

9th
UCAAT *User Conference on
Advanced Automated Testing*

How to make your
automation last
forever

Martin Gijsen

sogeti
Part of Capgemini 

15 September 2022



Forever?

The SUT

Lasts for years

The automation

As long as the SUT

Safe

Low maintenance:

- Low maintenance sensitivity
- High maintainability
- Flexible design & architecture

Transferable

The test case: What, not how

Open the browser
Open "https://my.webshop.com"
Type "mgijsen" in "username"
Type "S3cr3t!" in "password"
Click "Log in"

...

...

...

Log in to the webshop
Place an order
Check for an order confirmation email

Given I am logged in
When I place an order
Then I receive an order confirmation email

What, not how, in test data

Enter personal info "John Doe" at "123 Main street" in "Oxford" email "... " telephone "... "

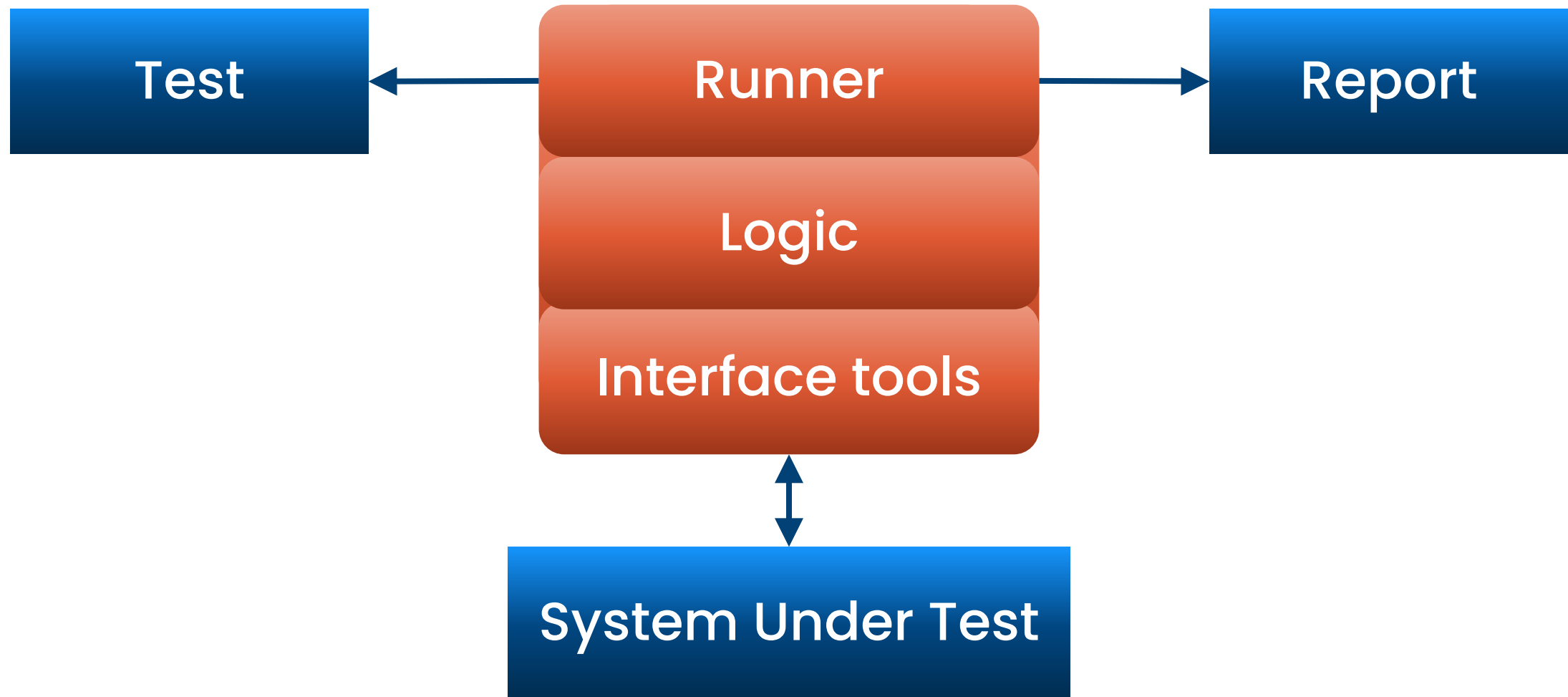
Enter personal info for "an international customer"

