

# Test Automation with Keywords

---

## A Case Study

*Thomas Mahler, Volkswagen AG*

*Alexandra Imrie, BREDEX GmbH*



Das Auto.



# Agenda

---

- Introductions
- Methods and Concepts
- Results

# Introductions

---

- EPUS project
  - Testing and control for measuring equipment
  - 5 man-years
  - Began January 2006
  - Testing began one month later
  - Handover in May 2007
- Thomas Mahler
  - Project lead for EPUS project
- Alexandra Imrie
  - Test consultant for EPUS project

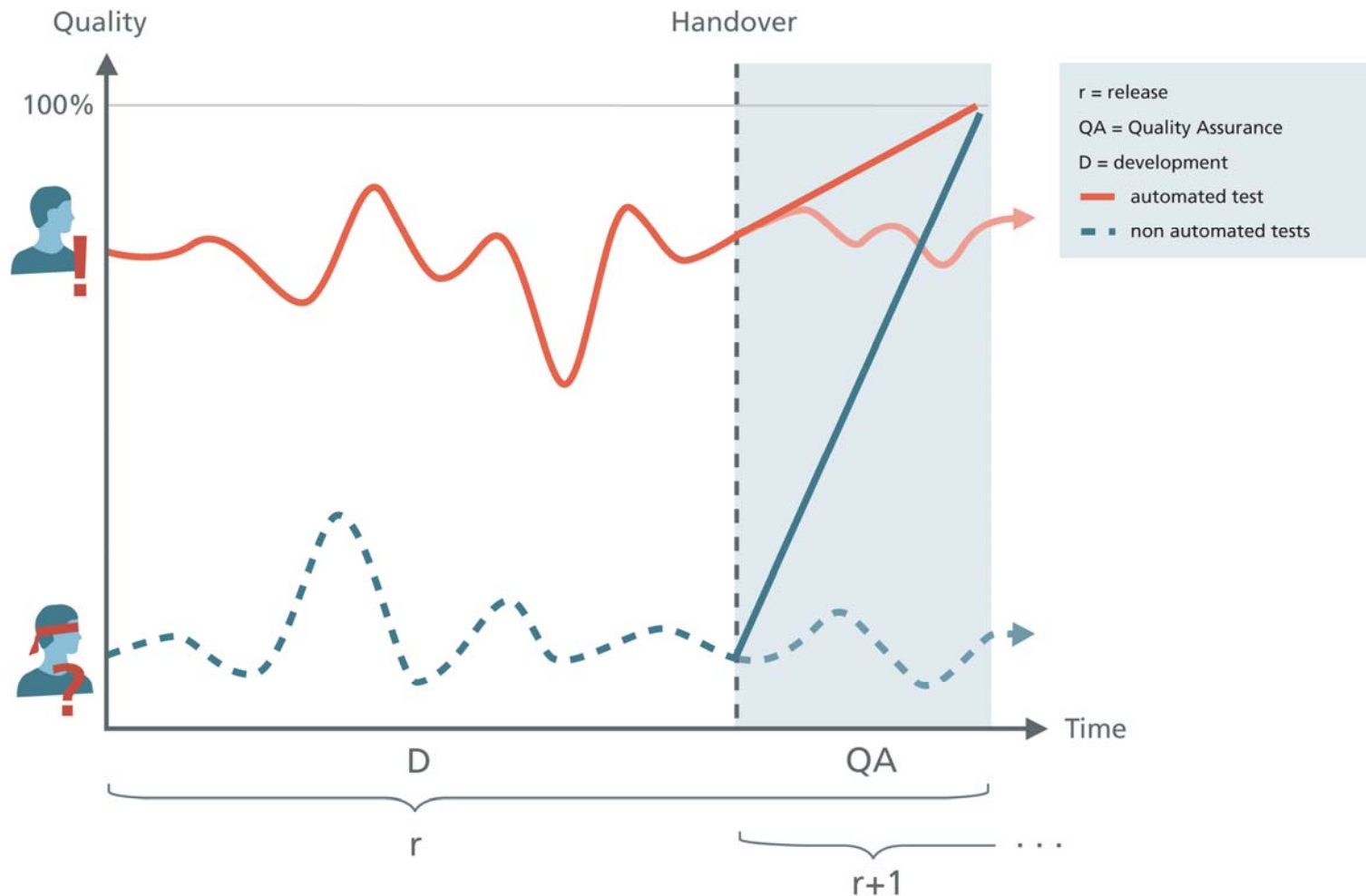


# *Methods and Concepts:* Motivation for automated testing

---

- Large project
  - Manual tests can't keep up
  - Good base of automatically tested areas
- Critical area of use
- (Finding errors as early as possible)

