ttc
the testing consultancy
A Mosiac Group Company

Accelerating software testing effectiveness using Agile methodologies..

“How can testing be completed faster, and more efficiently, within short iterations?”

The Problem

It is a painful paradox that while the workplace has become fast paced today because of technological innovations, the development team is also under enormous pressure to deliver on schedule. The unending pressure to accelerate the delivery of shippable products or a working software can also be impacted by the quality assurance process which includes software testing. In Agile, the slice of work a team commits to delivering cannot be considered done until tested. However, except for Test Driven Development (TDD), it’s even harder for testing to be completed in time boxed sprints, because in most cases programmers deploy code for testing halfway or late in the sprint.

This is why accelerated testing could answer some of the paradoxical questions (especially around expectation and capability).

This whitepaper will attempt to offer practical solutions that could ensure prompt delivery of testing objectives and deliverables, so that the goal of a working software can be achieved faster incrementally for overall business value. Using the available resources there are some tips that can help to get testing done faster in an Agile environment….

1. Co-locate your Teams

The benefit of co-locating is more visible in Agile Testing than any other model. One of the core Agile Principles is to **welcome changing requirements,** even late in development. The process is to harness change for the customer's competitive advantage. It will be harder to accomplish and continuously adopt the changing requirements’ principle in a competitive market if testing is slow paced or impacted because time for testing is too short. When a team is co-located it will be easier and faster for test preparation, planning, and execution to be delivered as feedback and clarification required around acceptance criteria with the product owner can be achieved in real time. Through co-location, rework and confusion that characterise large scale projects can be eliminated.

When team members and Test Consultants take advantage of collocation and are able to convince and convey their opinions and suggestions efficiently testing becomes even more effective and accelerated. Agile Testers are expected to initiate communication so that gaps are bridged within and outside the development team.

Another Agile principle states **"the most efficient and effective method of conveying information to and within a development team is via face-to-face conversation."** Any development or testing team that is co-located enjoys continuous communication prior to and during development, test planning, test preparation and test execution. Communication gaps that slow down testing can be shutdown for testing to be accelerated when a development team and testers are co-located. It is however better to seek the services of a trusted testing services provider such as The Testing Consultancy (TTC) who can provide experienced and trained Agile Test Consultants.

Other factors that enable Test Consultants to accelerate testing (when co-located) are;

- They understand the business ecosystem, priorities and the development environment before and during testing
- They enjoy the benefits of Continuous Integration, Test Driven Development (TDD) and Behavior-Driven Development (BDD)
- Ability to leverage Scrum, Extreme programming, Rational Unified Process and KanBan practices
- They take full benefits of sustainable development through strengthened iterative coding and improved infrastructure

