

Introduction to Hands-on-Labs for building out a Continuous Delivery Release pipeline with Team Foundation Server



Table of Contents

- Objectives 2
- Contents of the HOLs 3
 - Lab Course Map – how to walk through them 4
 - Core Path – Brian Keller VM..... 4
- Brian Keller VM Overview 4
- Check Off List for the HOLs 4
 - Operating Systems 5
 - Tools..... 5
 - SDKs..... 6
 - Third-party Libraries..... 6
- Setup Information 6
 - Operating System Links..... 6
 - Windows 7 6
 - Windows Server 2008 R2 7
 - Windows 8 7
 - Windows Server 2012 7
 - Windows Azure 7
 - Tools Links..... 7

Visual Studio 2013.....	7
Team Foundation Server 2013.....	7
TFS Build Controller.....	7
Test Controller and Agents	8
Microsoft Test Manager.....	8
WiX Tool Set.....	8
TFS Build Extension	8
SDKs.....	9
Windows Phone 8 SDK.....	9
Windows Azure SDK.....	9
ALM Rangers Material.....	9
Standard IIS Configurations	9
Copyright.....	11

Objectives

This series of labs teaches you how to implement a continuous delivery release pipeline by using Team Foundation Server 2012. You begin by building a simple release pipeline that supports continuous integration, but has no other automation. Subsequent labs modify and extend the pipeline until it supports continuous delivery and includes features such as automated deployments and automated testing.

A good starting point is reading the book that the labs are based from patterns & practices guide [Building a Release Pipeline with Team Foundation Server 2012](#), available online. This guidance emphasizes four patterns that are critical to developing a continuous delivery pipeline and whose importance is reinforced by the hands-on labs (HOL). The patterns are:

- Orchestration
- Automation
- Monitoring
- Measuring

Even as new technologies become available, the underlying patterns that shape your development efforts will remain the same.

This document is an introduction to the labs and discusses the following topics:

- Contents of the HOLs.
 - Prerequisites for the HOLs.
 - Where to find more information about the prerequisites.
 - ALM Rangers Material
 - Standard IIS Configurations.
-

Contents of the HOLs

There are six labs themes in this series although most of them are actually collections of sub-labs underneath them.

One the biggest challenges in learning about distributed architectures such as the ones required in building out Release Pipeline architecture is dealing with infrastructure issues. The details in setting up and tearing down infrastructures becomes a serious time impediment in learning. To help you get to cover the core learnings the overall lab is organized with 2 levels. Starting with Lab 1 you work core TreyResearch.SLN which will works on a [Brian Keller VM](#) so there is minimum infrastructure to deal with and you can focus on the learning. In Lab 6, there exist an advance part that extends the learning of the Release Pipeline to work with Windows Phone 8 and WCF Azure Service.

- **Lab 1: Creating the Trey Research Environment.** In this lab you install the Trey Research application and create the initial version of the Trey Research release pipeline.
- **Lab 2: Orchestrating the Trey Research Pipeline.** This is a series contains 4 sub-labs. In the first sub-lab you orchestrate the commit stage of the pipeline. In the second sub-lab you orchestrate the rest of the pipeline stages. In the third sub-lab you configure the pipeline. In the fourth sub-lab you test the orchestration.
- **Lab 3: Automating the Trey Research Pipeline.** This is a series contains 4 sub-labs. In the first sub-lab you automate the deployment of the WCF service. In the second sub-lab you automate the deployment of the WPF application. In the third sub-lab you automate the tests. At the conclusion of this sub-lab you have created a continuous delivery pipeline.
- **Lab 4: Monitoring and Measuring the Trey Research Pipeline.** This is a set of 2 sub-labs. In the first sub-lab you learn how to monitor a continuous delivery pipeline. In the second sub-lab you learn how to create custom reports in Microsoft Team Foundation Server (TFS) to track some of the key metrics used with continuous delivery projects.
- **Lab 5: Extending the Trey Research Pipeline.** In this lab you learn how to add new stages to the pipeline.

There are completed-lab sections for the various labs. Allowing you to see a final solution. Ensuring you don't get lost.

Lab Course Map – how to walk through them

There are 2 ways to work through the labs as mentioned above. The following lays out how to walk through the Core Labs on Brian Keller VM and how to walk through the Advance Labs.

