Know the Difference

Unified Functional Testing (UFT) and Lean Functional Testing (LeanFT) from HP
Evolution in Functional Testing
Demanding a “shift-left” approach

Current challenges…

Agile and DevOps challenge testing processes
• Lack of proven agile testing approach and testing expertise within agile teams
• Applying test automation to agile projects
• Lack of right tools to build reusable test sets

Open Source and Continuous Integration Tools
• Agile & Dev/ Ops prefer open source
• “Good Enough” test automation
• Open integration in dev ecosystem is very important

Mobile Devices change definition of quality
• Mobile is everywhere
• Mobile development is “Agile development on steroids”
• User experience is EVERYTHING

…make it critical to “shift-left”

Improve development efficiency - Reduce the test burden - Accelerate application delivery

© Copyright 2015 Hewlett-Packard Development Company, L.P.  The information contained herein is subject to change without notice.
Challenges in the real world

Agile and DevOps

- 93% of IT leaders use Agile for software development projects
- 61% report a lack of a good testing approach that fits agile
- 55% can’t apply test automation at appropriate levels

Open Source

- $60B annual savings by businesses using open source
- $2B open source Software Sales*
- 140% increased interest in purchasing open source software since 2010
- 200% increase in open source projects in 2014 vs. 2012 (2M projects projected)

Mobile Devices / IoT

- 57% Global population estimated to be connected by 2019*
- 30B autonomous things estimated to be attached to the Internet by 2020*

Sources: World Quality Report 2014-2015; Sixth Edition; Capgemini, HP, Sogeti
“2015: Open Source Wins Over the Boardroom” (Object Computing Inc, ociweb.com)
What is UFT?

Unified Functional Testing from HP
A Modern Application Architecture
The challenge of testing complex interactions between many systems

**WHAT ARE THE LAYERS?**

- **Business Process Layer**
  (End-to-end business process testing)

- **Graphical User-Interface Layer**
  (GUI testing, user acceptance testing)

- **Headless Layer**
  (Web services, APIs, components, unit, integration)

**WHY TEST AUTOMATION?**

- **Unites People, Process and Information with technology**
- **Key point to validate user experience, and check many functional aspects of the GUI**
- **Independently validate services and components crucial for integration testing earlier and faster in the lifecycle**

**WHO DOES IT?**

- **Non-Technical / High Business Knowledge**
  - Subject Matter Expert
  - Business Analyst
  - Quality Assurance

- **Technical / Advanced Technical Knowledge**
  - Test Automation Engineer
  - Developer
  - Quality Assurance
Unified Functional Testing (UFT)
The complete, industry-standard solution for automation of web, mobile, API and packaged applications

**Cross-browser Testing**
Script once, replay all tests across browsers including Chrome, Firefox, IE, and Safari

**Continuous Testing**
UFT includes the Lean Functional Testing (LeanFT) plugin for test creation in Visual Studio/C# or Eclipse/Java

Broad ecosystem of integrations for source control management, CI tools, and HP tools such as HP Sprinter and HP Service Virtualization

Supports over 30 different technologies, more than any competitor in the market

**API & Web Service Testing**
Create tests beyond the UI for increased velocity

**Mobile Application Testing**
Ensure mobile app quality via integration with HP Mobile Center

**Business Process Testing (BPT)**
Use the BPT testing framework for keyword-driven and scriptless test automation
Unified Functional Testing (UFT)
Robust integrations accelerate testing

HP ALM
HP Quality Center
HP Mobile Center
BPT Accelerators
HP LoadRunner
HP Sprinter
HP Service Virtualization
Business Process Monitoring (BPM)

Jenkins
Subversion (SVN)
GIT
UFT with Business Process Testing (BPT)

A robust, scalable test framework for enterprise automation

**Scriptless testing**
Rich keyword driven (KWD)
Component scanning and creation

**Data separation**
Data separation from BPT logic for increasing testing coverage
Synthetic data generation to accelerate data creation time

**Acceleration**
Auto and smart componentization
Auto test flow creation

**Maintenance**
AUT auto changes detection and fix for component based and flows
Smart reuse mechanism to increase test flow ROI
UFT v12.5
What’s New: Features and Enhancements

UI improvements
Improved usability with only the relevant toolbars showing
New lightweight, HTML-based run results reports
Keyword view enhancements
New start page look and feel

Cross-browser testing
Improved Chrome recording
Improved performance for across multiple browser types
Improved Firefox browser testing enhancements

SiebelOpenUI improvements
Object hierarchy for objects
New objects: SblOUIApplet and SblOUIDropDown button

Business Process Testing (BPT)
Canvas view for BPT test
Data use improvements: import and export from Excel and map to parameters
Scan components and automatically parameterize objects that are scanned
Recording of components in a BPT test
Support for Mobile

Test execution and maintenance
UFT Execution Engine – run UFT tests without installing the entire UFT IDE
Integration with GIT source code repositories – store and version test scripts

Mobile
Checkpoints for verification while recording mobile tests
Component based testing with BPT
What is LeanFT?

