Agile & Risk Based Testing

SOGETI

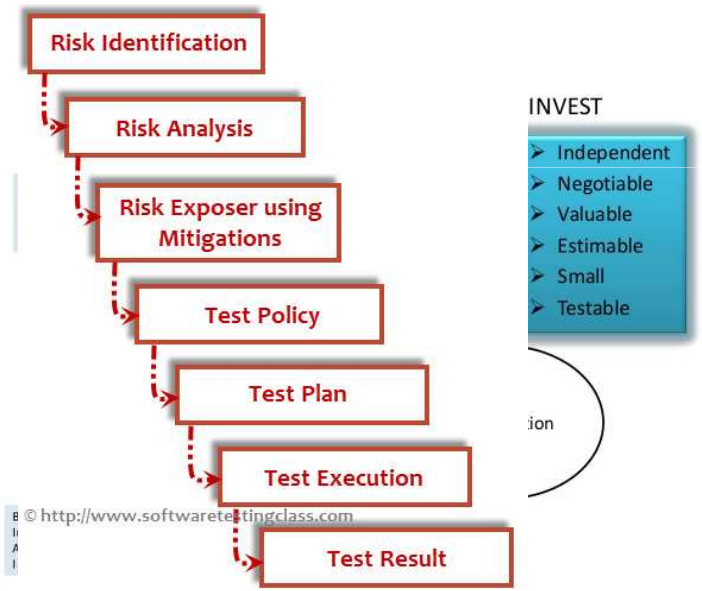
Risk based test automation in an Agile environment.

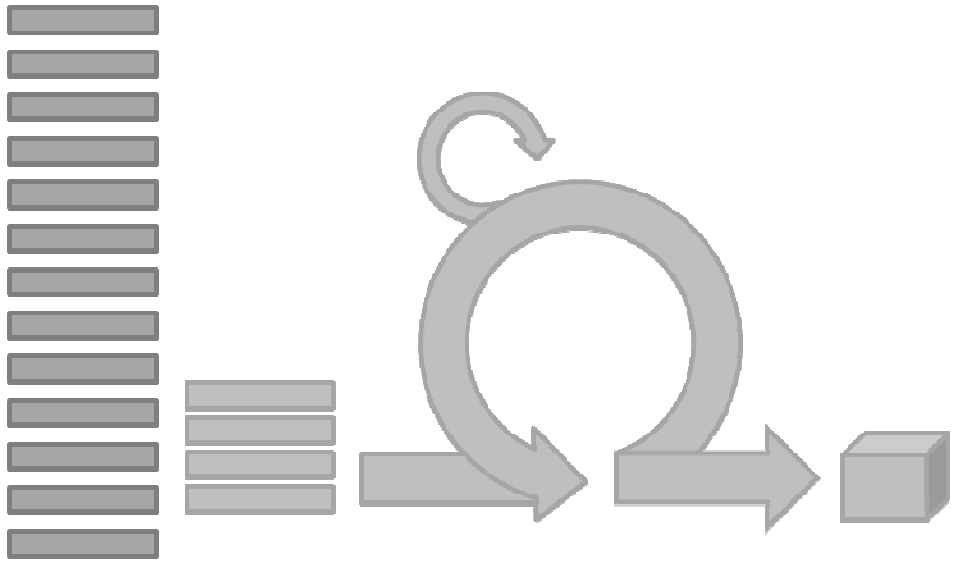
..

