Microsoft Azure - Starter Kits for Partners

Introduction to Starter Kits

Development and Test Scenario

Last Update: March 2016





**MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.**

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The descriptions of other companies’ products in this document, if any, are provided only as a convenience to you. Any such references should not be considered an endorsement or support by Microsoft. Microsoft cannot guarantee their accuracy, and the products may change over time. Also, the descriptions are intended as brief highlights to aid understanding, rather than as thorough coverage. For authoritative descriptions of these products, please consult their respective manufacturers.

© 2016 Microsoft Corporation. All rights reserved. Any use or distribution of these materials without express authorization of Microsoft Corp. is strictly prohibited.

Microsoft and Windows are either registered trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

[Overview 4](#_Toc446688756)

[The Development and Test Problem 4](#_Toc446688757)

[The Proposed Solution 4](#_Toc446688758)

[The Value of Microsoft Azure for Development and Test 6](#_Toc446688759)

[Azure DevTest Lab 6](#_Toc446688760)

[Why DevTest Lab? 7](#_Toc446688761)

[Worry-Free Self-Service 7](#_Toc446688762)

[Quickly get to "Ready to Test" 7](#_Toc446688763)

[Create once, use everywhere 7](#_Toc446688764)

[Integrates with your existing Toolchain 7](#_Toc446688765)

[Partner Needs 7](#_Toc446688766)

[Introduction to Starter Kits 8](#_Toc446688767)

[Starter Kits - Partner Benefits 8](#_Toc446688768)

# Overview

In this document, we will cover the challenges our customers are facing with Development and Test scenarios, and how Microsoft Azure can solve this problem. Finally, we introduce the concept of a “Starter Kit”, essentially demonstrate how to leverage this concept of a packaged offering to accelerate the Partner sales and deployment cycle.

# The Development and Test Problem

Whether you are a developer and a CEO – at the same time - in a one-person startup getting your mobile app out the door, or a multi-national enterprise with 100,000 employees, you need a **sandbox to build, test and learn. In addition, you need a safe, isolated zone to do it**.

* Do you have **server** boxes under your **developers’ desks** where they create build environments and mimic production servers?
* Are you **upgrading** a line of **business application to Windows Server 2012** and need to stress test your stack top to bottom?
* Do you have customer support teams that need to **re-produce errors reported by internal and external** clients in order to resolve app problems?
* Does your sales team need to demo an app running at scale on SUSE Enterprise Linux in front of a customer?
* Do you need to build a prototype of a SharePoint site before you bid for a new project?

# The Proposed Solution

You can count on Microsoft Azure Infrastructure Services to quickly standup labs for testing and developing apps, validating app behavior, and creating presales, training, and teaching environments. Provisioning Virtual Machines in minutes gives, you access to a sandbox and a lab. Most development, test and lab environments are project-based and do not need to be up and running 24x7. When you are done, simply turn your virtual machines off. **You pay for what you use and no more**.

**Do you have server boxes under your developers’ desks where they create build environments and mimic production servers?**

Let Microsoft Azure Virtual Machines handle that for you - quickly. That server box under the desk does not need to run 24x7, be maintained all year long and refreshed every three years. Get access to pre-built Windows Server and many other Microsoft workload images in the gallery, pick from the open source community VM Depot or bring a VHD image of your own.

**Are you upgrading a line of business application to Windows Server 2012 and need to stress test your stack top to bottom?**

You can grow or shrink your sandbox and application load in real-time in Virtual Machines. You get to stress your application load and infrastructure capacity in order to mitigate issues before they begin. When you are done, move to on-premises or simply connect to on-premises via a virtual network.

**Do you have customer support teams that need to re-produce errors reported by internal and external clients in order to resolve app problems?**

**Do you need to build a prototype of a SharePoint site before you bid for a new project?**

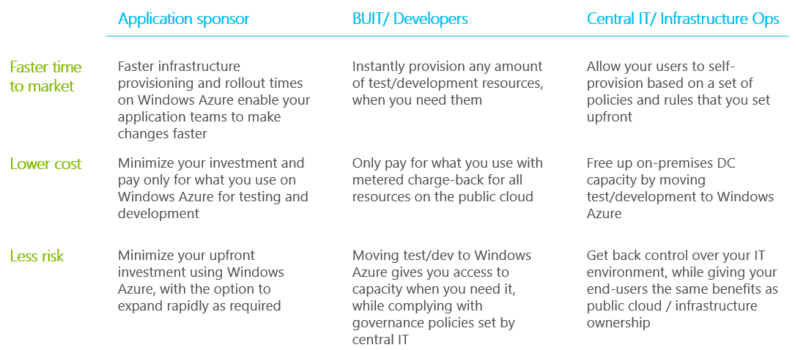
You do not have to dedicate on-premises hardware for these if you do not choose to; and you do not need to acquire new hardware if you are short on time or budget. This is where Microsoft Azure Infrastructure Services comes through.