Core Path – Brian Keller VM

1. Introduction.docx
2. Lab01 – Starting Point
3. Lab2_1 – Orchestrating the Commit Stage
4. Lab 2_2 – Orchestrating the Remaining Stages
5. Lab 2_3 – Configuring the Pipeline
6. Lab 2_4 – Testing the Orchestration
7. Lab 3_1 – Automating the Deployment of WCF Service
8. Lab 3_2 – Automating the Deployment of the WPF app
9. Lab 3_3 – Running the Automated Tests
10. Lab 4_1 – Running and Monitoring the Continuous Delivery Pipeline
11. Lab 4_2 – Metrics for Continuous Delivery in TFS
12. Lab 5 – Evolving the Pipeline by adding new stages

Brian Keller VM Overview

The VM gives you a strong starting and stable starting point. The virtual machine is fully configured with Team Foundation Server, Visual Studio Ultimate and the necessary software and tools that you need to get running. Without having to give any thought about the supporting infrastructure or spend time installing and configuring the necessary software.

Check Off List for the HOLs

Here are the items you'll require for the HOL. Note most of them will be on [Brian Keller VM](#). For the Advance lab you'll have to setup a machine to support the various tools. The labs are designed to give you options about the technologies and number of computers that you want to use. For example, you can deploy to a Windows Azure virtual machine (VM) to run an IIS service if you want, but you can run the labs on either a local or remote IIS machine.

To use these HOLs, you should be familiar with the technologies listed here.

Operating Systems

Item	Brian Keller VM
Windows 7	Yes
Windows Server 2008 R2	Yes
Windows Server 2012	
Windows 8	

Tools

Item	Brian Keller VM
Microsoft Visual Studio 2013	Loaded
Team Foundation Server 2013 (TFS)	Loaded
TFS build controller	Loaded
Microsoft Test Manager (MTM) 2013	Loaded
Test controller and agents 2013	Loaded

Microsoft Office Excel	Loaded

SDKs

Item	Brian Keller VM
Windows Phone SDK	no

Third-party Libraries

Item	Brian Keller VM
TFS Build Extensions	No, but in lab 1 Exercise 6 you install it.

Setup Information

This section contains links where you can find more information about the required items listed in the previous section.

Operating System Links

These links give information about how to develop applications for each operating system.

Windows 7

The Windows 7 Training Kit for Developers includes presentations, hands-on labs, and demos designed to help you learn how to build applications that are compatible with Windows 7. For more information

see Windows 7 Training Kit for Developers at <http://www.microsoft.com/en-us/download/details.aspx?id=6450>.

Windows Server 2008 R2

For more information about Windows 2008 R2, see Windows Server 2008 R2 at <http://technet.microsoft.com/en-us/windowsserver/bb310558.aspx>.

Windows 8

The Windows Dev Center is a great place to start learning about Windows 8. For more information see Dev Center — Windows Store apps at <http://msdn.microsoft.com/en-us/library/windows/apps/br211386.aspx>.

Windows Server 2012

For more about Windows Server 2012, go to Windows Server 2012 at <http://www.microsoft.com/en-us/server-cloud/ws2012/default.aspx?EP=200072478>.

Windows Azure

The Windows Azure team has created a great deal of documentation. To sign up for a Windows Azure subscription or a trial, see Windows Azure Purchase Options at <http://www.windowsazure.com/en-us/pricing/purchase-options/>. You will also need to use the Windows Management Portal so that you can configure your Windows Azure service namespaces. For more information about Windows Azure, go to the product website, which has a link to the documentation, at <http://www.windowsazure.com/en-us/>. One tutorial that may interest you is Create a Virtual Machine Running Windows Server at <http://www.windowsazure.com/en-us/develop/net/tutorials/windows-virtual-machine/>.

Tools Links

The following links give information about the various tools.

Visual Studio 2013

Visual Studio is the central tool for developing code on the Microsoft platform. For more information, go to the Visual Studio website at <http://www.microsoft.com/visualstudio/eng/products/visual-studio-overview>.

Team Foundation Server 2013

Team Foundation Server 2013 is the collaboration platform at the center of Microsoft's application lifecycle management solution. For more information, go to the Team Foundation Server website at <http://msdn.microsoft.com/en-us/vstudio/ff637362.aspx>.

TFS Build Controller

To learn more about how to configure and manage your build controller see Configuring and Managing Your Build System at <http://msdn.microsoft.com/en-us/library/vstudio/ms252495.aspx>.

Test Controller and Agents

To learn more about test controllers and agents see [Setting Up Test Machines to Run Tests or Collect Data](http://msdn.microsoft.com/en-us/library/dd293551.aspx) at <http://msdn.microsoft.com/en-us/library/dd293551.aspx>.

Microsoft Test Manager

The MTM does both Lab Management and Test Plan Management. Both capabilities are used during various parts of the HOLs. The MTM works with Team Foundation Server using the build and test controller to configure and control machines forming your Lab Environments used for manual and automated testing. The Lab Management handles the management of creating, modifying and deleting the Lab Environments. The Test Plan Management deals with the Test Plan which organizes the various manual and automated test to be run on the designated lab environment for a selected build. For more information about these tasks, you can refer to the following resources.