Coding standard:

- Consistency
- Comments
- Maximum size of file, class, method, ...
- Naming

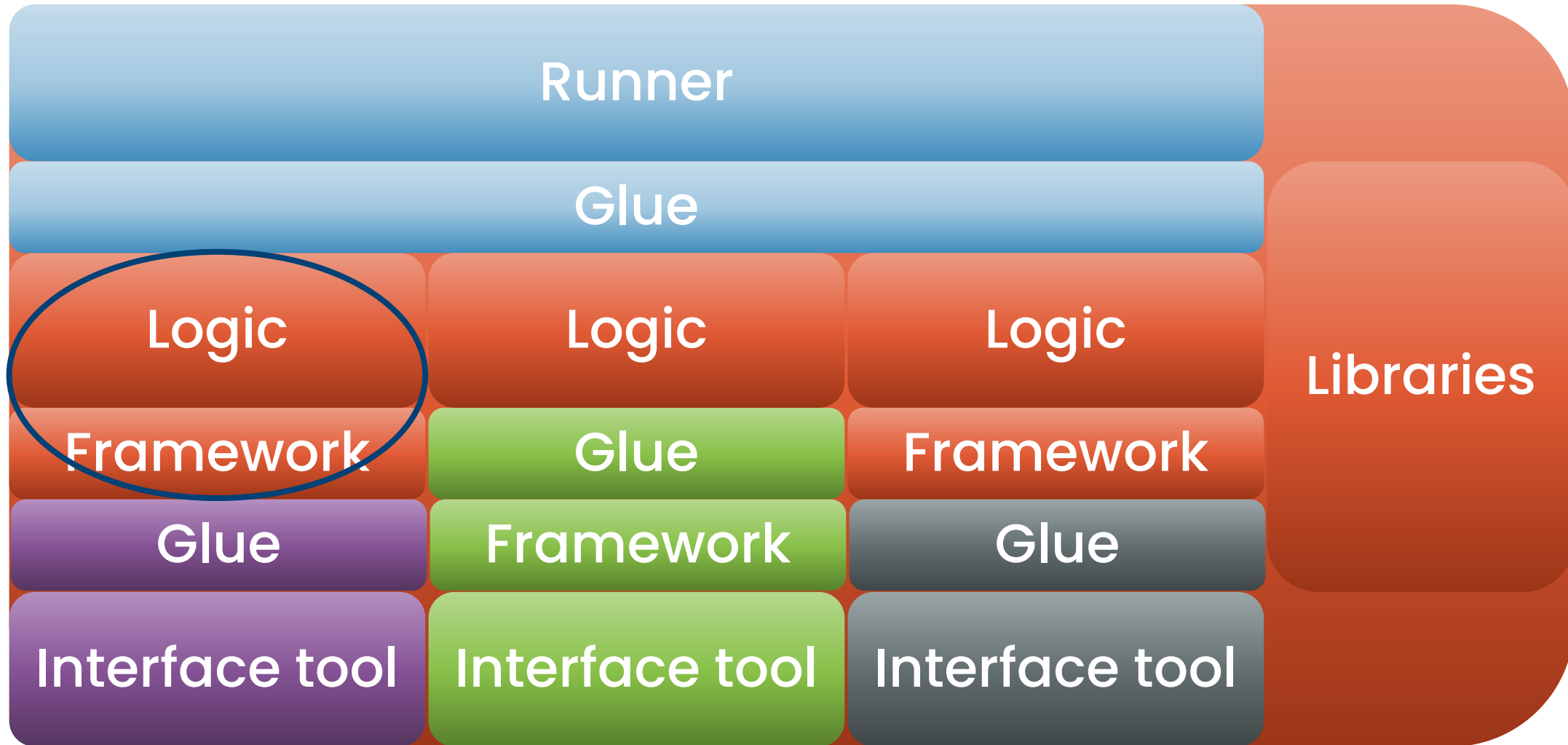
Code reviews

Avoid duplication (DRY / DIE)