# Methods and Concepts: Making time for manual testing



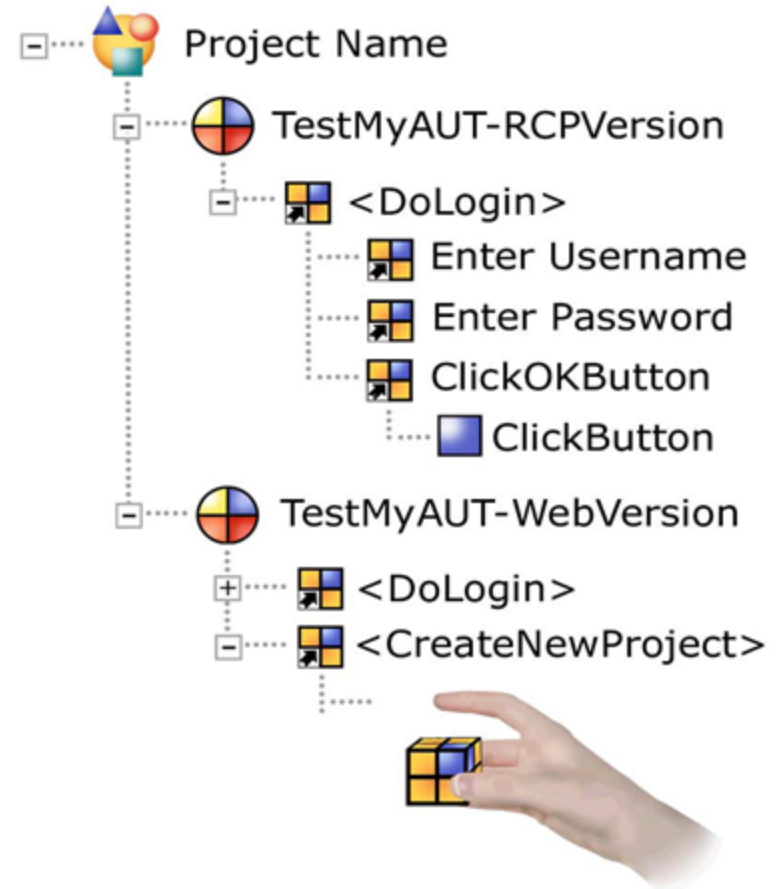
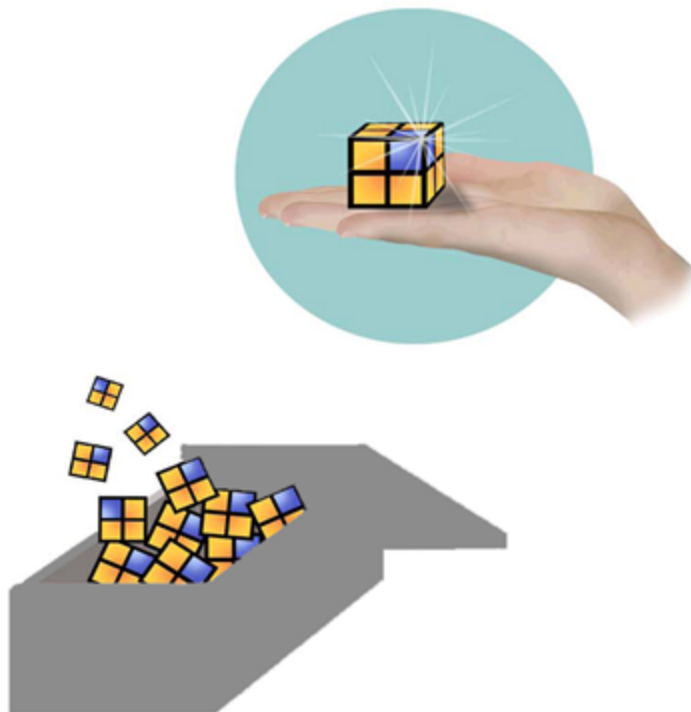


