



A Collaborative Approach to Automated UI Testing in OFBiz

**Olivier Heintz,
OFBiz Business & Functional consultant
in IT department O'Toit Company**

Business Feedback and How to increase « Functional » contributions

- Apache OFBiz® implementation Feedback
 - ✓ *Agile development team*
 - With only 20% with an ofbiz large experience
 - ✓ *Selenium Webdriver test used*
- My working experience in ERP implementation companies
 - ✓ *Functional and Development separate team*
 - ✓ *Lack of collaboration*
- How to contribute for an non technical actors (consultant or key users)
 - ✓ *Business or Functional*
 - ✓ *Help for other consultant or key users*
- Software quality and ERP modularity
 - ✓ *Key points for me for Free open-source ERP driven by a community*
 - ✓ *Collaboration to and for contribute*

All these points illustrated with the GUI test context

Business and ERP context

- O'Toit
- ERP, Actors type, Functional Consultant
- GUI Test, requirement and cost

Technical context and definition

- OFBiz Selenium Webdriver Test reminder & summary
- Junit and Functional test
- Tests & Documentations

Collaborative Approach

- Unit test & Scenario test
- Javadoc as Help
- When and How

Tutorial & Implementation

- Help & Maintenance
- Modularity
- More contributor

A Market Place with open distribution location (Toits)

A digital Market Place associated to physical local distribution locations to be able to propose offers of the farming, production, retail and craft sectors available **in than same time** and in the **same place** (the toits).