Yet, it is not just about replicating infrastructure. Your developers need tools to work, too.

Sometimes it is about building new rich apps and reducing developer cycle times. When you want to reduce dependencies of your developer teams on IT operations for rolling-out test environments, consider running Team Foundation Server (TFS) in Virtual Machines. You can roll out farms on-demand and continue to access TFS from on-premises installed tools such as Visual Studio, Microsoft Test Manager and browser apps. Testers in your team also can configure and manage test labs using TFS Test Lab Management quickly.

Alternatively, when you want to collaborate with development partners who do not have access to your corpnet, TFS in Virtual Machines provides the right infrastructure.

# The Value of Microsoft Azure for Development and Test



We have seen over 40% growth in our all up enterprise cloud businesses with strong contribution from both hosting service providers, private cloud solutions and our own public cloud services.  Microsoft Azure alone is outgrowing the overall cloud market by >150 points.

By 2016, 45% of IT budgets will be devoted to cloud initiatives with 15% of that devoted to public cloud. (Source: IDC).

## Azure DevTest Lab

Developers and testers are looking to solve the delays in creating and managing their environments by going to the cloud. Azure solves the problem of environment delays and allows self-service within a new cost efficient structure. However, developers and testers still need to spend considerable time configuring their self-served environments. Also, decision makers are uncertain about how to leverage the cloud to maximize their cost savings without adding too much process overhead.

Azure DevTest Lab is a service that helps developers and testers quickly create environments in Azure while minimizing waste and controlling cost. You can test the latest version of your application by quickly provisioning Windows and Linux environments using reusable templates and artifacts. Easily integrate your deployment pipeline with DevTest Lab to provision on-demand environments. Scale up your load testing by provisioning multiple test agents, and create pre-provisioned environments for training and demos.

## Why DevTest Lab?

DevTest Lab provides the following benefits in creating, configuring, and managing developer and test environments in the cloud

### Worry-Free Self-Service

DevTest Lab makes it easier to control costs by allowing you to set policies on your lab - such as number of virtual machines (VM) per user and number of VMs per lab. DevTest Lab also enables you to create policies to automatically shut down and start VMs.

### Quickly get to "Ready to Test"

DevTest Lab enables you to create pre-provisioned environments with everything your team needs to start developing and testing applications. Simply claim the environments where the last good build of your application is installed and get working right away. Or, use containers for even faster and leaner environment creation.

### Create once, use everywhere

Capture and share environment templates and artifacts within your team or organization - all in source control - to create developer and test environments easily.

### Integrates with your existing Toolchain

Leverage pre-made plug-ins or our API to provision Dev/Test environments directly from your preferred continuous integration (CI) tool, integrated development environment (IDE), or automated release pipeline. You can also use our comprehensive command-line tool.

# Partner Needs

Partners want to make profitable businesses by deploying and selling Microsoft Azure.

They also want to sell and deploy Azure for Development and Test Scenarios, but do not have the experience or understanding to say what is possible or know possible ways to sell the value of it or set it up.

At the same time, Partners and Customers have expressed a need to have more capacity that is flexible in their businesses for running VMs for various purposes. They want to spend less time managing hardware and IT and more time selling and deploying.

# Introduction to Starter Kits

This concept is a set of deliverables, packaged as an offering that are named as a starter kit. Starter kits are designed to show a partner a specific scenario in Azure that could be possible for them to build and equip them in the technical sales cycle. Each kit will include:

1. A Description of the partner benefit for using and participating in the kit.
2. An assessment questionnaire and guidance that Partners could use with a customer.
3. An Architecture Topology presentation for a recommended way to implement the specified scenario.
4. A cost estimator (based on retail pricing) for implementing the recommended scenario on Azure.
5. A Statement of Work template for implementing the recommended scenario that a partner could use.
6. Hands on Labs a Partner can self-study to build technical skill implementing the recommended scenario.

# Starter Kits - Partner Benefits

1. Reduce time in creating a proposal for a customer through a sale and deployment template
2. Reduce the learning curve cost by focusing on a proven scenario
3. Help assess and determine the technical requirements for migrating existing Applications to the Cloud
4. Sell, estimate cost and deploy working solutions to your customer.
5. Get tools and templates to use when discussing a Dev/Test deployment with your customers.
6. Receive a recommended set of topology diagrams for implementing a Dev/Test scenario on Azure.
7. Receive guidance for self-study for learning the recommended Dev/Test scenario at a technology level.