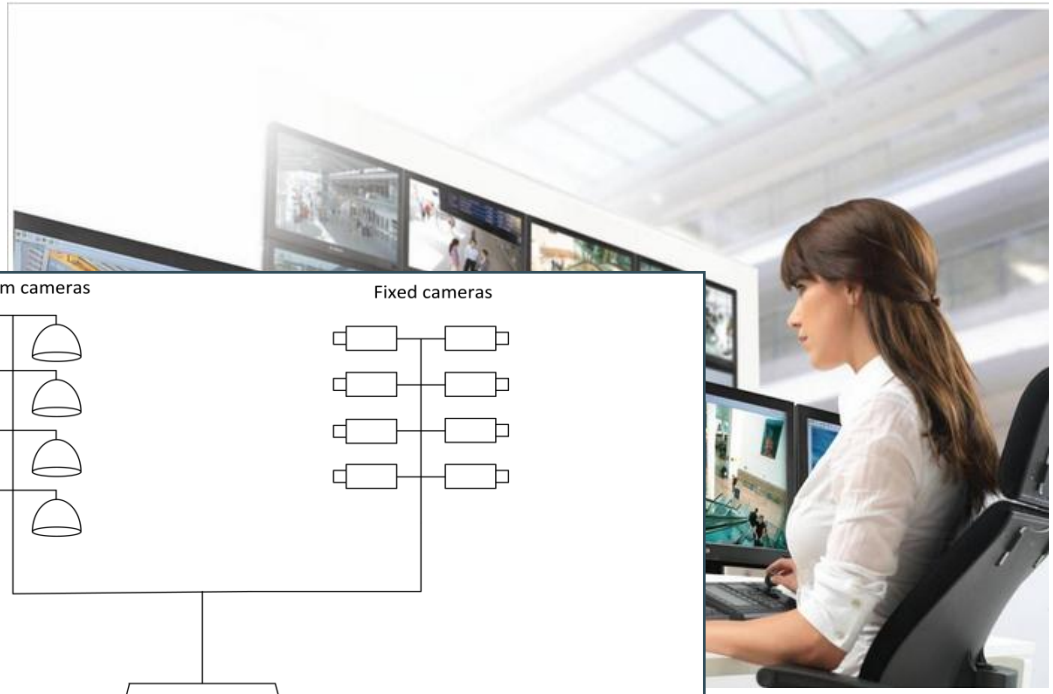
Access to system

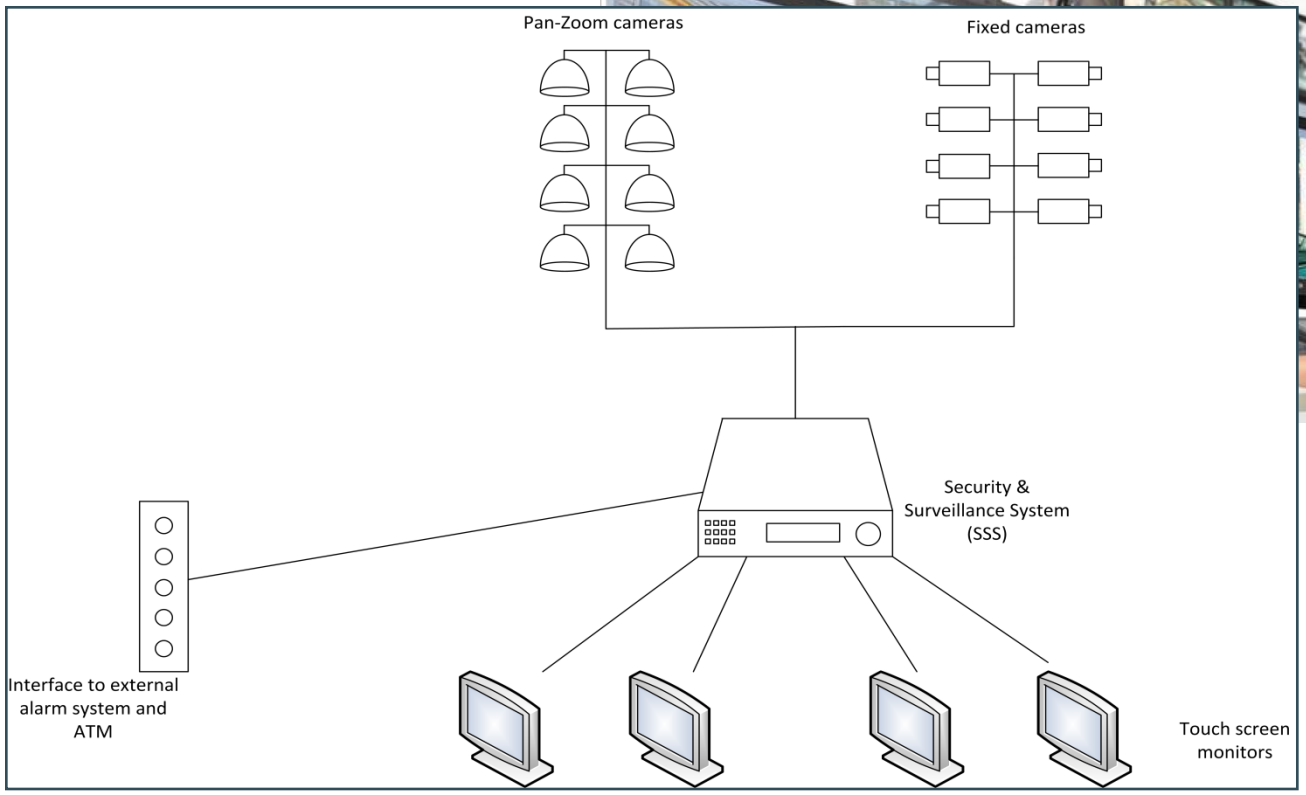
- There is no best way to interface with a SUT
- Depends on e.g.
  - Product
  - Technology
  - Project
  - (Test) Maturity of organization

