

Future proof Test Automation Architecture based

Jos van Rooyen



- Employed at Identify as partner / principal consultant
- 30 years in software testing & quality management
- Co-author TestGrip, TestFrame, Project de Baas, Quality Supervision, Textbook; “Aan de slag met software testen”, Cleantxt, Test Automation Architecture (available soon)
- Test expert online magazine Computable
- Publication areas; Test process Improvement, BI-testing, Test automation, Test Education, Risk Based Testing, Quality Supervision
- Member NESMA working party; Metrics in Contracts
- Visiting lecturer Universities of Applied Science
- Member advisory board Hogeschool Utrecht
- Member of several working parties Dutch Testing Society
 - Member of the board
 - Test Education universities of applied science



Introduction

- Test automation is being applied for more than 30 years in administrative software systems
- Not always successful due to several reasons
- A lot of attention is paid to the technique but embedding test automation into the organisation is in many cases not arranged
- One of the reasons of failing test automation projects
- Future proof test automation by hand of test automation architecture could be an answer
- The implementation of test automation architecture is the scope of this presentation

Agenda

- Current situation of applying test automation
- Pitfalls of test automation
- The solution:
 - Approach
 - Temple
 - Principles
- The benefits
- Assessment model
- Future work



Some pitfalls of test automation

Tool driven approach

Lack of technical skills

No vision in the organisation

Huge effort for running test automation

Local initiatives

A lack of learning cycle

Interpretation of the problem

How to organize test automation in such a way that it is applicable not only today but also tomorrow with as less maintenance as possible

The solution; Future Proof Test Automation

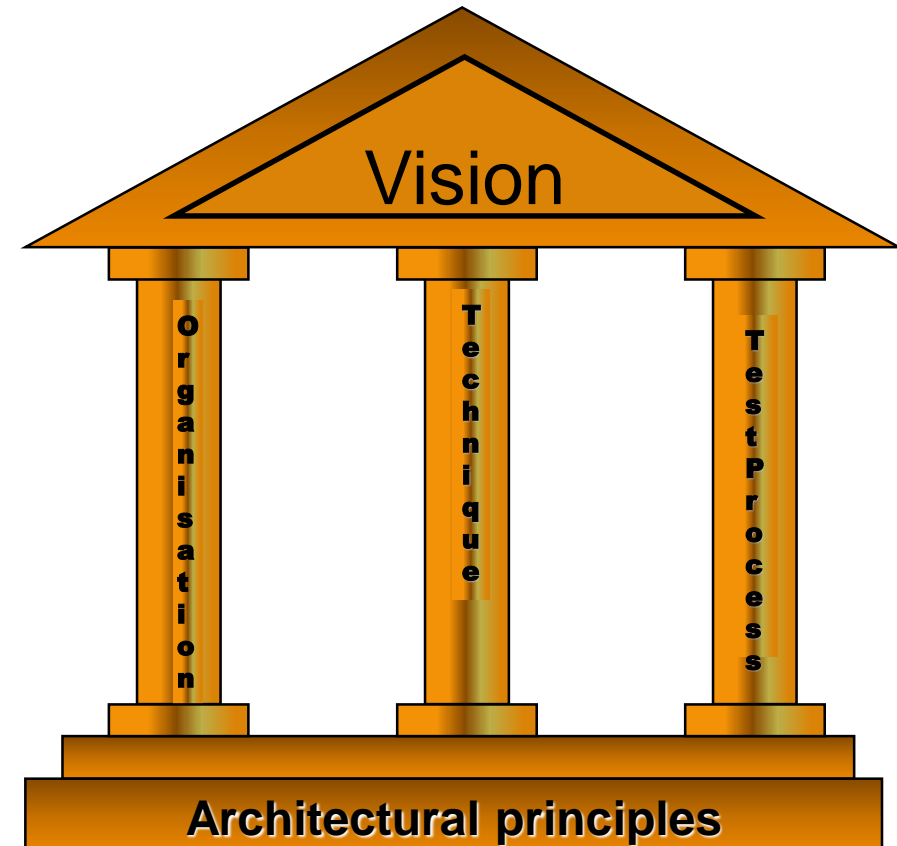
Future proof test automation takes care of the structural embedding of test automation into the organisation in such a way that processes are adapted (SDLC/Vendor management), test engineers with the required skills are available, necessary tools are available and test automation is secured on the right level in the organisation.

This way, developed test scripts are reusable, repeatable and transferable between projects and people inside domains, with as less maintenance as possible.

All developed material is available inside a central knowledge base.

The approach

- The approach is based on a clear vision; what does the organisation want to achieve
- Based on 3 main pillars:
 - Organisation
 - Technique
 - Test process
- Fundament based on architectural principles



Vision

Which business goals must be supported by hand of test automation, such as:

- Increase quality-to-market
- Increase time-to-market
- Automation of the devops pipeline
- Decrease the dependency of scarcity of business knowledge
- Less dependent on certain resources

What are the test driven goals that must be fulfilled by hand of the implementation of test automation, such as:

- A framework for the total organisation
- 90% of manual test execution must be automated

The 8 Architecture principles

Organisation principles

- Test automations fits into the goals and maturity level of the organisation
- Test automation is based on a clear vision, policy and architecture
- Test automation takes the human dimension into account
- Test automation requires a well-considered consideration off effort and risk

Information principles:

- Test automation is model-based
- Data for test automation must be managed
- Test automation explicitly takes into account information security
- Test automation tools are necessary but only supportive

The pillars

- Organisation:
 - How to organise test automation. On which level into the organisation
 - Sponsorship by management
 - Defining roles, tasks and responsibilities regarding test automation
 - New and adapted roles
- Technique:
 - Type of tools available
 - Skills of test automation engineers
 - Reusable, repeatable and maintainable testscripts
 - Test data management

The pillars

Test process:

- Adaptation of the software development life cycle
- New processes like automating test scripts, structuring of test data and availability test environment (roll-back, tool implementation)

Implementation of Test Automation Architecture

- How to start?
- Starting from scratch or re-use available testware
- What is the vision of the organisation
- What are the desired objectives that must be reached?