Lean Functional Testing from HP
Traditional Testing
Focus is placed at the end of the software development cycle

Unit Testing → API / Service Testing → User Interface Testing

Traditional test automation is mostly based on the front end…

…but results in a focus at the end of the software development cycle
Agile Testing
Agile and DevOps moves focus to the left

Agile’s focus on continuous testing and continuous delivery causes testing to “shift left”, occurring much earlier in the development lifecycle.
The Problem…
The Automation Engineer

I have poor test execution coverage.
WHY?
High TCO doesn’t allow additional license purchase so running full coverage takes ages

I don’t have enough time to create and maintain my tests.
WHY?
Heavy tool
Legacy language / IDE
No collaboration with dev

I’m considered a dinosaur and my influence is decreasing.
WHY?
Negative tool perception by the developer community
Tool is not open enough to meet the Continuous Delivery practices
The Problem...

The Dev/Tester

My tests are fragile. Doing the RCA and Test Maintenance takes too much time.

**WHY?**
- Fragile object recognition based on Xpath only
- No good reporting/checkpoint

It’s too challenging to build a good test covering the business flow end-to-end.

**WHY?**
- Can’t test beyond the web
- Weak data driving
- No API accelerators
- Weak test auditing

The learning curve is too long.

**WHY?**
- Open Source solution is not straightforward and requires high ramp-up
Introducing Lean Functional Testing (LeanFT)
Built specifically for continuous integration and continuous testing

**LeanFT** is a powerful yet lightweight functional test automation solution, that supports a wide range of AUT technologies.

Targeted to **technical automation engineers** and **dev-tests** in Agile teams, LeanFT is fully embedded in standard IDEs and integrates naturally with the Dev and QA ecosystems.
LeanFT: IDE Integration

LeanFT **fully integrates** with and provides plugins for the standard IDEs:

Tests are authored in the IDEs using **C#** (in Visual Studio) or **Java** (in Eclipse)

The LeanFT **library** is used in the scripts to define the interaction with the AUT

Tests are authored in the context of any testing framework, or in a custom framework

The user can leverage the existing IDE capabilities (e.g.: IntelliSense & debugging)

The .NET / Java libraries and 3rd party libraries can be used to extend the scripts

Dev & QA share the same tool (IDE), enabling **better collaboration in Agile teams**

LeanFT **plugins extend** the IDEs with:

LeanFT project templates for standard unit-testing frameworks (NUnit, MSTest and JUnit)

LeanFT tools:

**Object Identification Center**
A central tool for any object identification related task

**Application Models**
Enables modeling of the AUT and its objects
LeanFT: Technology Support  
Based on UFT’s technology support

LeanFT supports the most common AUT technologies

• Windows Standard
• Web
• .NET Windows Forms
• WPF
• Mobile
• Insight Image Recognition
LeanFT: How it works
An end-to-end usage flow

1. In the IDE, create a testing project:

- LeanFT plugin provides OOTB project templates for standard Unit Testing Frameworks (NUnit & MSTest in VS, JUnit in Eclipse)
- It is also possible to use the LeanFT in other testing frameworks or in a custom framework

2. A project is created with the LeanFT and testing framework* libraries already referenced:

* The testing framework must be installed on the machine
LeanFT: How it works
An end-to-end usage flow

3 Implement the test using the LeanFT library:

```csharp
[Test]
public void VerifyTabletPrice()
{
    IBrowser browser = BrowserFactory.Launch(BrowserType.InternetExplorer);
    browser.Navigate("hp.com");

    var tabletsLinks = browser.Describe<ILink>(new LinkDescription { InnerText = "Tablets", Index = 4 });
    tabletsLinks.Click();

    var tablet1PriceElement = browser.Describe<IWebElement>(new CSSDescription("#modelBox_135512 > div.item1column"));
}
```