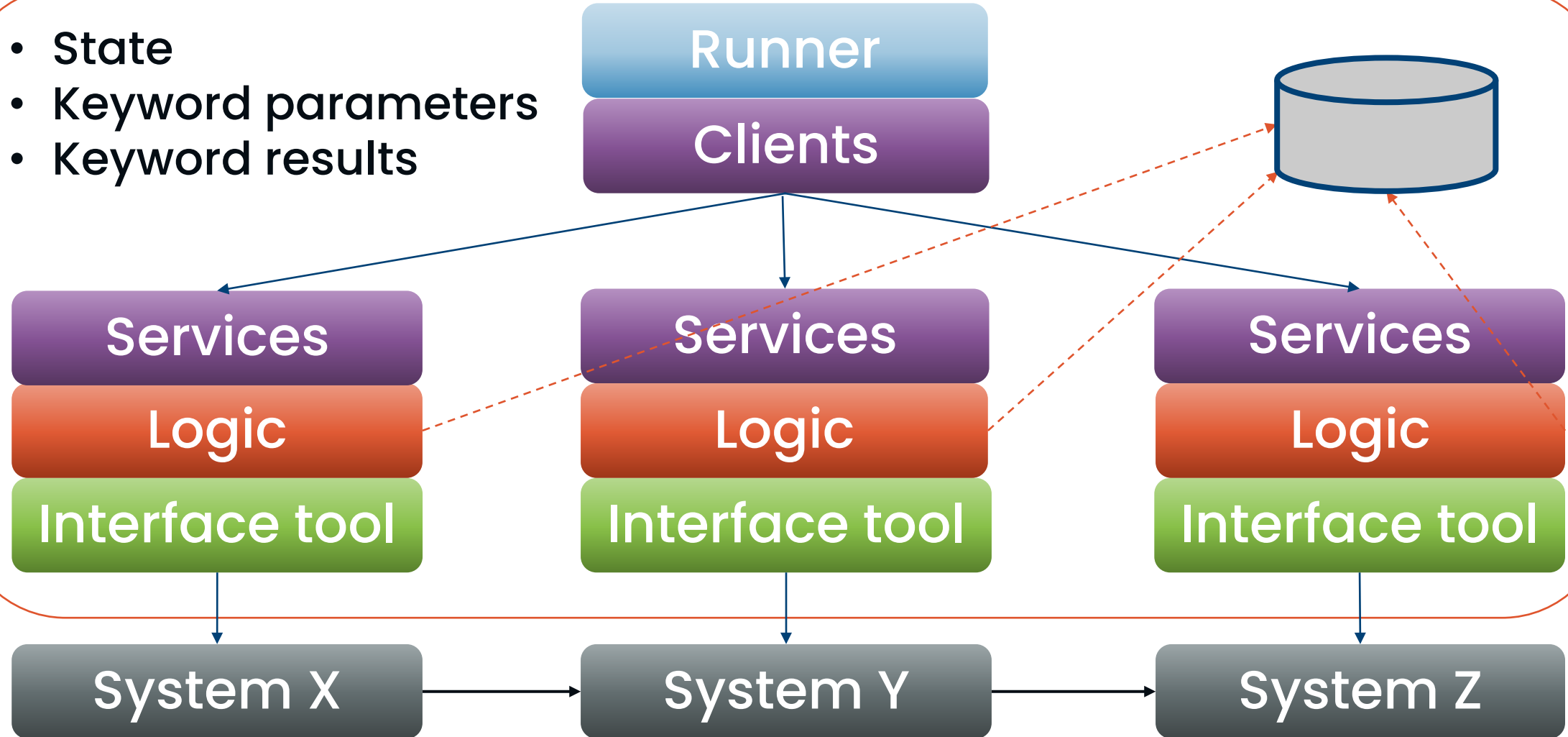
Design measures for lasting automation ^{9th}UCAAT

Design patterns



Architecture: Enterprise automation

- State
- Keyword parameters
- Keyword results



How long can it last?