  → Context

- Three examples… different approaches to test automation

Case study 1
Security & observation system





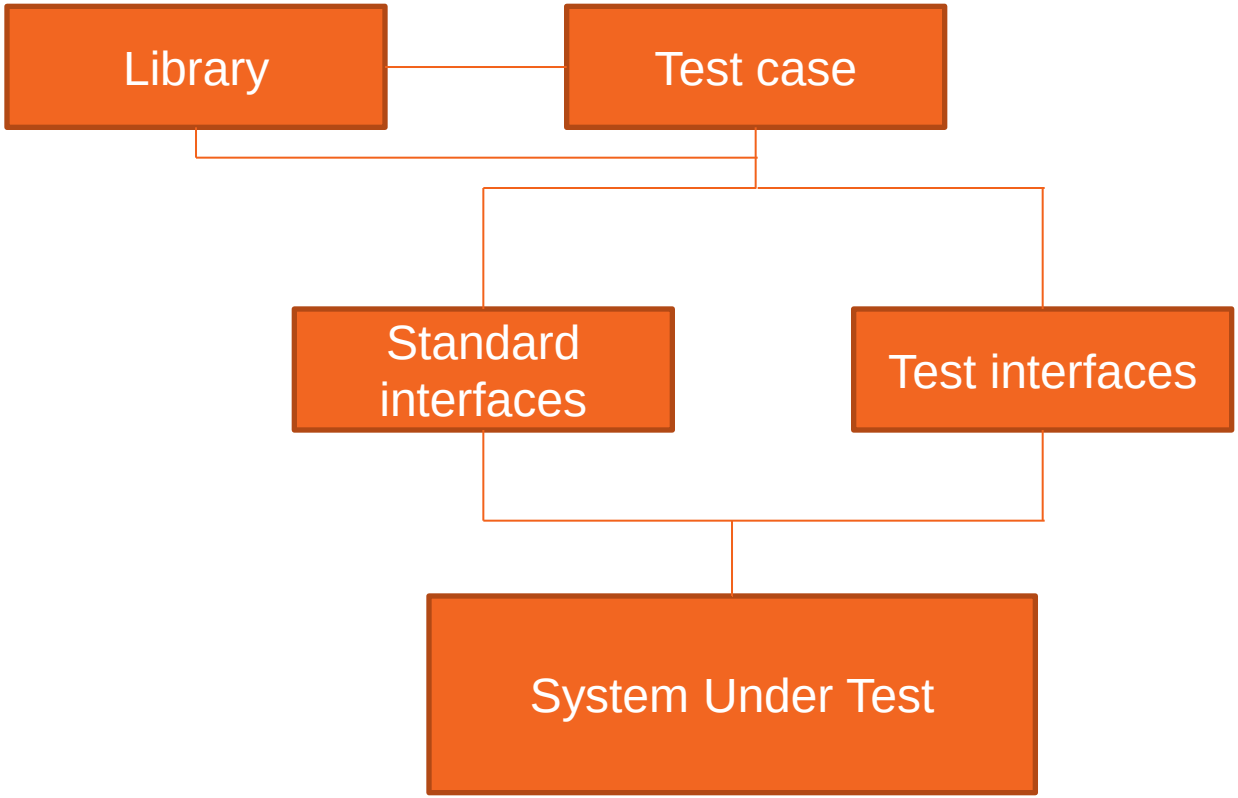
SOURCE OF YOUR TECHNOLOGY

Case study 1 Approach

- Existing Command Line Interface (via RS-232 and TCP/IP)
- For customers to build “own” applications
- Also used for test automation
- Standard functionality used
- Dedicated test interfaces added
  - “Pressing physical buttons” → as low as possible in the software architecture
  - Simulating external events: motion, alarms, video-loss
  - Simulation exceptional behavior: disk full, disk failure
  - Retrieving internal states, and variable information