- [Setting Up Test Controllers in Lab Environments](http://msdn.microsoft.com/en-us/library/hh546460.aspx) at <http://msdn.microsoft.com/en-us/library/hh546460.aspx>.
- [Getting Started with Lab Management](http://msdn.microsoft.com/en-us/library/ee943321.aspx) at <http://msdn.microsoft.com/en-us/library/ee943321.aspx>.
- [Configuring and Administering Lab Management](http://msdn.microsoft.com/en-us/library/dd936084.aspx) at <http://msdn.microsoft.com/en-us/library/dd936084.aspx>.
- [Creating Lab Environments](http://msdn.microsoft.com/en-us/library/dd380688.aspx) at <http://msdn.microsoft.com/en-us/library/dd380688.aspx>
- [Using a Lab Environment for Your Application Lifecycle](http://msdn.microsoft.com/en-us/library/dd997438.aspx) at <http://msdn.microsoft.com/en-us/library/dd997438.aspx>.
- [ALM Rangers' Visual Studio Lab Management Guide](http://vsarlabman.codeplex.com/) at <http://vsarlabman.codeplex.com/>.

WiX Tool Set

This lab uses the WiX tool set. WiX allows you to build Windows installation packages from XML source code. WiX supports completely automated installations and uninstallations, and is integrated with Visual Studio and MSBuild. For more information about the Windows Installer, see [Windows Installer](http://msdn.microsoft.com/en-us/library/windows/desktop/cc185688(v=vs.85).aspx) at [http://msdn.microsoft.com/en-us/library/windows/desktop/cc185688\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/cc185688(v=vs.85).aspx). For more information about writing WiX files, refer to the [WiX tutorial](http://wix.tramontana.co.hu/) at <http://wix.tramontana.co.hu/> or the [WiX manual](http://wix.sourceforge.net/manual-wix3/main.htm) at <http://wix.sourceforge.net/manual-wix3/main.htm>.

Download it from CodePlex: <http://wix.codeplex.com/releases/view/99514>

TFS Build Extension

The Team Foundation Server Build Extension is a Ranger project on CodePlex that offer an number of useful workflow activities to construct your pipeline from the build workflow. This is used in Lab 2 to build out the orchestrations. <http://tfsbuildextensions.codeplex.com>

SDKs

Here's where you can download the SDKs that are required to complete the labs.

Windows Phone 8 SDK

You can download the Windows Phone 8 SDK at <http://www.windowsphone.com/en-us/how-to/wp8>
<http://dev.windowsphone.com/en-us/>.

Windows Azure SDK

You can download the Windows Azure SDK at <http://www.windowsazure.com/en-us/downloads/>.

ALM Rangers Material

The ALM Rangers have a great deal of material that you will find helpful as you create your release pipeline.

- Visual Studio ALM Rangers, Visual Studio Test Tooling Guidance:
<http://vsartesttoolingguide.codeplex.com/>
- Visual Studio ALM Rangers, *Visual Studio Lab Management Guide*:
<http://vsarlabman.codeplex.com/>
- Visual Studio ALM Rangers, *Team Foundation Build Customization Guide*:
<http://vsarbuildguide.codeplex.com/>
- Visual Studio ALM Rangers, *Visual Studio Team Foundation Server Branching and Merging Guide*: <http://vsarbranchingguide.codeplex.com/>
- Visual Studio ALM Rangers, *Visual Studio ALM Quick Reference Guidance*:
<http://vsarquickguide.codeplex.com/>
- Visual Studio ALM Rangers: <http://msdn.microsoft.com/vstudio/ee358786.aspx>

Standard IIS Configurations

This section is optional and is provided as a reference for Lab 3_1. To ensure that you have the correct configurations, you can consult the IIS Configuration Reference at <http://www.iis.net/configreference>. You should also make sure that the WCF Service is **ON**.

This is a list of all the IIS services that need to be installed in order for the WCF service to work correctly.