4 Add additional classes / tests.

5 Build the project.
LeanFT: How it works
An end-to-end usage flow

6 Execute the tests with the testing framework’s runner, to make sure the tests run as expected:

7 Check the source into the SCM.

8 Execute the tests either from the CI system or from ALM
Skillset and Task Comparison
When to use UFT, When to “go lean” with LeanFT

| Role                                      | UFT + BPT                                                                 | LeanFT                                                                 
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Analyst</td>
<td>Needs more wizard-based options</td>
<td></td>
</tr>
<tr>
<td>Subject Matter Expert (SME)</td>
<td>Needs a more keyword-driven and/or scriptless approach to test automation</td>
<td></td>
</tr>
<tr>
<td>Test Automation Engineer</td>
<td>Builds frameworks and is comfortable with creating test automation in UFT and some skill using other languages</td>
<td></td>
</tr>
<tr>
<td>QA Analyst</td>
<td></td>
<td>DevTest (Developer/Tester)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Works with popular dev languages such as Java, C#, Ruby, Python, etc.</td>
</tr>
<tr>
<td>Tester</td>
<td></td>
<td>Continuous Testing, Agile, Dev/Ops Centric</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Teams that typically have a more developer centric focus for continuous delivery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Test Automation Engineer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Builds frameworks and is comfortable with creating test automation in UFT and some skill using other languages</td>
</tr>
</tbody>
</table>

Vendor Tested Applications (VTA)

- Oracle
- SAP
- Powerbuilder
- Banking Apps
- Mainframe
UFT and LeanFT: How they are similar

UFT knowledge can be leveraged when using LeanFT
LeanFT enables creating robust tests that deal well with changes in the application

LeanFT is based on existing UFT concepts, mechanisms & tools:

From UFT… | LeanFT
---|---
Adopts the UFT concepts of **Test Objects** and **Descriptions**

Uses UFT’s **Object Identification** mechanisms
Property-based identification, Ordinal Identifiers (Index), VRI, XPath Identification, etc..

Adopts and enhances UFT’s **tools** (e.g. Spy and **Object Repositories**)

23 © Copyright 2015 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice.
## UFT vs. LeanFT: Feature Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>Unified Functional Testing (UFT)</th>
<th>Lean Functional Testing (LeanFT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Develop tests in Visual Studio using C#</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Develop tests in Eclipse using Java</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Open Source TDD &amp; BDD Framework Integration</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>(Cucumber, Fitness, Junit, Nunit, etc.)</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Source Control &amp; Version Control tools</td>
<td>Subversion &amp; GIT only</td>
<td>✅</td>
</tr>
<tr>
<td>Application Models</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Visual API Testing/Web Services Testing</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Business Process Testing (BPT) Integration</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Record &amp; Playback</td>
<td></td>
<td>Scripting only</td>
</tr>
<tr>
<td>Insight (image based object capture)</td>
<td></td>
<td>✅</td>
</tr>
<tr>
<td>Supported Test Platforms</td>
<td>20+ platforms for legacy, desktop, mainframe, windows, web, mobile and supported, including packaged apps, such as SAP</td>
<td>Windows Standard, Web, .NET Windows Forms, WPF, Mobile, Insight Image Recognition</td>
</tr>
</tbody>
</table>
Automated Functional Testing Offerings

Scenario 1: **Two for One**
Existing UFT customers get LeanFT for free

Scenario 2: **Mix and Match**
Buy either full UFT (w/LeanFT included) or LeanFT only

Scenario 3: **Go Lean**
Just LeanFT for CI / CD and small Dev/Test teams
High Velocity Innovation

... applying UFT and LeanFT puts DevOps on the path to Continuous Assessment

![Diagram of continuous assessment cycle]

- **Build**: Continuous Integration & Testing
- **Release**: Continuous Delivery & Deployment
- **Operate**: Continuous Operations
- **Test**: UFT and LeanFT
- **Deploy**: FASTER Time to Market, Higher Predictability, LOWER Costs, EXCELLENT User experience

- Faster Time to Market: Shorter Application release cycles
- High Predictability: End to end visibility
- Lower Costs: High resource utilization, reduced rework cycles
- Excellent User experience: High application quality

Business Demands
Deliver amazing apps with confidence

Accelerate application delivery with the power of big data

- Application Lifecycle Mgmt
- Quality Center
- Agile Manager
- ALI (App Lifecycle Intelligence)

Test across multiple devices / OS without compromising time to market

- Lean Functional Testing
- Unified Functional Testing
- Business Process Testing
- Sprinter
- Mobile Center

Deliver beautiful experiences that perform and scale

- LoadRunner/Performance Center
- StormRunner Load
- Service and Network Virtualization
- App Pulse Mobile
Thank you