

Bordeaux, 22-24 October 2019



ADA Project Evolution: test execution in Python toward a TLD driven testing platform

Presented by Sergio Borghese

Who We Are

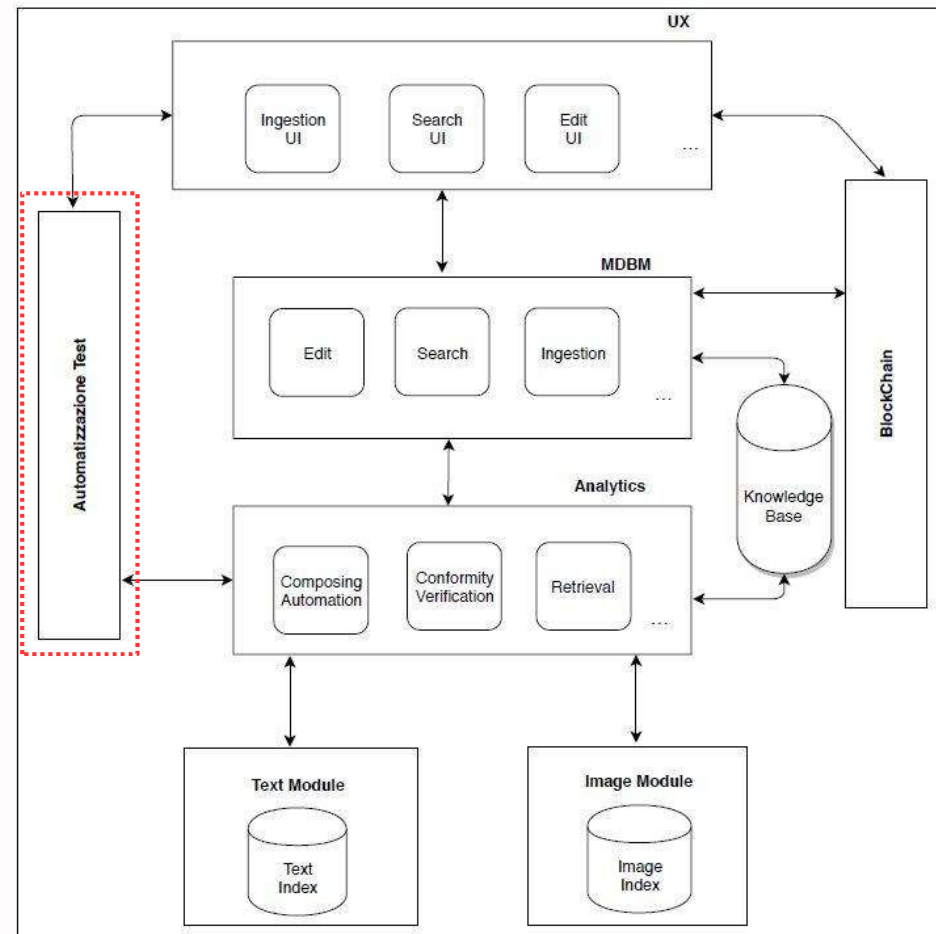
- Born in 2016 from the **research group in Telecommunication** at the University of Pisa (Italy)
- Spin-off of Pisa's Univeristy
- SME with strong skills on **VoIP/MoIP** and **network performance testing**
- People: 26
- With degree: 17/26

The ADA Project

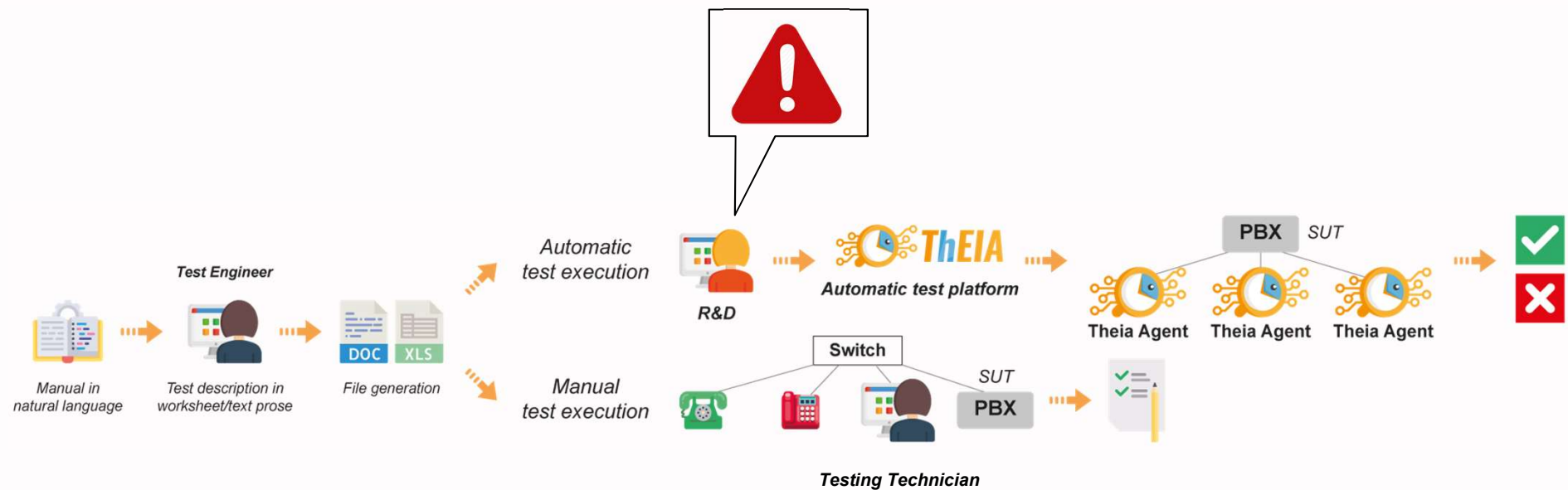
- ADA is a research project founded by Tuscany Region (Italy)
- It aims at augmenting **human driven processes** by the use of **advanced data and document analysis**
- NR's role is to contribute to the **enhancement of the testing processes**
- Make more efficient the test automation activity carried out by the TAE (**T**esting **A**utomation **E**ngineer)