SOURCE OF YOUR TECHNOLOGY

Case study 1 Approach



SIOUX
SOURCE OF YOUR TECHNOLOGY

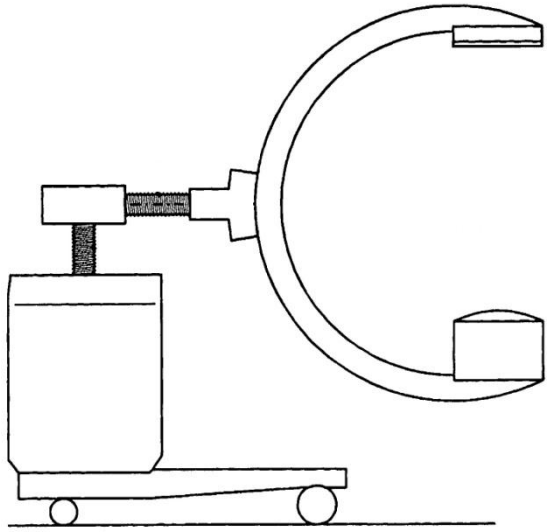
Case study 1 Approach

- Used for
  - Extensive regression testing
  - Performance testing
  - Reliability testing
- Results
  - Identified numerous defects
  - Frequent false alarms
    - Unreliable test case results (9x passed, 1x failed…?!?)
    - Defects in dedicated test interfaces
    - Defects which are not possible in the field
      - How to convince product owner to fix these issues…?
  - High probe effect!

Case study 2
X-Ray medical device

Medical Surgery Device:

- X-ray exposure + acquisition during surgery activities
- Real-time image chain
- Mobile device (frequently off/on)
- Quality and testing considered important in organization

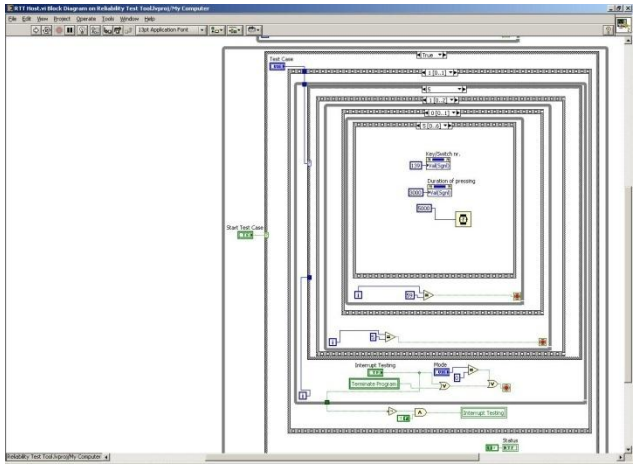


Reliability was an issue:

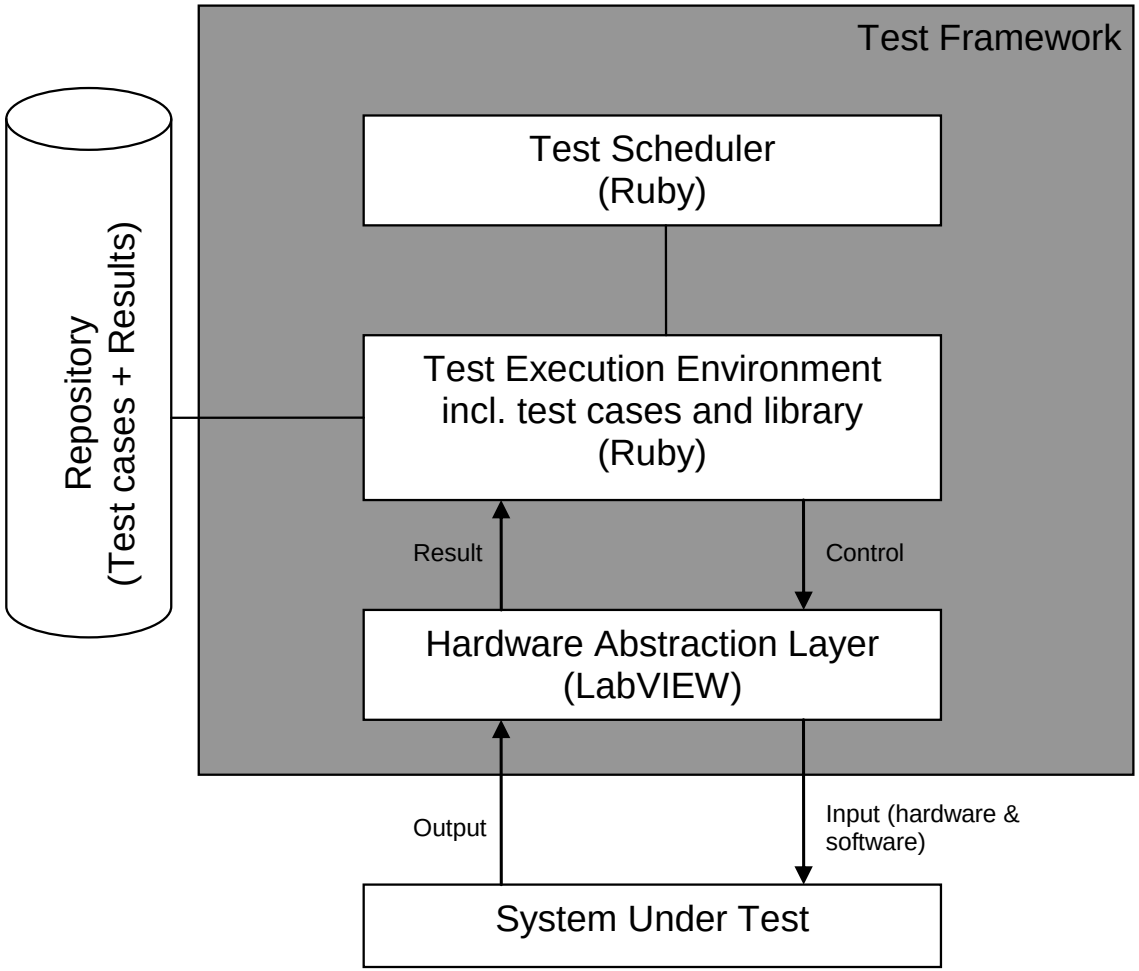
- “Frequent” startup failures
- Aborted acquisitions
- Always safe… but not reliable!

Case study 2 Approach

- Hardware interfaces used to invoke actions on SUT
  - Buttons on different keyboards
  - Handswitches
  - Footswitches
  - Different power-switches
- LabVIEW generates electrical signals
- Logfiles used for verification
- No software changes needed for this approach
- Later also extended with software test interfaces



Case study 2 Approach



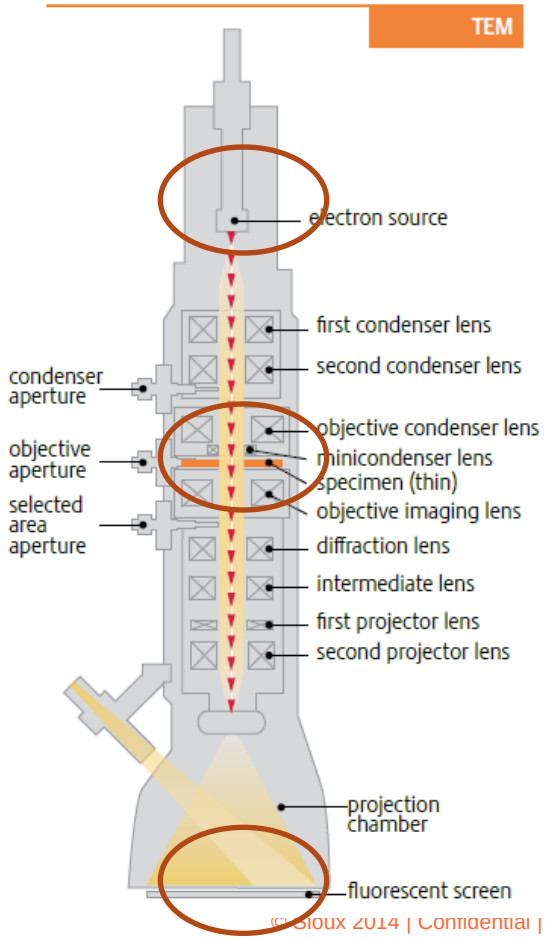
SOURCE OF YOUR TECHNOLOGY

Case study 2 Results

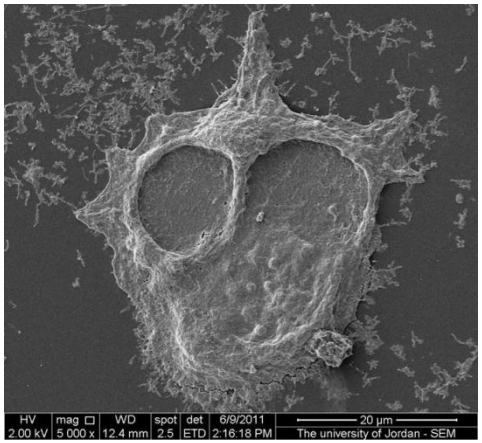
- Numerous reliability hits identified + solved
- Low probe effect (not a single false alarm)
- Easily ported to different products
- More projects wanted this approach
- Only 5 system test cycles remaining (was 15)
- LabVIEW layer (+dedicated hardware) developed by HW-Engineer
- LabVIEW complexity not part of test scripts