The SUT



The test cases



The logic



The tools



The framework

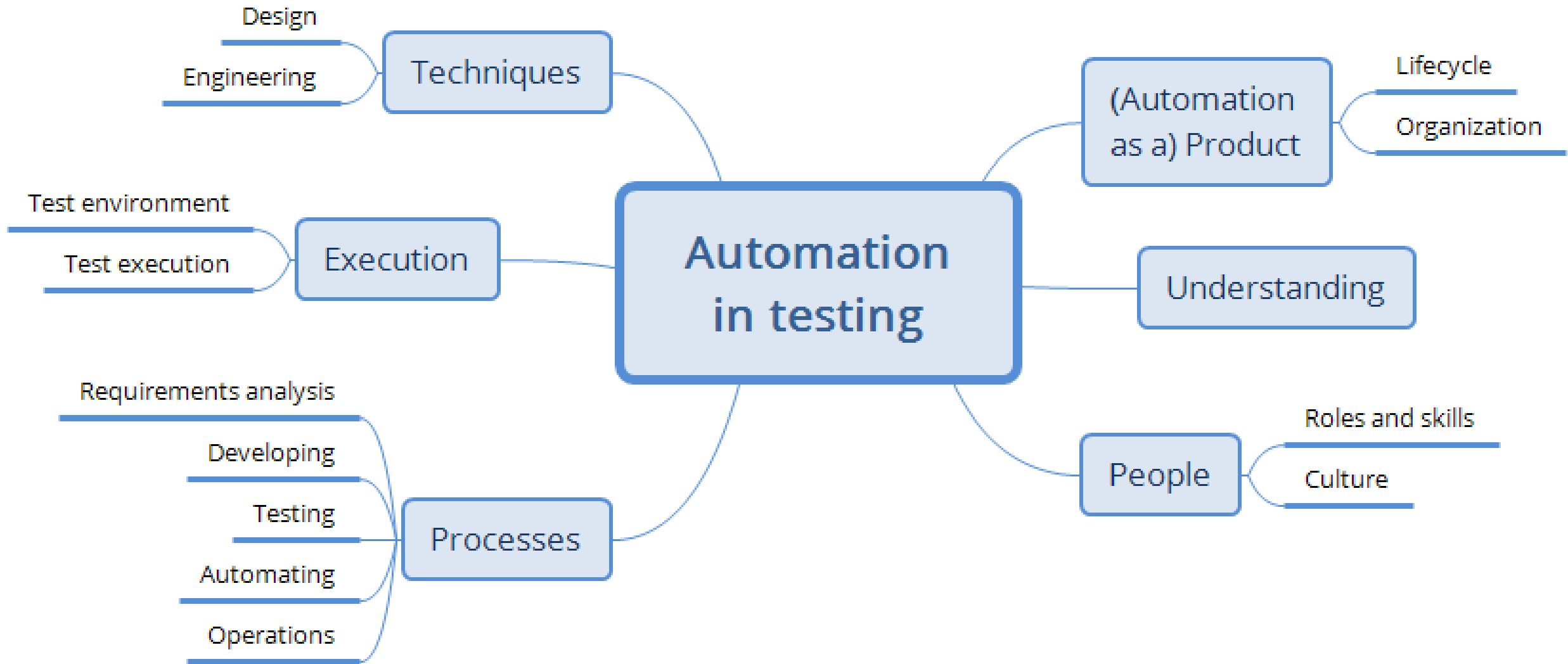


So ...

Apply the engineering techniques
that are used for production software
to your automation to enjoy it
for ~~much~~ longer

Is that it?

No ...



Any further questions?

Let's chat!