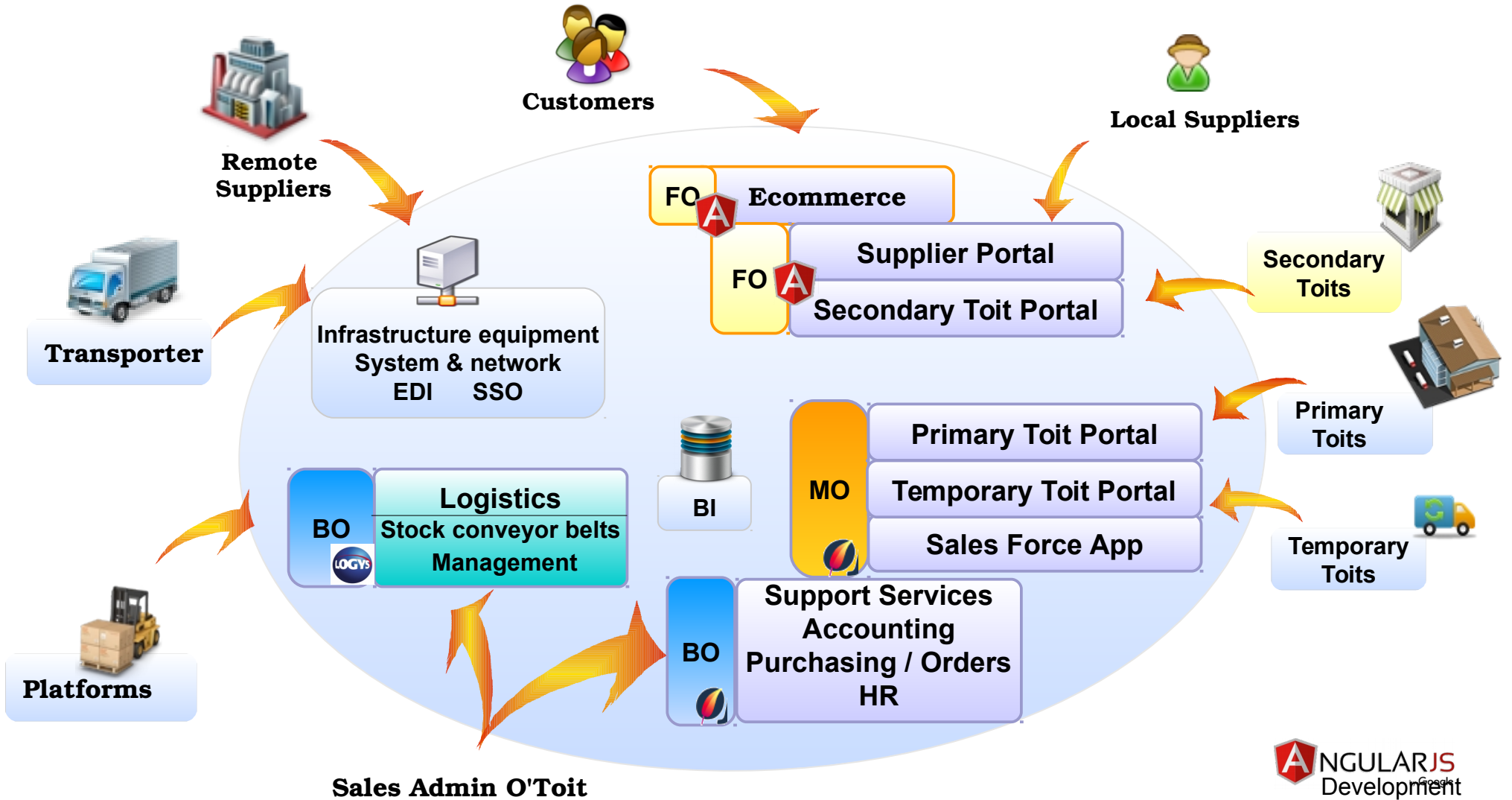


With the facility to order on **internet** or on **toit** or on the **local supplier** or **merchant** with scanner

And with convinience to collect the shopping **« à la carte »** form the near toit, or at home delivery or at local mechant or supplier (toit secondary)

A new distribution model which meets all the structural requirements of the farming, production, retail and craft sectors and the satisfaction of consumer needed.