# *Methods and Concepts:* Motivation for keywords

---

- Prompt feedback
- Flexibility
- Modularity
- Focus on testing requirements
- No programming effort involved

# Methods and Concepts: Keywords



# *Methods and Concepts:*

## Keywords

---

- Reusable, flexible modules of test functionality
  - Component, Action, Parameter
- Based on same principles as software development
  - Reusability & maintainability
  - Combine small modules to make complex modules
- GUI tests well suited to keywords
  - Actions repeated throughout test





# *Methods and Concepts:* Keywords require planning

---

- Plan for reuse
- Plan for change
- Refactor as you go

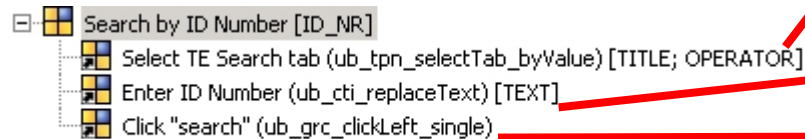
# *Methods and Concepts:* Good Keywords

---

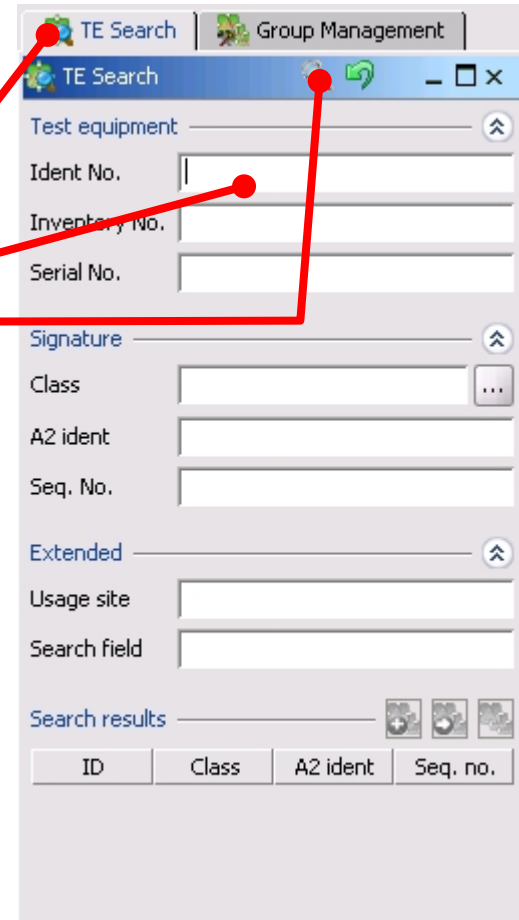
- Synchronization
  - Wait for application to start
    - Cues in the GUI →
    - Wait for component
    - Wait for main window
  - Wait for progress dialog to close
    - Wait for window to close

# Methods and Concepts: Good Keywords

- Search by ID-Nr
  - ID-Nr



- Makes sure correct tab selected
- Can also search for all equipment
- Working with results as separate keyword



# Methods and Concepts: Good Keywords

- Create new piece [Class, ID, Kind, Type...]

Test equipment

Ident No. <auto> Inventory No. Serial No. Modification date

Class A2 identification Seq. No. Modified by

Description Producer

Specification

⚠ The panel contains incomplete or invalid input data.

Master data | Criteria | Documents | Sampling master data

Master data

Properties Location

Kind Documentation required  Factory

Type Deadline monitoring  Storage location

Permanent user Automatically issue  Issuing office











Initial certification Transferstandard  Checkpoint

Search field Relocation date

Comment (long) Relocation site











# Methods and Concepts: Good Keywords

---

- ☐  Create new piece of test equipment [CLASS; IDENTIFIER; KIND; TYPE; FACTORY; CHECKPOINT]
  -  Click "new test equipment"
  -  Enter class [TEXT]
  -  Enter identifier [TEXT]
  -  Select "master data" tab [TITLE; OPERATOR]
  -  Enter kind [TEXT]
  -  Enter type [TEXT]
  -  Enter factory [TEXT]
  -  Enter checkpoint [TEXT]
  -  Click "save"