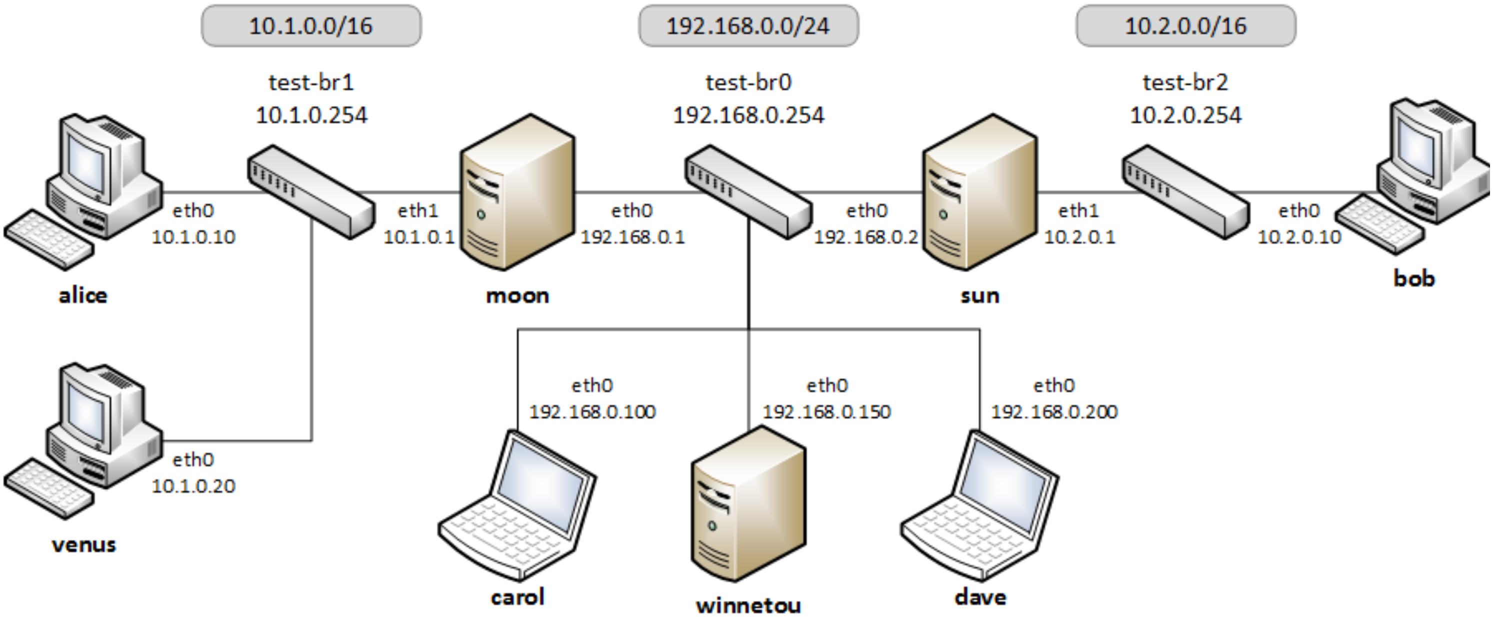
2. Virtualize Test Environments

In line with an Agile Principle we must build projects around motivated individuals, give them the environment and support they need, and trust them to get the job done. When struggling to set up a physical testing environment, the solution is to be Agile in the real sense of it and look in the direction of virtualized test environment.

Getting the physical development and test environments ready for the feature team to use for software development and testing could be expensive and time consuming, especially for medium size Agile projects that do not have a component team working one sprint ahead of the feature team. This will greatly impact the progress and pace of testing in this Agile age. Why?

- When hardware configuration is modified the image must be updated (as it's hardware specific)
- When hardware configuration fails, all data and test results can be lost and become a potential risk for testing and the project.

The following is an example of a virtualized test environments:



Virtualizing development and test environments can accelerate testing …

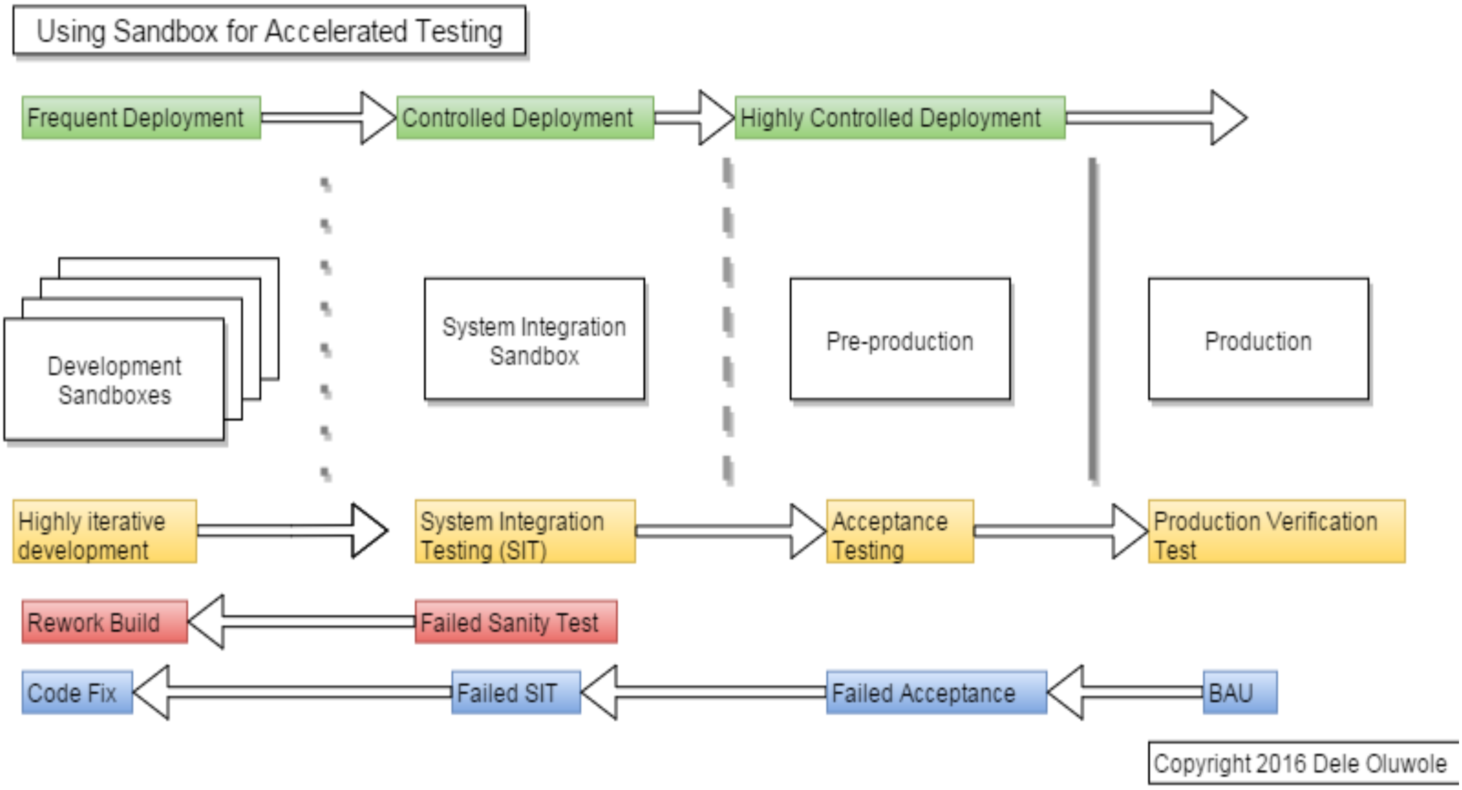
- If there are enough resources, more than one virtual machine (VM) on a single piece of hardware can accelerate testing especially automation and exploratory.
- VM can help speed up debugging issues (using snapshots).
- There is no hardware failure with virtualized infrastructure. VMs are files that can be backed up in order to minimise loss whenever a server crashes, and testing can be stable and faster when there is no hardware failure.
- There will definitely be no need to reconfigure and recreate images when new hardware are introduced. It's as simple as adding the new one in to the pool of resources for testing to continue unhindered.