Software Architecture



ERP implementation and development with Apache OFBiz®

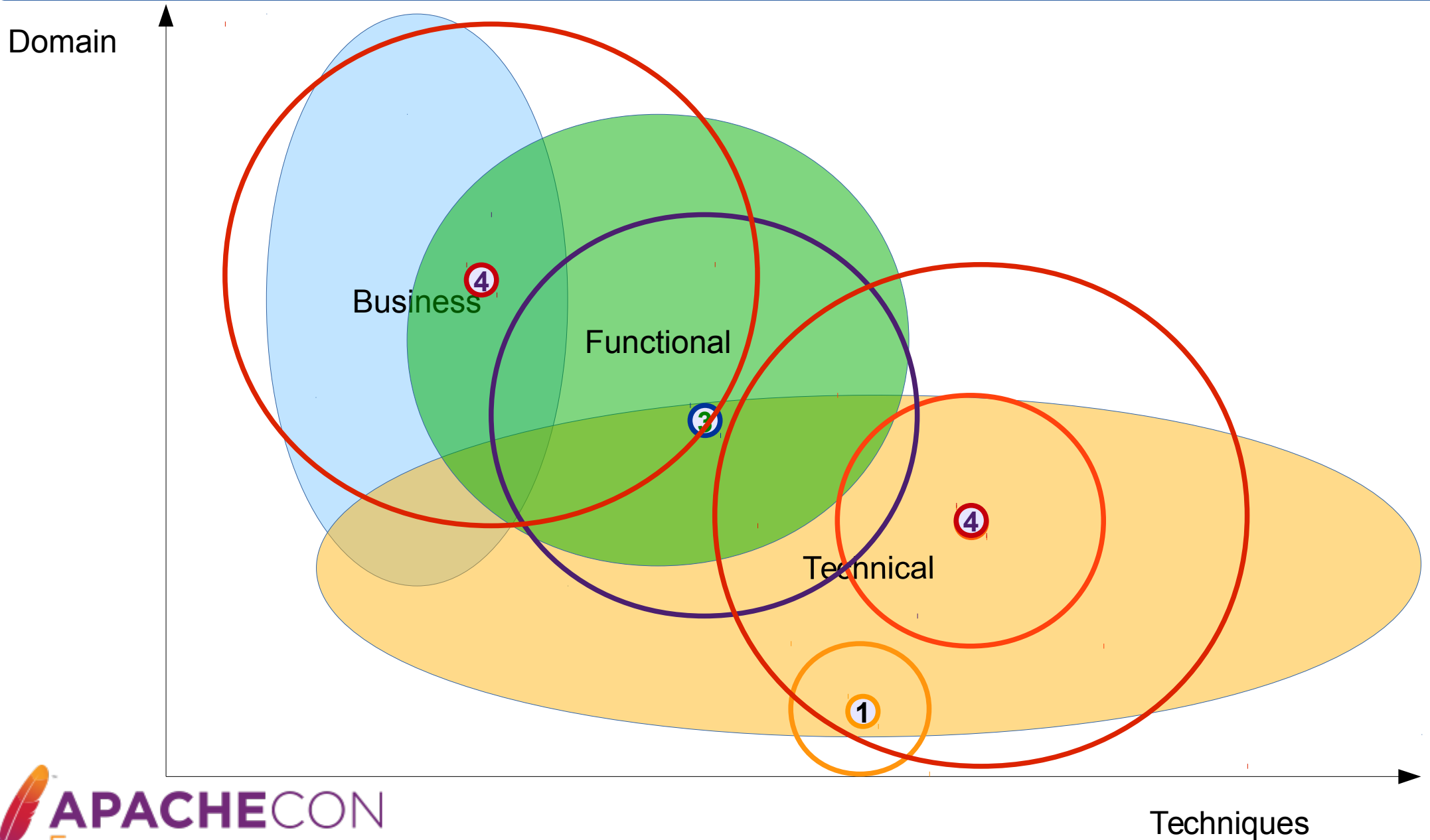
- Functionalities
 - ✓ *Back-Office :*
 - GUI very homogeneous, number of action type limited
 - Usable as is (but with param) for prod environment
 - A lot of functional domain
 - Need some data and context to understand
 - ✓ *Front-Office*
 - Most of time personalized
 - With or without ofbiz GUI framework
 - Should be user intuitive
- Implementation
 - ✓ *Not yet like other ERP*
- Development
 - ✓ *More efficient and fast than for other large ERP*

ERP implementation Team

- Product Owner – Business Analyst – Functional consultant – Technical consultant
 - ✓ *Knowledge center versus domain*
 - ✓ *Need to be a Team with multiple knowledge and view*

Knowledge / skill : domain or center ?

Collaboration needed for success



My definition

Technical Level

- Able to read code, not to write it
 - ✓ *Understanding OFBiz directory tree*
- Data structure knowledge but not all the details
- Writing « code » (if not too complex) if auto-completion is efficient
 - ✓ *XML, Groovy or Java*

Its roles & tools

- Trying a process by using screen
- Use parameters / « type » data / demo data
- Help / wiki / book
- Writing post-it, speaking to Product Owner or Technical consultant
- Promote the "Use existing more than Develop new"

In charge for all Functional test

OFBiz User and possible contributor

Customer / user requirement

- No understanding about regression
- Not ready to paid a large overload
- Not a lot of time to define scenario tests

Functional Framework

- Existing ready to use screen
- Existing demo data
- Build with layer
 - ✓ *Technical os interface, sgdb reading, html generation, ...*

Why a test failed

- Framework Technical change
- Function of Functional framework change
- Development change (std or internal)
- Data change (seed or internal)

Is the Gui test slow down the innovation ?