Ben Visser & Peter Nwankwo
Amsterdam, June, 2017

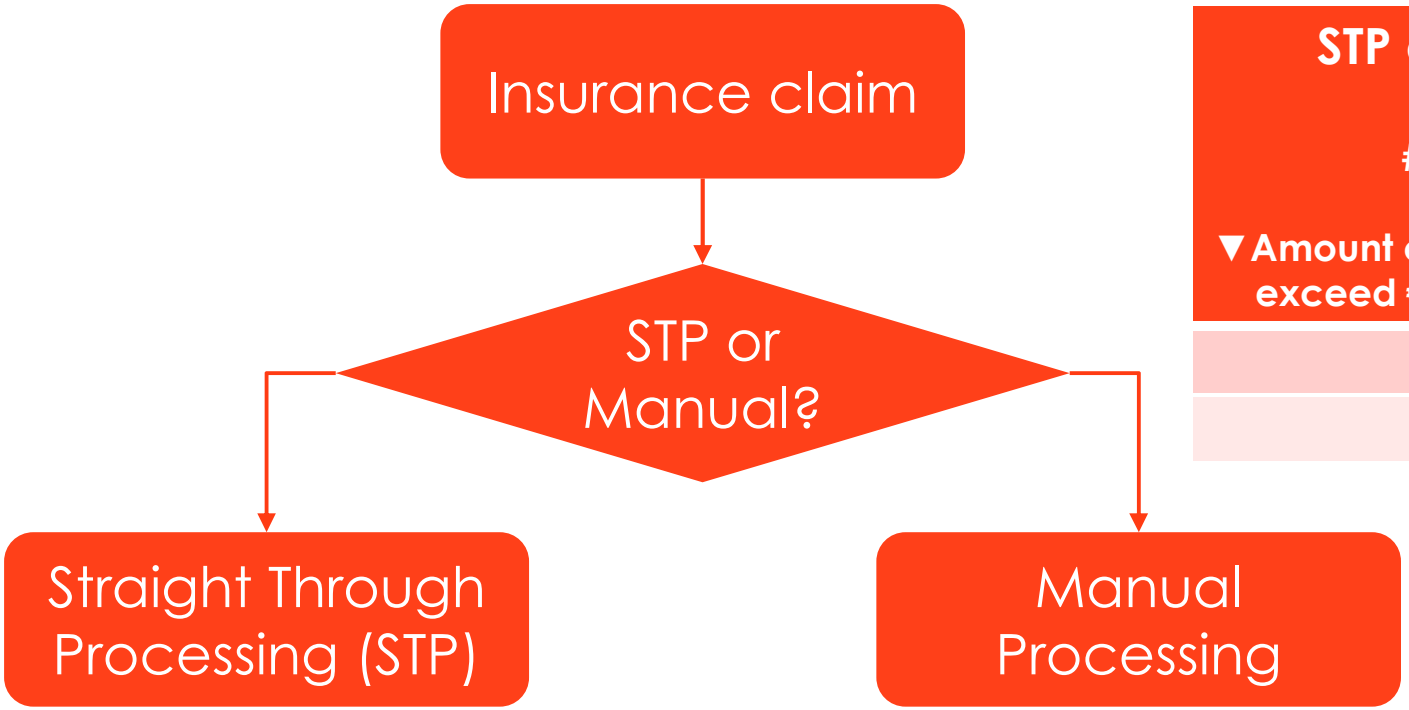
SOGETI

Risk Based Testing ⇔ Agile development



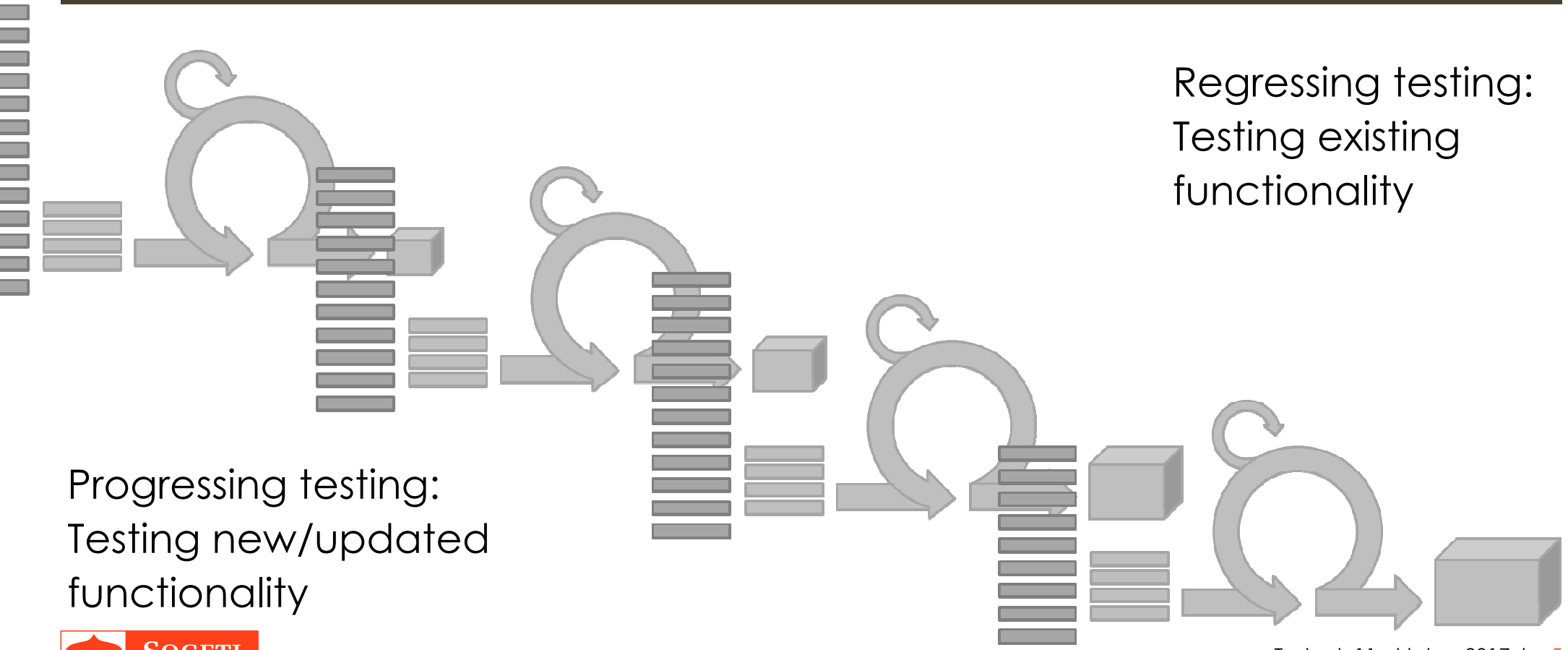


No risk, no test



STP or Manual? # claims exceed 3 ► / ▼Amount of claim exceed €10.000	Yes	No
Yes	Manual	Manual
No	Manual	STP

Risk Based Testing ⇔ Agile development



Regressing testing:
Testing existing functionality

Progressing testing:
Testing new/updated functionality

SOGETI

Developing a Premium Calculator

Sprint 1:
- Traffic
- Consumers and Home
- Health and Family

Progression testing:
- Explorary testing
- Automation: data coverage

Sprint 2:
- Work and Income

- Progression testing:

- Regression testing

Developing a Premium Calculator

Sprint 3:
- Pension and Social Security

Progression testing:
- Explorary testing
- Automated?

Regression testing
- …

Sprint 4:
- Taxes and Assets

So, where is the risk based aspect …?
- New functionality in sprint: all combinations
- Regression test: all pairs

- Cross browser testing
  - All values in daily build
  - All combinations every weekend

Cost Requirements For Insurances

Insurance	Family Single		Family Married	
	Without kids	With Kids	Without kids	With Kids
Traffic	€ 4,42	€ 4,60	€ 4,60	€ 4,60
Consumers & home	€ 7,21	€ 7,51	€ 7,51	€ 7,51
Health & family	€ 1,68	€ 1,68	€ 4,71	€ 4,71
Work & income	€ 7,79	€ 8,12	€ 8,12	€ 8,12
Pension and Social Security	€ 0,44	€ 0,46	€ 0,46	€ 0,46

Test Data Generation

Some Challenges with Test Data

- Changes in requirements
- *Updates needed in automated test data*
- *Less test coverage due to insufficient test data*
- *Need for test data in formats like JSON, CSV*