First STEP: Risk Test Automation Assessment (RTAA)

The RTAA model

- Getting insight in current situation of applying test automation into the organisation. What is already available?
- Determine the desired situation
- Organise the transformation
- Assessment around:
 - People
 - Organisation
 - Data
 - Technology
 - Process



RTAA, *Example*



New question

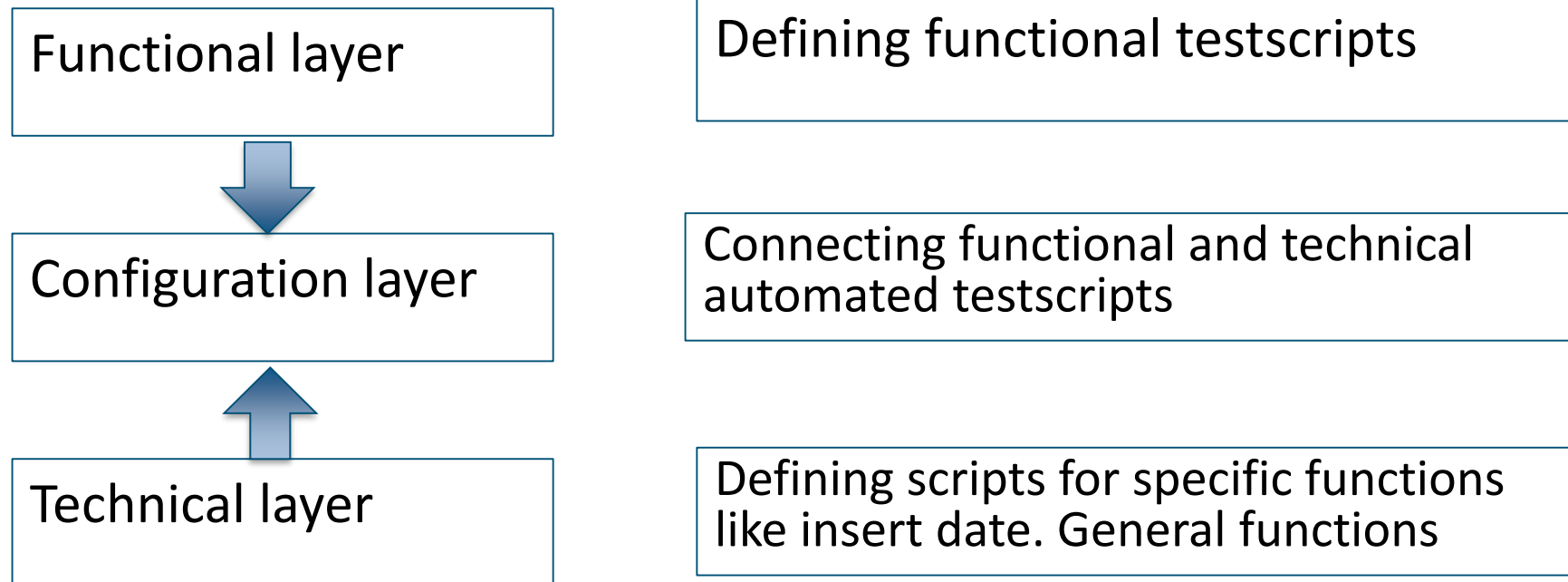
Are there employees with experience in test automation?
Do developers see added value in test automation?
Are employees consciously working with test automation?
Is a sufficient number of employees available with knowledge of, and experience in, test automation?
Do employees have sufficient knowledge and experience regarding test automation?
Are employees motivated to start with test automation?



The model

Maturity assessment test automation		<i>[company name]</i>			
Carried out on		<i>[date]</i>			
	Level 1	Level 2	Level 3	Niveau 4	
People					
Organisation					
Process					
Data					
Technology					

3-layer set up



Experiences

- Last 5 years we have been using and adapting the approach
- Companies in the industry, government and semi-government are using the approach
- Some experiences are:
 - A lot of attention is paid to the technical layer and learning the tools
 - Using the presented model increases the success of applying test automation
 - Embedding into the organisation is a problem
 - By hand of RTAA people get aware of the problem why test automation is not always successful
 - Time must be allocated for setting up and adapting the test scripts
 - People must be selected and trained for the test engineering role

The benefits

- Test automation is secured into the organisation
- Test automation is repeatable and reusable between projects and domains
- Always available to run
- Support of devops development
- Reliable quality of the system under test
- Less dependency of resources
- Shortening of the required test execution time

Future work

- Implementing the RTAA model and gain experiences
- Enhance the method based on the experiences
- Collecting metrics implementing TAA
- Start up an experience of repeatable test automation building blocks