- Virtualizing a test environment can accelerate completion of test automation, at the same time test automation can be used to efficiently achieve testing objectives faster with wider coverage. This is because with minimal effort VMs can be replicated quite quickly using existing automation scripts.

3. Use Sandboxes

Sandbox is an isolated computing environment in which a program or file can be executed without affecting the application in which it runs. Sandboxes are used by software developers to test new programming code. Isolated "sandbox" environments for application development and pre-deployment testing will reduce the time and cost for setting up a physical development and test environments that take forever to sysadmin, configure, and implement.

The following diagram can be used to achieve faster testing and immediate feedback loop because testing can be done without affecting other integrating applications. Once the test environments, phases, and feedback loop are implemented as seen in the diagram the testing process, objective, and deliverable becomes easier to accelerate and accentuate for the development team and project stakeholders.


Using Sandbox for Accelerated Testing
Frequent Deployment
Controlled Deployment
Highly Controlled Deployment
Development Sandboxes
System Integration Sandbox
Pre-production
Production
Highly iterative development
System Integration Testing (SIT)
Acceptance Testing
Production Verification Test
Rework Build
Failed Sanity Test
Code Fix
Failed SIT
Failed Acceptance
BAU
Copyright 2016 Dele Oluwole

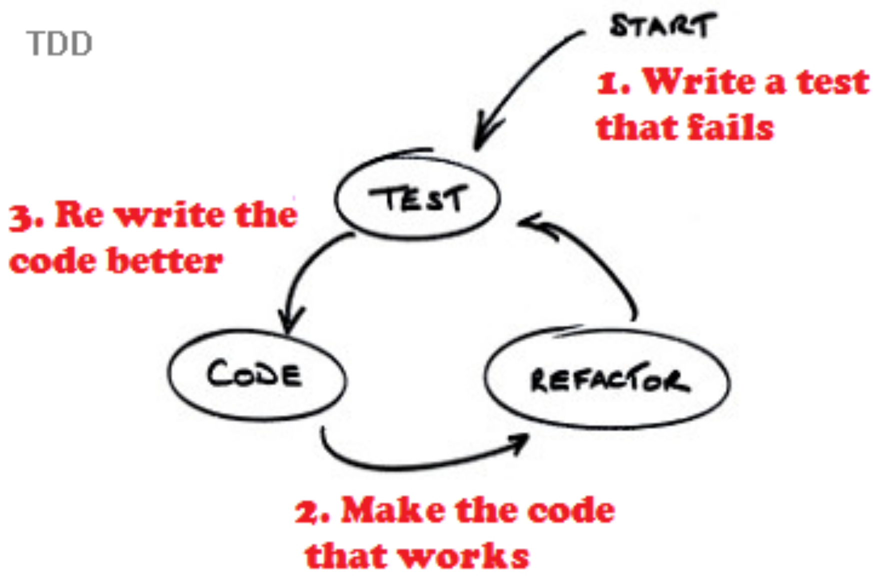
4. Try Test Driven Development (TDD)