Business and ERP context

- O'Toit
- ERP, Actors type, Functional Consultant
- GUI Test, requirement and cost

Technical context and definition

- OFBiz Selenium Webdriver Test reminder & summary
- Junit and Functional test
- Tests & Documentations

Collaborative Approach

- Unit test & Scenario test
- Javadoc as Help
- When and How

Tutorial & Implementation

- Help & Maintenance
- Modularity
- More contributor

Reminder and Summary

<http://www.seleniumhq.org/>

Selenium is a suite of tools to automate web browsers across many platforms

- runs in many browsers and operating systems
- can be controlled by many programming languages and testing frameworks

OFBiz integration

- Code (java) and Data (xml) separation
- Utils method on a framework spirit
- Video & Log

Continuous integration

- Jenkins / docker
- Unit test and Scenario test
- Video is not record as default value for a lot of test-case

Scenario / test-suite / test-case

- Data-obj
- Included

Context initialisation

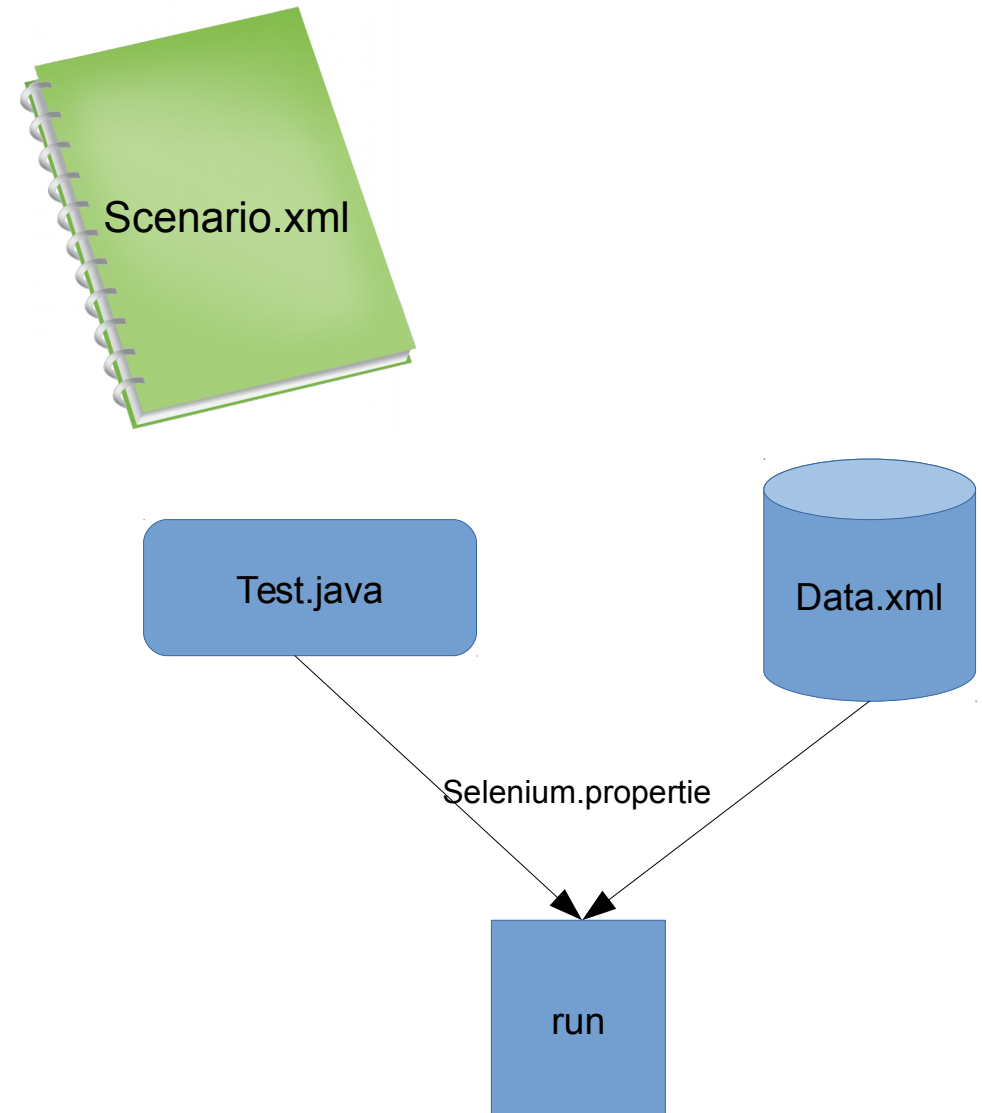
- Import xml
- Dedicated scenario

Data type

- String, **indexed**, cssSelector
- Date, Tmstamp
- Integer, Double

selenium.properties

- Scenario to use
- Index to use
- Browser, Screen dimension, log level,



Goal : framework for simplification developers life

SimpleTestSuite

- Data, log, video, ... integration
- Simplification seleniumQa
 - ✓ *Click*
 - ✓ *ClickForm*

Utils

- For data integration
- Complexe Dom Operations
 - ✓ *SearchLine*

LogSelenium

- *showInfoPanel, zoomId, logInfo, verbose, error, ..*

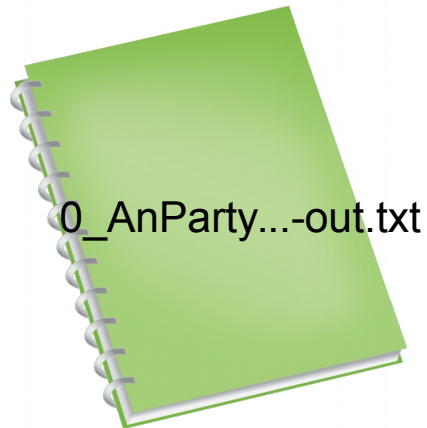
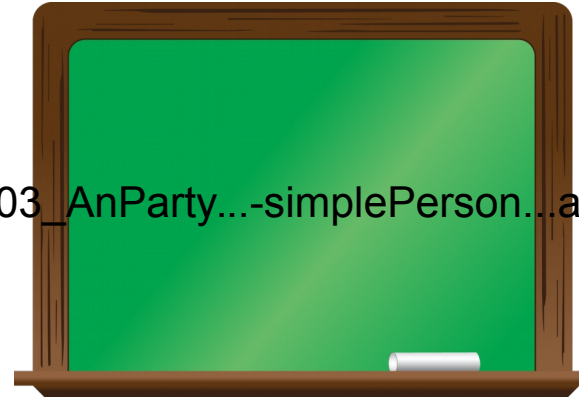
OfbizSWdHelper

- Classic ofbiz operation
 - ✓ *Login, logout, menu, ..*
- Complex classic ofbiz selenium task
 - ✓ *Lookup*
 - ✓ *WaitNextOrMessage*

For debugg or demonstrate

- ✓ *infoPanelEnabled=yes*
- ✓ *logPanelMessage=yes*
- ✓ *record.video=yes*
- ✓ *log.print.error=yes*
- ✓ *log.print.info=yes*
- ✓ *log.print.verbose=no*