# Methods and Concepts: Good Keywords

---

- ☐  Create new piece of test equipment [CLASS; IDENTIFIER; KIND; TYPE; FACTORY; CHECKPOINT]
  -  Click "new test equipment"
  -  Enter class [TEXT] ←
  -  Enter identifier [TEXT] ←
  -  Select "master data" tab [TITLE; OPERATOR]
  -  Enter kind [TEXT] ←
  -  Enter type [TEXT] ←
  -  Enter factory [TEXT] ←
  -  Enter checkpoint [TEXT] ←
  -  Click "save"

"replace text"



# *Methods and Concepts:* Bad Keywords

---

- Static data and workflows
- Too complex
- Only used once
- Many dependencies
- Lacking in synchronization
- Highly specific



# *Methods and Concepts:* Managing keywords

---

- Logical categories for keywords
- Maximal flexibility for minimal complexity
- Consideration about when to create keywords
  - To avoid redundancy, but →
  - Not just for readability (rename preferred)
  - Not for all data options (e.g. true / false)
  - Not “just in case”





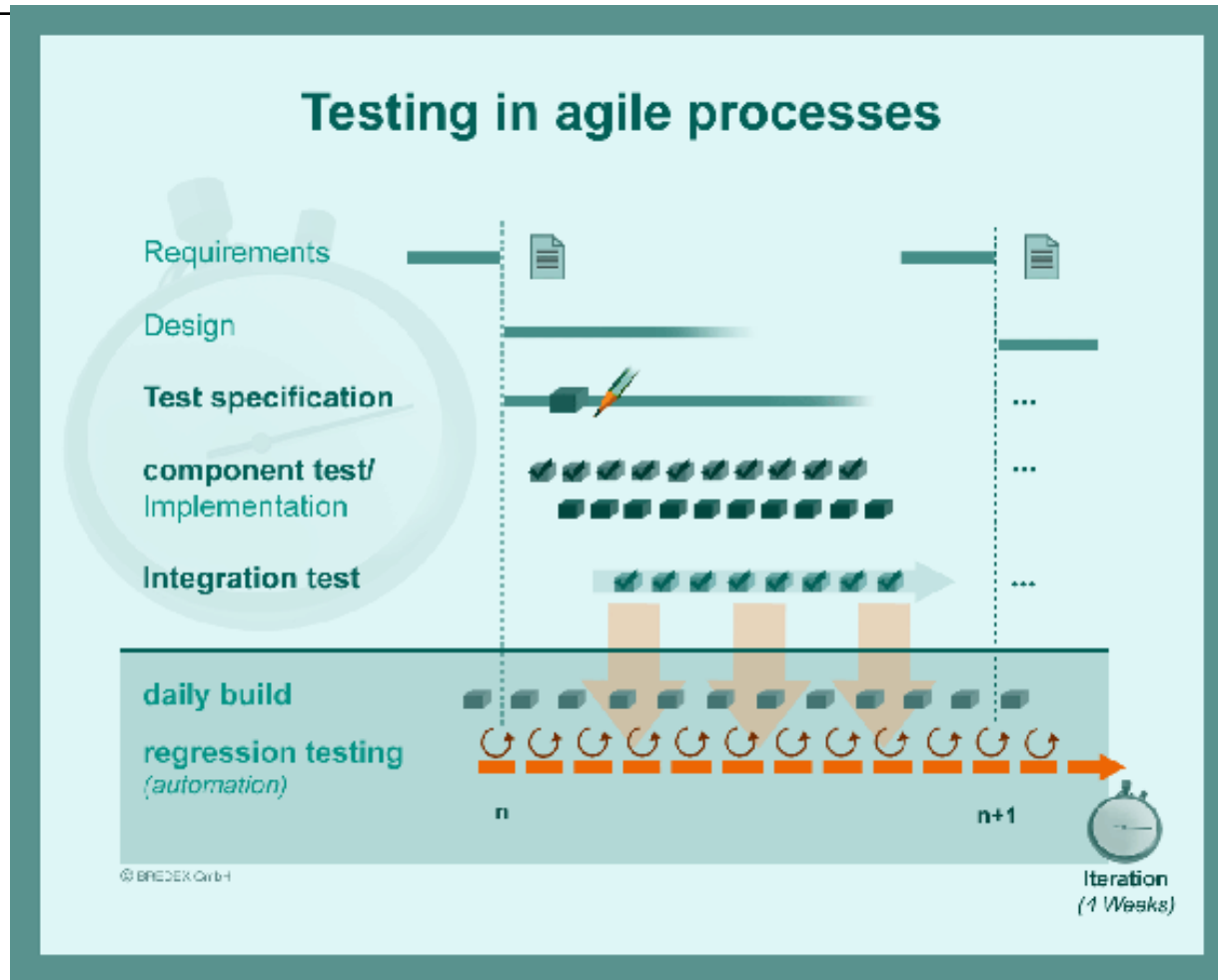
## *Results:*

# Effect on test and development process

---

- Agility
  - Tests created parallel to development
  - Regular releases for automated tests
- Communication
  - Misunderstandings cleared up
- → A movement to more agile processes within Bredex