Case study 3
Electron microscope

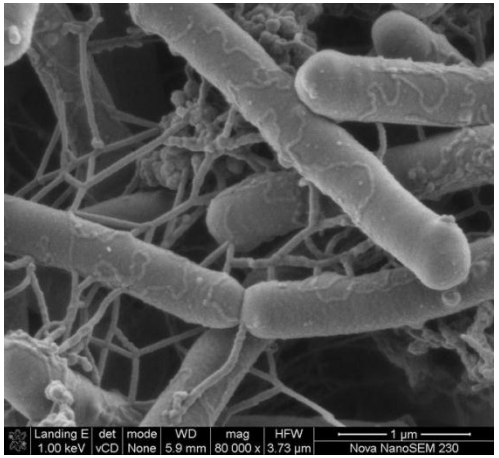
- World leader in electron microscopes
- Light microscope: 1000x → 200nm (limited by the wavelength of light)
- Electron microscope: 4Mx → 0.05 nm
- Nm = a billionth of a meter (10^{-9} meter)



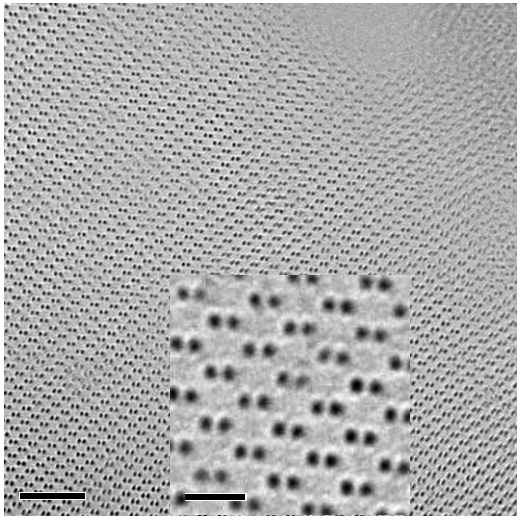
Case study 3
Electron microscope



Breast cancer cell.
Magnification 5.000x

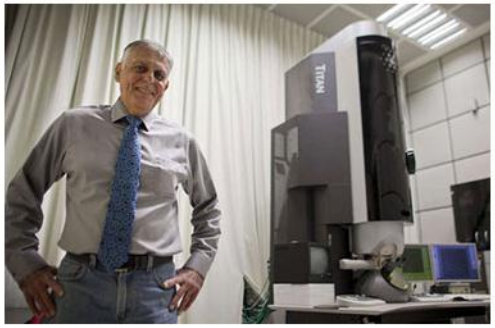


Salmonella bacteria.
Magnification 80.000x



Atomic structur of Ge (Germanium). Distance is 0.5 nm

Prof. Daniel Shechtman Wins Nobel Prize in Chemistry with Titan TEM



"The Titan Microscope, which is considered the most advanced of its field, is a high resolution electron microscope that can easily detect atoms and is used for discoveries."

- Professor Daniel Shechtman, Recipient of the 2011 Nobel Prize in Chemistry

SOURCE OF YOUR TECHNOLOGY

Case study 3 Approach

- First steps: quick and dirty
- Fast feedback
- Automation via GUI

Case study 3 Approach

- Vacuum SW redesign
- New HW items

- Test automation focus on vacuum

SIOUX
SOURCE OF YOUR TECHNOLOGY

Case study 3 Approach

- UI-Controls and graphics used for
  - Test actions
  - Test verifications
- Reliability tests
  - System tests on complete system
  - Executed for long time (days)
  - Not only software but whole system is tested
- “Machine fathers” were afraid of
  - Machine damage
  - Lost vacuum