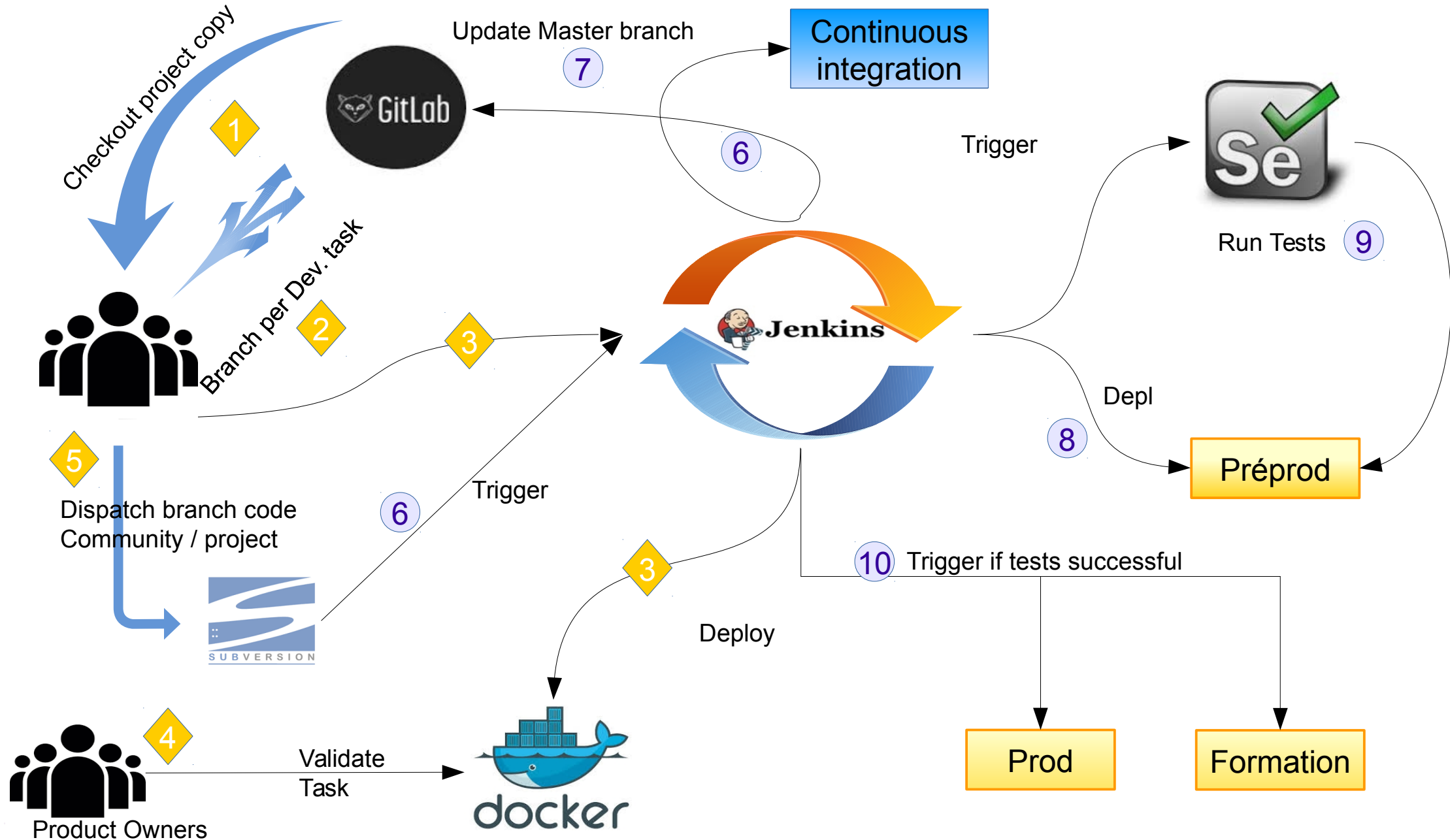
03_AnParty...-simplePerson...avi



Best Practice

- Be able to run gui test multiple time
 - ✓ *Indexed data*
 - ✓ *Webtools actions*

O'Toit Development Infrastructure



Business and ERP context

- O'Toit
- ERP, Actors type, Functional Consultant
- GUI Test, requirement and cost

Technical context and definition

- OFBiz Selenium Webdriver Test reminder & summary
- Junit and Functional test
- Tests & Documentations

Collaborative Approach

- Unit test & Scenario test
- Javadoc as Help
- When and How

Tutorial & Implementation

- Help & Maintenance
- Modularity
- More contributor

Acceptance test

Junit Gui test

- “mechanics” test without context constraint
- Done by developpers for the developpers