TDD is an Agile technical practice that can be used to accelerate testing within a short sprint because of its short feedback loop and reduced time in rework. Less time is also spent in debugging and even when required, testers usually get closer to problem quickly, thereby reducing the rework, retesting time.

Consultants at TTC have found out that the more levels of testing (unit, integration, system, and UAT) the higher the level of confidence on the product. However, one way to significantly reduce the volume of defects is to have developers practice Test Driven Development, i.e. build in quality earlier so testing can be completed faster.

TDD is very good for immediate feedback. A team will have almost immediate feedback on the components and applications they are developing and testing. The shorter feedback loop enables a faster turnaround on testing and defect resolution compared to traditional waterfall methodology where code is tested days or weeks after deployment.

TDD definitely can increase the testing pace, avoid scope creep and prevent unwarranted design or components from sneaking into the product. The test cases define the exact set of required features. With TDD it is easy to identify redundant code, detect and remove irrelevant development and testing tasks.



Adaptation, Dele O. 01/ 2016

5 Stub Unavailable Integrating Components

"Responding to change over following a plan" - Agile Manifesto.

When the desired is not achievable, an Agile mind thinks of a good alternative, and that's responding to change over following a plan. This literally means when the required integrating module/data is unavailable the ideal thing to do is to use stub so that early development testing can be completed unhindered.

A stub is a small piece of code that temporarily takes the place of another component (being expected to be integrated) during testing. One outstanding benefit of using a stub is that it returns consistent results, making the test easier to write and execute. Tests can be run progressively while test scripting and even when the required components are either not working as expected or unavailable.

It can also serve as a placeholder that can accelerate testing when the required component is still being developed, thereby saving time and money that would have been used waiting for work-in-progress components.

6. Integrate Continuously and Automate

Scaled Agile Framework (SAFe), a knowledge based community believes that Continuous Integration is perhaps the most essential technical practice of an ART and Value Stream, a quality heartbeat that reduces risk and establish **fast**, sustainable development. Continuous Integration is an Agile Technical practice of checking in code in small increments and testing it against the larger

code. Being able to continuously deliver quality code that works in production or anywhere else is highly dependent on automated continuous integration.

Continuous integration system could be a complex task to drag to 'done' on the Agile board, especially with embedded tasks such as connecting the Development, Test, and Staging environments. It could be even more challenging and time consuming to 'sysadmin', provision, configure and maintain these environments. Aside the challenges of setting up these environments and getting them to work, to make them mirror the production environment.

Automating the continuous integration process and tests is the key to cutting these tasks down to size. Just as automation makes server provisioning and patch management easier and more reliable, it can enable the development team deliver code faster.

7. Automate the Regression Tests

There is a general fear about shipping code frequently into test environments. The fear emanates from the high risk of breaking working code after deploying new code. Inasmuch as it's a realistic fear most software is complex, often reaching into multiple parts of an organisation. It will suffice to point out that a low severity defect at first can eventually have far-reaching consequences on the application and the project.

In order to deliver code incrementally and to be able to test as they are delivered for increased level of confidence, the regression script MUST be automated so as to achieve reduced testing time that can fit into a sprint delivery goal.

TTC has assembled a team of experienced test Consultants which are skilled in Agile and automation tools (TTC maintains a core of permanent staff who are specifically trained in the Model Based Testing Tool,TOSCA TestSuite since 2009). TOSCA provides a unique benefit to clients in that it separates the business and technology elements in a highly intuitive User Interface.

About The Testing Consultancy

Founded in 2004, TTC seeks to work on interesting projects across the Asia Pacific region with smart people to solve 'real world' software quality problems. Our approach is grounded and pragmatic which, when blended with world class thinking and experience, provides us with great confidence to assist firms of all sizes, industries and level of maturity…

New Zealand: +64 9 948 2225   Singapore: +65 9822 6679

www.testingconsultancy.com