ADA Architecture

- *UX*
- *MDBM*
- *Analytics*
- *Blockchain*
- *Analytics*
- *Text Module*
- *Image Module*
- *Test Automation*



Original Testing Process



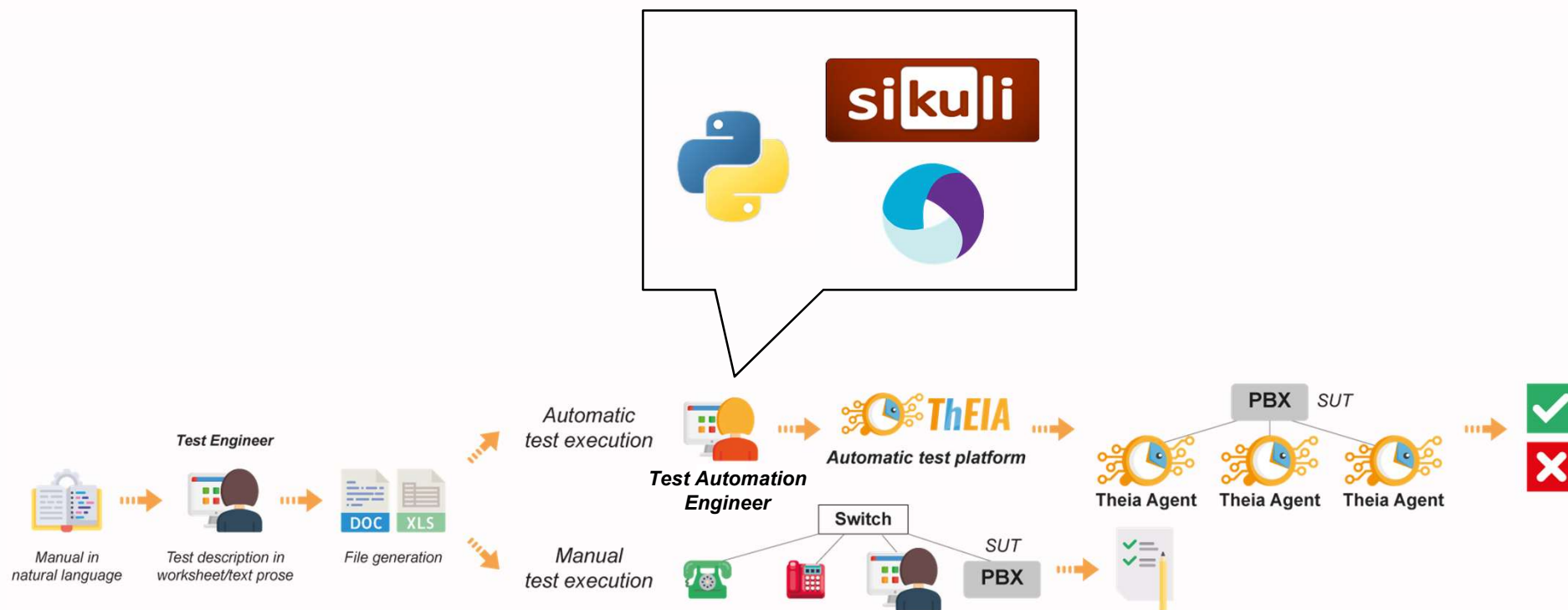
R&D Scaleup

- Today's most important activity in startups and scaleups is “**product discovery**” (PD)
- PD heavily inspired by “lean thinking” movement and “Lean Startup” approach
- ZLRD: Zero Latency R&D
- Testing Automation is key