8 Conditions to test = 2^8 =256 possible test case.

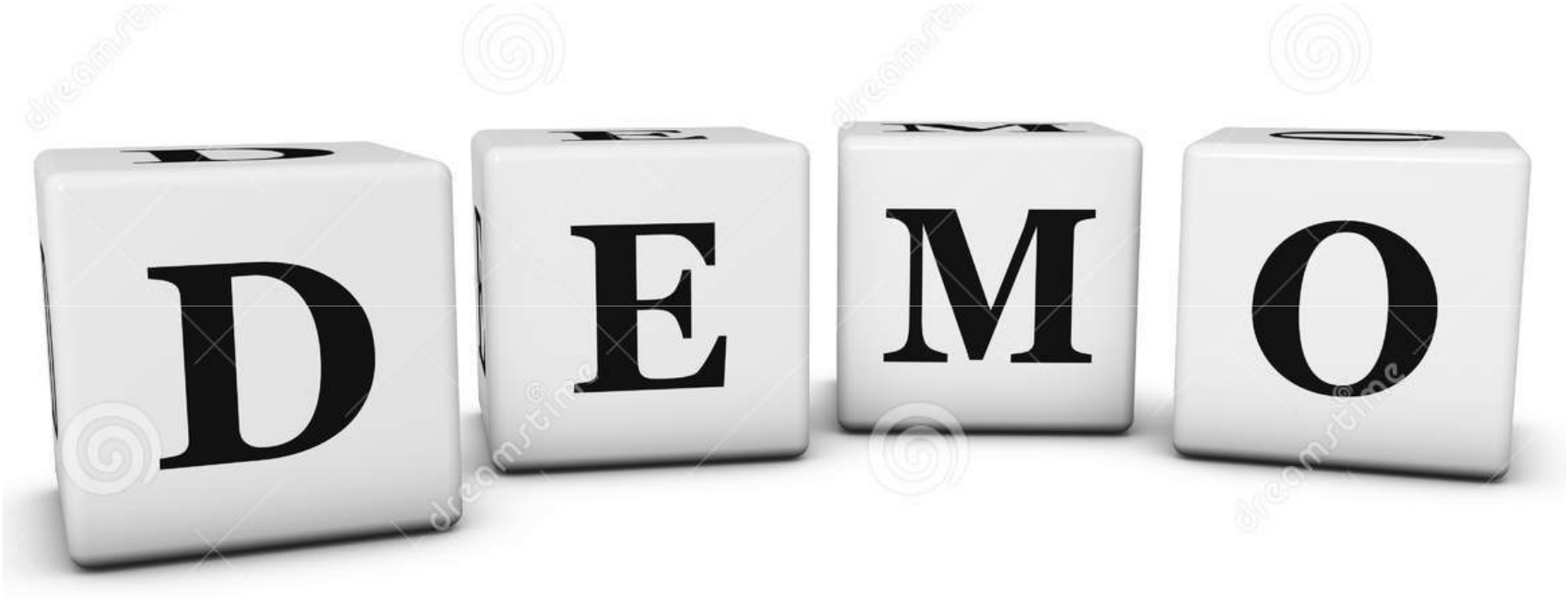
Solution for every challenge

Challenge	Metigation with tooling and methodologies
Incorrect test data leads to incorrect test results	Test data provisioned by tooling based on test conditions
Less test coverage due to insufficient test data	Tooling enables generating of required combinations of test data (All pairs, All values & All combinations)
Changes in requirements	Tooling enables regeneration of test cases easily
Updates needed in automated test data	Tooling enable regeneration of input test data
Need for test data in formats like JSON, CSV	Tooling enables regeneration of test data in required format i.e JSON, CSV etc.

Tools used for TDM

- Data Combinations for generating the different input combinations based on the conditions.
  - Input combinations based on **All Pairs**
  - Input combinations based on **All values**
  - Input combinations based on **All combinations**

- Excel for determining the expected outcome for the different input combinations


DEMO

Summary

- Not everything needs to be tested in the CICD pipeline.
- Don't maintain individual test cases, maintain the test model and generate the test cases.
- Don't maintain expected results for your regression test set, get them from production.

Thank you!