# Results: More agility in process





## *Results:*

# Prompt feedback about quality

---

- Manual tests only performed at the end of milestone release
- Automated tests run on each new version (2x week)
  - Changes in GUI
  - New workflows
  - New use cases



## *Results:*

# Errors found early

---

- Before the handover to the manual test team, errors found:
  - 10 A-Errors
  - 20 B-Errors
  - 20 C-Errors
- Comparison with other projects:
  - ~ 1000 Errors



# *Results:*

## Minimal manual test effort

---

- Only 26 errors found in manual test phase
  - 4 A-Errors
  - 9 B-Errors
  - 3 C-Errors
  - 10 non-errors
  
- All errors fixed during test phase
  - Regression tests ensure that fixes don't cause new problems

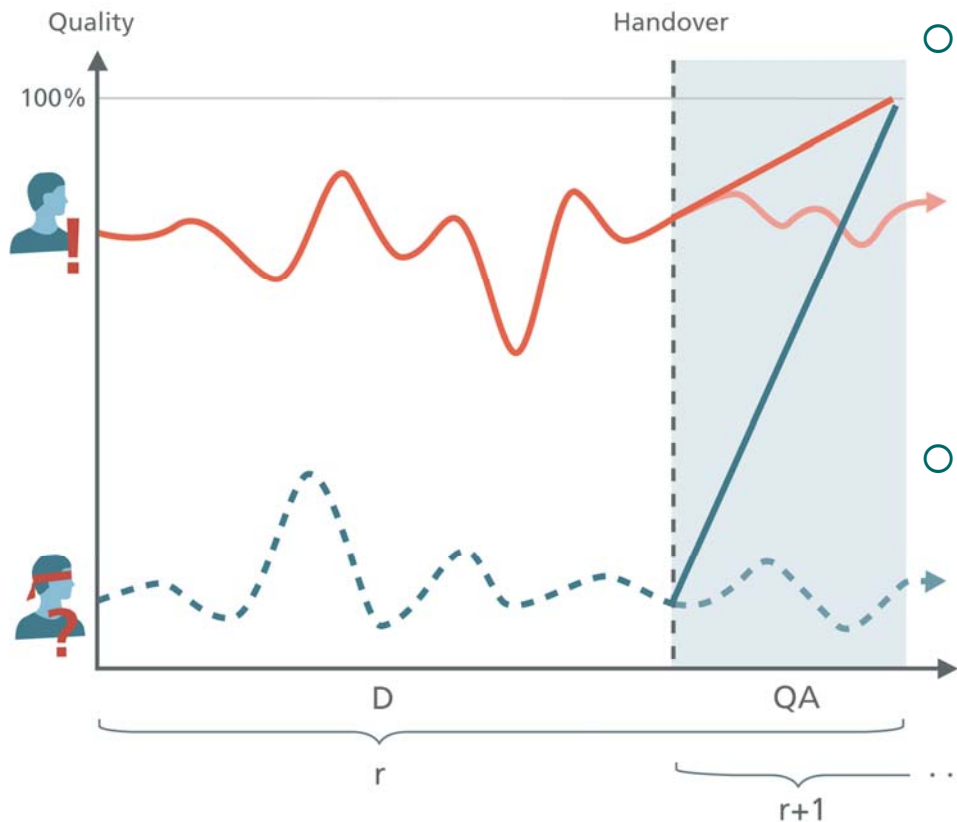
## *Results:*

### Low automation effort

---

- Less than 10% effort for test automation
  - 6,900 h for development
  - 600 h for automated test
  - Weekly maintenance effort 1 h
- 275 GUI components tested
- 13,000+ single test steps
- Final tests ran 9h

# Conclusion



## ○ Results for customer

- ROI on test automation
- High quality for low price

## ○ Results for company

- Project complete
- Reputation
- Low or no service