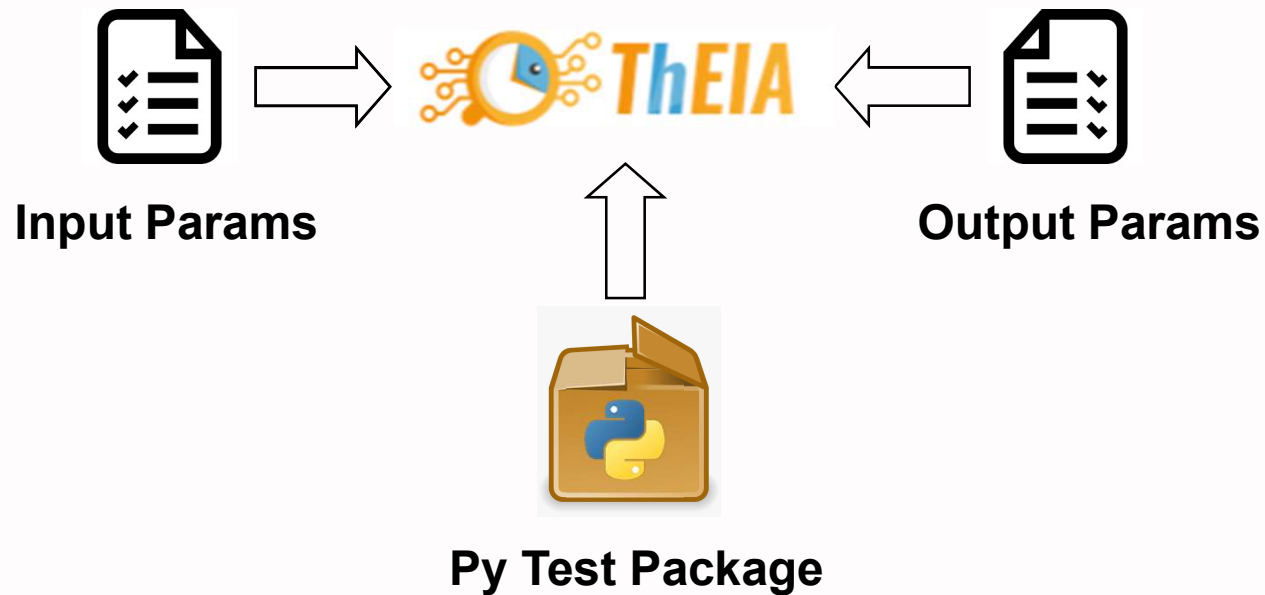
Removing the R&D bottleneck

- *R&D must not be in-line on the testing process*
- Allow TAE to develop new tests and test scenario independently from R&D
 - » TAE develop tests using python and package it
 - » TAE upload the test package to the test platform
 - » The test platform deploy the test package to agents
 - » Agents run the newly deployed test

Current Testing Process



Extending Platform Tests



Why Python...

- One of the most used programming language
- Huge ecosystem of third party libraries
- Binding for several automation tools:
 - **Sikuli**: test of non API GUI systems
 - **Appium**: mobile testing
 - **Selenium**: webgui testing
- **Tensor Flow**: machine learning

Target Testing Process



TDL & Python

- **Python** library to implement **test primitives**
- **TDL** to describe **test scenarios & interaction patterns**:
 - Full mesh
 - External Target
 - Hub & Spoke (i.e conference call)
- Centralize timing and sync management
- **PyTDL**

Mapping TDL to Python (1)

```
run {  
    // presenter calls the confcall system  
    presenterUserAgent.g_dialPad sends request_conn to confCallSystem.g_dialPad with {  
        STEP : "1" ;  
        PROCEDURE : "Set up Call" ;  
    } ;  
    // send user id  
    presenterUserAgent.g_dialPad sends UserId to confCallSystem.g_dialPad with {  
        STEP : "2" ;  
        PROCEDURE : "Send Digits" ;  
    } ;  
    // conference pin  
    presenterUserAgent.g_dialPad sends UserPasswd to confCallSystem.g_dialPad with {  
        STEP : "3" ;  
        PROCEDURE : "Send Digits" ;  
    } ;  
}
```


Mapping TDL to Python (2)

- **Open Points:**
 - Pass parameters
 - Map TDL Testers to Agents

Q&A

Authors:

- Sergio Borghese
 - s.borghese@netresults.it
- Enrico La Vela
 - lavela@netresults.it
- Francesco Oppedisano
 - oppedisano@netresults.it

Bordeaux, 22-24 October 2019



DIVIDER TITLE

DIVIDER SUBTITLE