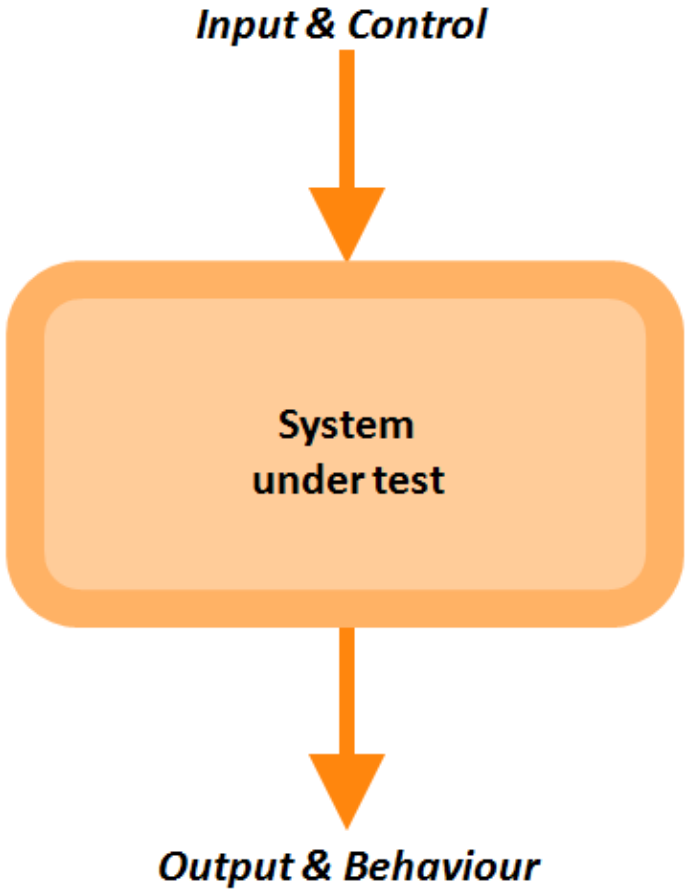
Case study 3 Results

- Low probe effect, although high probe effect was expected
- Almost no SW failures found
- HW failures identified within a few days
  - Excessive wear
  - Bad batches
  - Design flaws
- Note: Vacuum SW was modelled + generated
- Later also extended with dedicated test interfaces → phase out the UI automation

Level of Intrusion

- The level to which the SUT is adapted in order to make it possible to automate testing

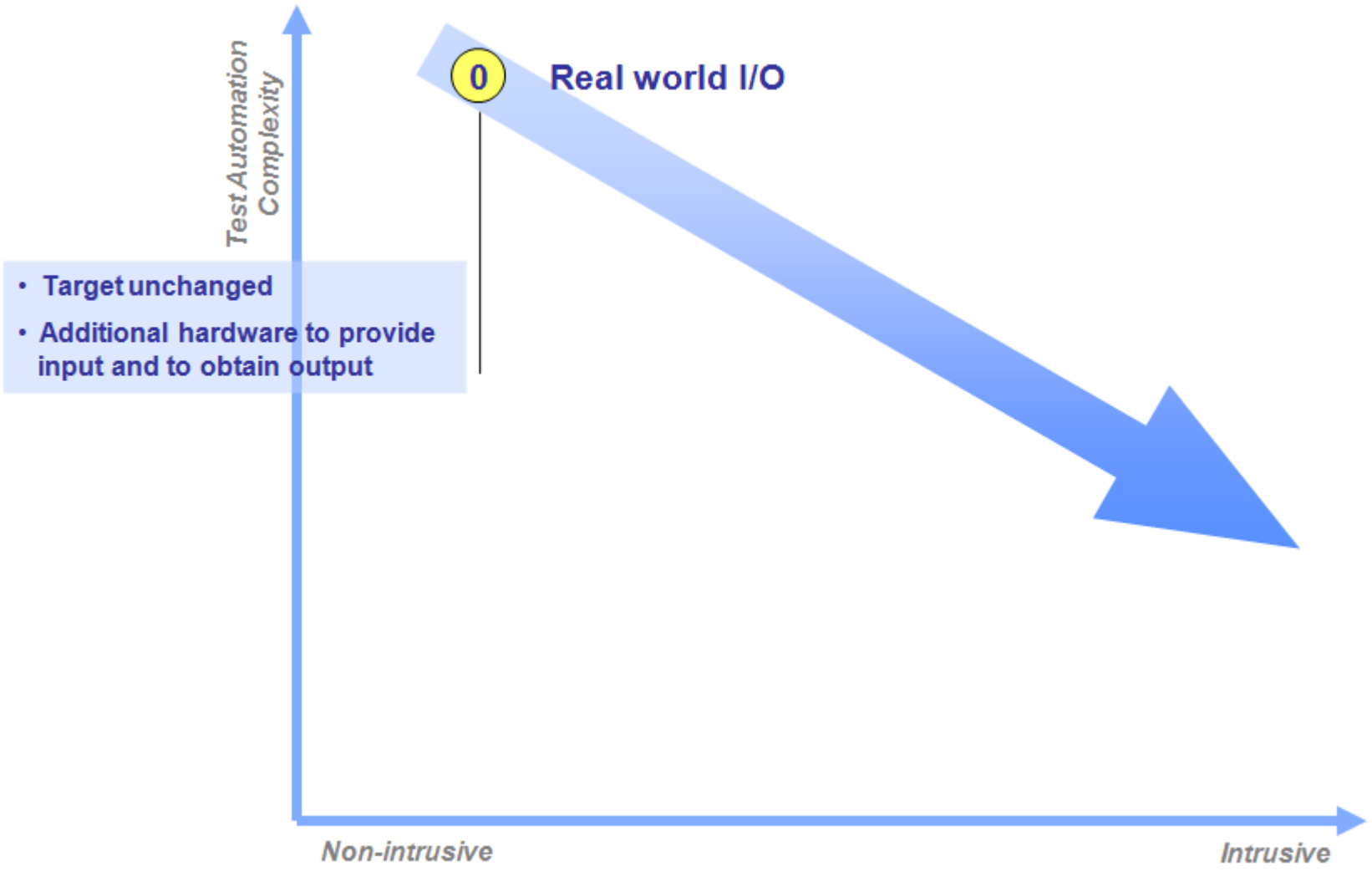
- Probe effect:
  - “unintended alteration in system behavior caused by measuring that system” (wikipedia)