At UCAAT

On LinkedIn : Martin Gijsen

On Twitter : martin_gijsen

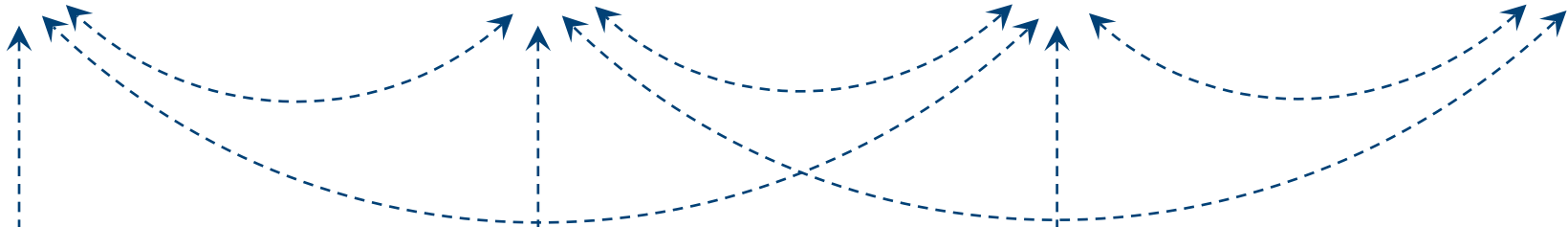


Organization

People



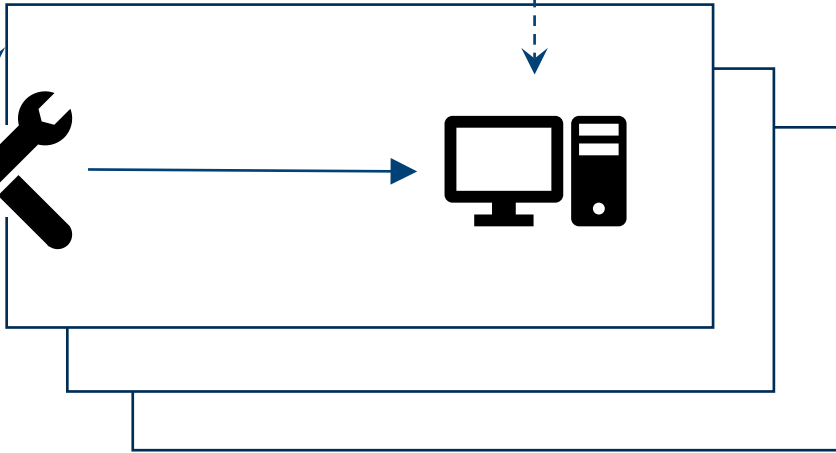
Processes



Execution



Techniques



The PUPPET model

Organizational context

Automation in testing

(Automation as a) Product

Lifecycle

- What is the scope (and how may it change)?
- What is the main goal and how is progress measured?
- How will you proceed through the lifecycle?

Organization

- How are decisions made?
- Where does it take place?
- How is it supported?

Understanding

- Automation must support testing
- Automation = testing + automating (+ infrastructure)
- Automating is software development
- The tool is neither the solution nor the problem
- What are you doing it for?
- Lasting value requires good care
- Irrelevant detail in checks is dangerous

People

Roles and skills

- What roles are there and who perform them?
- How well do all stakeholders understand automation?
- What skills are needed and how are they developed?
- How are all stakeholder needs met?

Culture

Does culture support the effort?

Techniques

Design

- What tools will be used?
- How is the solution designed?

Engineering

What engineering practices are applied?

Execution

Test environment

- How will the SUT be deployed and interacted with?
- What test environments are needed?
- How can you test in isolation?
- How to deal with test data, setup and cleanup?

Test execution

- How are checks prioritized, selected and executed?
- How are analysis and debugging supported?

Processes

- What are the objectives?
- When are checks written and who uses them?
- What has the most value to automate (now)?
- What process does automating follow?
- What is involved in the execution of checks?
- When is a check deleted?

Lifecycle of checks