Functional test

- For functional (or business) consultant
 - ✓ *And product owners*
- Base on Use Case, for one context
- Multiple screens /applications

Technical architecture charge capacity test

- Not working on it, so no feedback, so nothing to say about 😊

Standard or Initial requirement : « Done by dev. for the dev. »

Development Best Practice

- Test is done in the same time that development

Apache OFBiz standard

- To deliver a quality application ⇐ same goal as service Junit test
- Menu / screen or part of screen

Portal Page and Portlet

- Modularity should help to have re-usable test
- Each portlet parameters should tested

Done by developpers

- Test all technical aspect, if possible without context constraint
- Put a lot of « assert » to easily localize error when test failed.
- Do action methods usable for functional test
 - ✓ *Work together with functional consultant to deliver usable and documented tools*

Customer requirement : « Function acceptance & no regression »

Acceptance process

- Base on Use case, define by Product Owner
 - ✓ *Linear : in one context*
 - ✓ *Included Data to use*
 - ✓ *Describe by user actions : manual or with the application*
- Build during development process
 - ✓ *With functional consultant*

No regression

- If new bug ⇒ is it sure it's a new bug ?
- When a test failed ⇒ is it a regression or a process evolution ?

Defined and used by Functional (& Business) consultant

- Product Owner translation to formal language
- “Type” data and screen parameters knowledge
- Bug and test failed first analyst level

Short test in Junit spirit (Developers domain)

- ✓ *Auto-documented by title or by one sentence*
- ✓ *Easy to read & understand what is tested*
- So, no documentation ?
 - ✓ *Only if "service" to test is simple*
 - ✓ *Only if used data is simple*

Test for who

- Developers
- Functional user

Developers pleasure priority

- ✓ *Code*
- ✓ *Test*
- ✓ *Documentation*

Development Best Practice

- Writing convention
- Modularity
- Documented
- With Junit test

Documentation reader

- Developers
- Functional user

Functional pleasure priority

- ✓ *Using*
- ✓ *Documentation*
- ✓ *Test*
- ✓ *Code*

Business and ERP context

- O'Toit
- ERP, Actors type, Functional Consultant
- GUI Test, requirement and cost

Technical context and definition

- OFBiz Selenium Webdriver Test reminder & summary
- Junit and Functional test
- Tests & Documentations

Collaborative Approach

- Unit test & Scenario test
- Javadoc as Help
- When and How

Tutorial & Implementation

- Help & Maintenance
- Modularity
- More contributor

Method type

Actions & Test

- In Junit test there are both
 - ✓ *Goto Menu-item1* -
check if Page is open
 - ✓ *CreatePerson* -
check if person is created
- Action == visual button or menu

Scenario test

- A serie of actions
 - ✓ *with user data*
- Some check (not a lot)
- Should be write by
Functional Consultant

Unit methods are usable by functional for Scenario

- Unit methods not only for one test and so should be correctly documented
 - ✓ *Creation of actions library*
- Functional want very simple method with simple parameters (data-obj)
 - ✓ *Find the method by looking on screen which action I want to do*
- Scenario validate (or demonstrate) a use case
 - ✓ *Documentation associated describe it*

Organize, web generation

- Easy to write, integrated in development process (commit, release, ...)
- Creating for sharing and re-usability
- Easy to create link between method

Associated to code

- Easy to check if up to date or not
- Technical image but usable even by Keys Users after explanations



Writing for the future reader

- Functional consultant or Product Owner (or Key users)
 - ✓ *Maybe package for test should be by "Application / Menu"*
 - ✓ *It's more easy for a reader to know what he is waiting for*
- Technical too
 - ✓ *For technical points*
 - ✓ *For understanding use case by example*
- Understanding or/and Debugging



Unit java method

- Junit test during developpemnt
 - ✓ *Actions and Test methods*
- Javadoc
 - ✓ *First release*