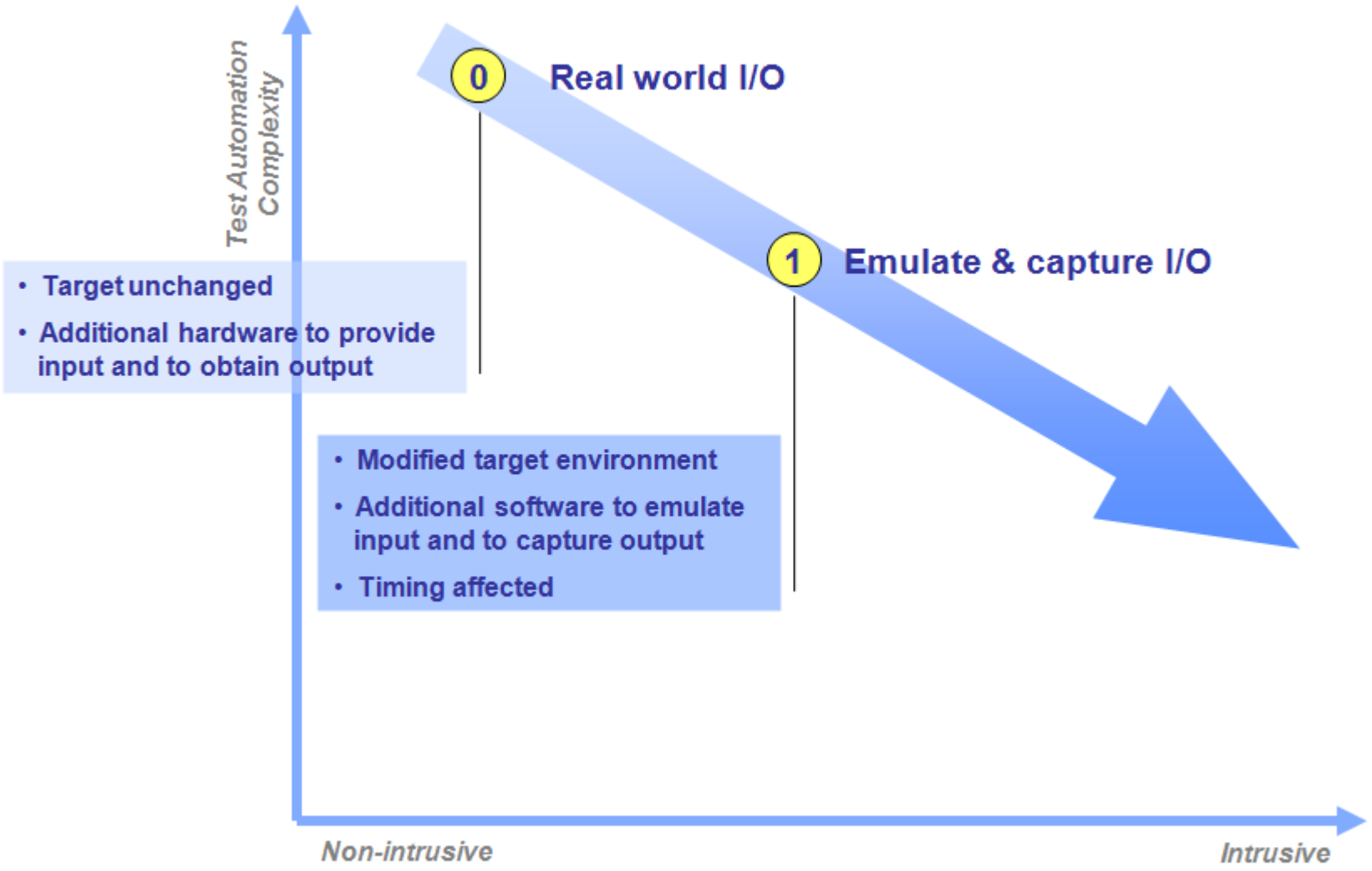
Level of Intrusion

SOURCE OF YOUR TECHNOLOGY

Level of Intrusion

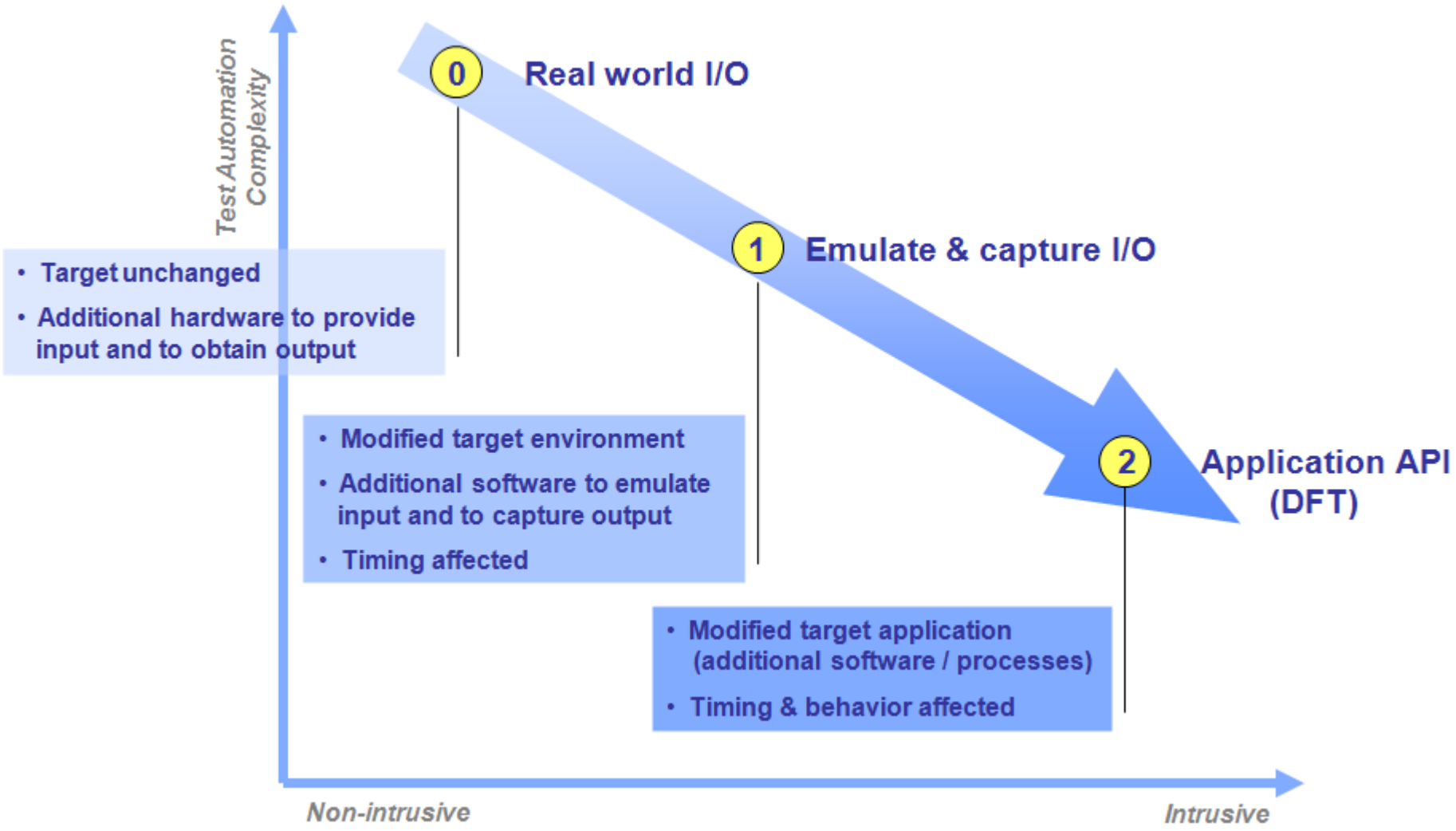


Level of Intrusion



SIOUX
SOURCE OF YOUR TECHNOLOGY

Level of Intrusion



SIOUX
SOURCE OF YOUR TECHNOLOGY

Conclusion

- Some differences
- Access to the SUT
- Test automation approach
  → context very important
- 3 examples with different approaches
  - All 3 approaches fitted in the current situation
  - Approach changed over time

SOURCE OF YOUR TECHNOLOGY

Questions

SOURCE OF YOUR TECHNOLOGY

www.sioux.eu

+31 (0)40 26 77 100

bryan.bakker@sioux.eu