Display Name	Name	Install State
-----	----	-----
[X] Application Server	Application-Server	Installed
[X] .NET Framework 4.5	AS-NET-Framework	Installed

[X] Web Server (IIS) Support	AS-Web-Support	Installed
[X] File And Storage Services	FileAndStorage-Services	Installed
[X] File and iSCSI Services	File-Services	Installed
[X] File Server	FS-FileServer	Installed
[X] Storage Services	Storage-Services	Installed
[X] Web Server (IIS)	Web-Server	Installed
[X] Web Server	Web-WebServer	Installed
[X] Common HTTP Features	Web-Common-Http	Installed
[X] Default Document	Web-Default-Doc	Installed
[X] Directory Browsing	Web-Dir-Browsing	Installed
[X] HTTP Errors	Web-Http-Errors	Installed
[X] Static Content	Web-Static-Content	Installed
[X] HTTP Redirection	Web-Http-Redirect	Installed
[X] Health and Diagnostics	Web-Health	Installed
[X] HTTP Logging	Web-Http-Logging	Installed
[X] Logging Tools	Web-Log-Libraries	Installed
[X] Request Monitor	Web-Request-Monitor	Installed
[X] Performance	Web-Performance	Installed
[X] Static Content Compression	Web-Stat-Compression	Installed
[X] Dynamic Content Compression	Web-Dyn-Compression	Installed
[X] Security	Web-Security	Installed
[X] Request Filtering	Web-Filtering	Installed
[X] Basic Authentication	Web-Basic-Auth	Installed
[X] Client Certificate Mapping Authentic...	Web-Client-Auth	Installed
[X] Digest Authentication	Web-Digest-Auth	Installed
[X] IIS Client Certificate Mapping Authe...	Web-Cert-Auth	Installed
[X] IP and Domain Restrictions	Web-IP-Security	Installed
[X] URL Authorization	Web-Ur1-Auth	Installed
[X] Windows Authentication	Web-Windows-Auth	Installed
[X] Application Development	Web-App-Dev	Installed
[X] .NET Extensibility 3.5	Web-Net-Ext	Installed
[X] .NET Extensibility 4.5	Web-Net-Ext45	Installed
[X] ASP.NET 3.5	Web-Asp-Net	Installed
[X] ASP.NET 4.5	Web-Asp-Net45	Installed
[X] ISAPI Extensions	Web-ISAPI-Ext	Installed
[X] ISAPI Filters	Web-ISAPI-Filter	Installed
[X] IIS Hostable Web Core	Web-WHC	Installed
[X] Management Tools	Web-Mgmt-Tools	Installed
[X] IIS Management Console	Web-Mgmt-Console	Installed
[X] IIS Management Scripts and Tools	Web-Scripting-Tools	Installed
[X] .NET Framework 3.5 Features	NET-Framework-Features	Installed
[X] .NET Framework 3.5 (includes .NET 2.0 and 3.0)	NET-Framework-Core	Installed
[X] HTTP Activation	NET-HTTP-Activation	Installed
[X] Non-HTTP Activation	NET-Non-HTTP-Activ	Installed
[X] .NET Framework 4.5 Features	NET-Framework-45-Fea...	Installed
[X] .NET Framework 4.5	NET-Framework-45-Core	Installed
[X] ASP.NET 4.5	NET-Framework-45-ASPNET	Installed
[X] WCF Services	NET-WCF-Services45	Installed
[X] HTTP Activation	NET-WCF-HTTP-Activat...	Installed

[X] Message Queuing (MSMQ) Activation	NET-WCF-MSMQ-Activat...	Installed
[X] Named Pipe Activation	NET-WCF-Pipe-Activat...	Installed
[X] TCP Activation	NET-WCF-TCP-Activati...	Installed
[X] TCP Port Sharing	NET-WCF-TCP-PortShar...	Installed
[X] Message Queuing	MSMQ	Installed
[X] Message Queuing Services	MSMQ-Services	Installed
[X] Message Queuing Server	MSMQ-Server	Installed
[X] Remote Differential Compression	RDC	Installed
[X] User Interfaces and Infrastructure	User-Interfaces-Infra	Installed
[X] Graphical Management Tools and Infrastructure	Server-Gui-Mgmt-Infra	Installed
[X] Server Graphical Shell	Server-Gui-Shell	Installed
[X] Windows PowerShell	PowerShellRoot	Installed
[X] Windows PowerShell 3.0	PowerShell	Installed
[X] Windows PowerShell 2.0 Engine	PowerShell-V2	Installed
[X] Windows PowerShell ISE	PowerShell-ISE	Installed
[X] Windows Process Activation Service	WAS	Installed
[X] Process Model	WAS-Process-Model	Installed
[X] .NET Environment 3.5	WAS-NET-Environment	Installed
[X] Configuration APIs	WAS-Config-APIs	Installed
[X] WoW64 Support	WoW64-Support	Installed

Copyright

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet website references, may change without notice. You bear the risk of using it. Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

© 2014 Microsoft. All rights reserved.

Microsoft, Windows, Windows Server, Windows Vista, Windows PowerShell, Silverlight, Expression, Expression Blend, MSDN, IntelliSense, IntelliTrace, Internet Explorer, SQL Azure, SQL Server, Visual C#, Visual C++, Visual Basic, and Visual Studio are trademarks of the Microsoft group of companies.

All other trademarks are the property of their respective owners.