Test Data xml file

- Data-obj structure

Log message

- Log message
 - ✓ *Debug*

Other

- Technical error
- Bug analysis

Iteration

- Add details

Scenario method

- Use Case definition or first test
 - ✓ *Using or defining action and test method*
- Testing documentation
 - ✓ *Add more details*

Use Case context

- Real or demo data

ShowInfoPanel

- Follow the actions
 - ✓ *User interaction*

Other

- User process error
- Parameters or Data error

Iteration

- Add (or ask to) bundle actions method

Collaboration for all actors type

- Each feature enhancement is a derivative of an existing Use Case
 - ✓ *Using corresponding scenario test to create the new one*
 - To explain context
 - To detail enhancement
 - ...
 - ✓ *Acceptance scenario can be at the beginning*
 - ✓ *Javadoc is directly the requirement*
- Simple proof of concept in enhancement definition
 - ✓ *Without development*

Scenario functionality coverage

- Business demonstration is more clear with multiple data
 - ✓ *Special data type*

Demo data become dynamics

- As part of a process



Business and ERP context

- O'Toit
- ERP, Actors type, Functional Consultant
- GUI Test, requirement and cost

Technical context and definition

- OFBiz Selenium Webdriver Test reminder & summary
- Junit and Functional test
- Tests & Documentations

Collaborative Approach

- Unit test & Scenario test
- Javadoc as Help
- When and How

Tutorial & Implementation

- Help & Maintenance
- Modularity
- More contributor

Feature == Use Case

- An example is better than a long explanation
 - ✓ *Need a lot of ShowInfoPanel or Clear JavaDoc*
 - ✓ *Need clear demo data*
- Video is up to date even if it run only one time per week or month
 - ✓ *Easy to reproduced for the user*

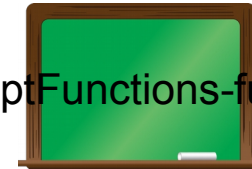
Help can focus on concept

- With list of example
 - ✓ *Very efficient when there are a lot of case*
 - Promotion type
 - Feature category associated to Product Category => filter type on front office
- Scenario is used to show all the details

Possible to enhance Scenario Test

- `zoomAndSendKeys(id, text)`
 - ✓ *Even if method is simple*
 - Directly unit action method modification
 - Ask to insert at the right place

04-TestJavaScriptFunctions-functionnalTest.avi



When test failed

- First analyst level : Functional consultant
 - ✓ *Retrieve / understand the context*
 - It's possible to connect to testing environment
 - ✓ *Using message and log and javadoc*
 - If not clear should complete existing
 - ✓ *If needed re-run with video*
- Technical is on support
 - ✓ *Technical error*
 - ✓ *Bug analyse*

Bug declaration

- Reproduced it with a scenario before trying to solve it
 - ✓ *Check existing test*
 - ✓ *Clarify the context*
- Formal description when resolution will not be immediately

Utils methods

- Back-Office is GUI homogeneous, so, corresponding test method exist

Unit actions and tests

- Same level as screen or sub-screen or portlet

Scenario to sub-part of process

- Include test-case, data-obj
- Technical can help

In Business environment

- Not necessary to test each screen in each place
 - ✓ *Multiply test type*
- If screen is never use in scenario, maybe it is not used in Business

OFBiz community and collaboration for non technical

- Usage and answer
 - ✓ *Help in requirement*
- Documentation
 - ✓ *Wiki*
 - ✓ *Help*
 - ✓ *Book*
- Demo data

Scenario test and / or tutorial

- Explain internal and external
 - ✓ *Usable in commercial process*
- Day to day evolution and enhancement
 - ✓ *First analyst level when it failed*
 - ✓ *Bugs or no more used*

A Collaborative Approach to Automated UI Testing in OFBiz



CONCLUSIONS

expertise in cooperation more than in techniques
Skill and Knowledge sharing

for implementation Team & Community are
the Keys Success Factors



The OFBiz Contributors Initiative

