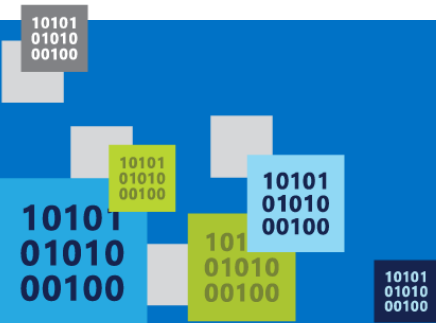


The Big Cloud Theory

Why the cloud?

- Rapidly setup environments to drive business priorities
- Scale to meet peak demands
- Increase daily activities, efficiency and reduced cost.

Why the cloud?



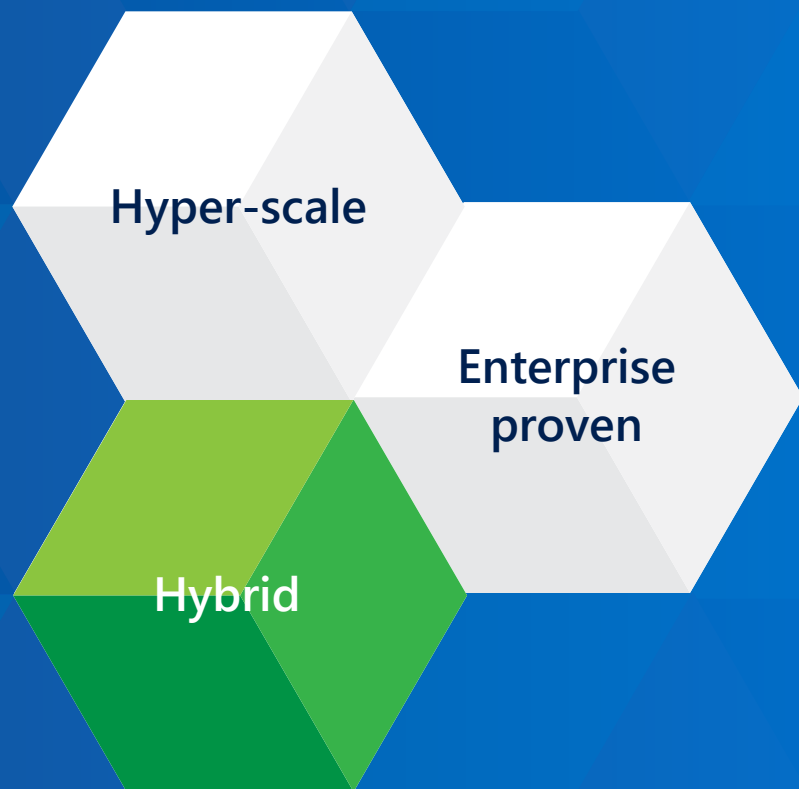


Hyper-scale

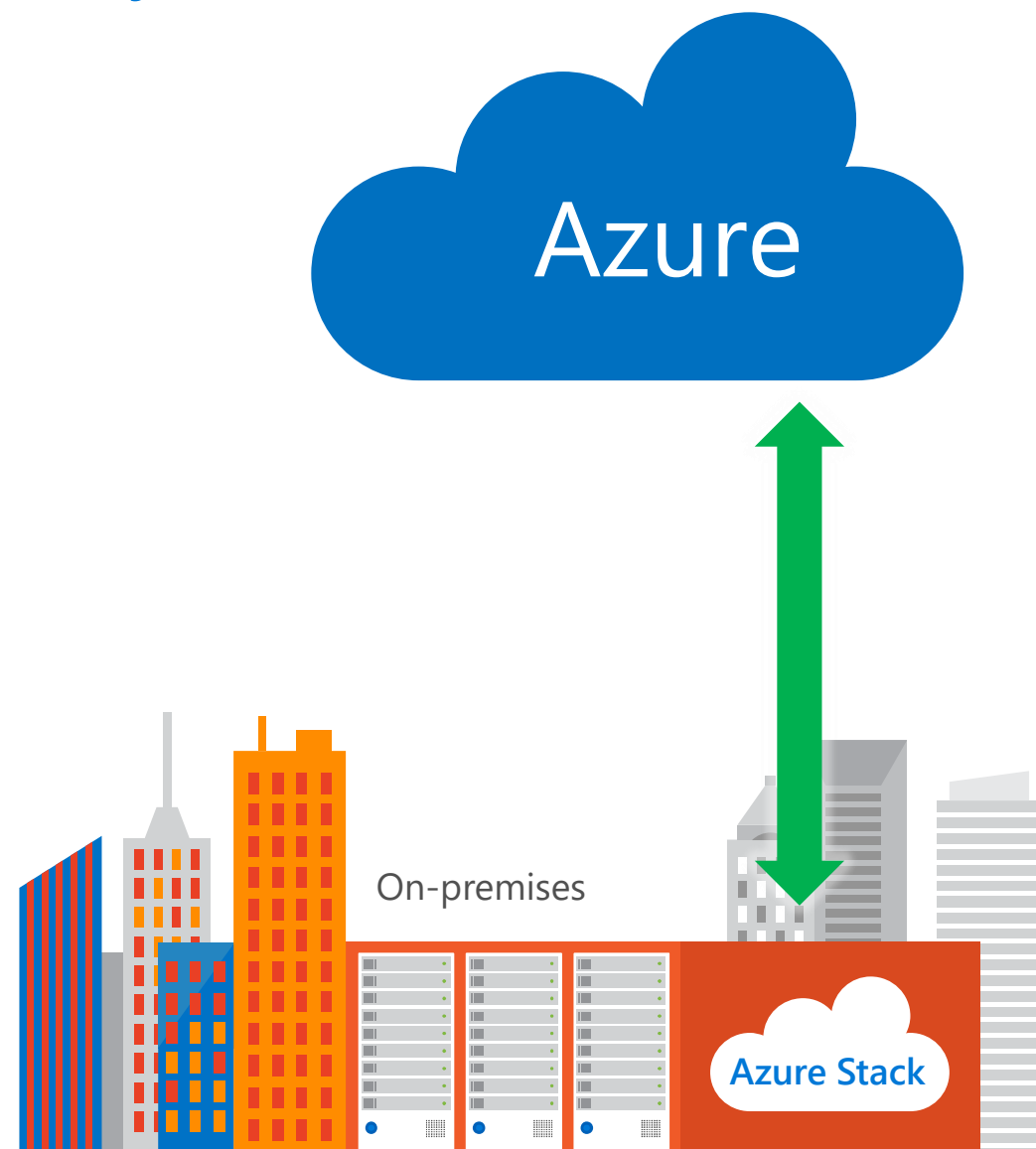
Enterprise
proven

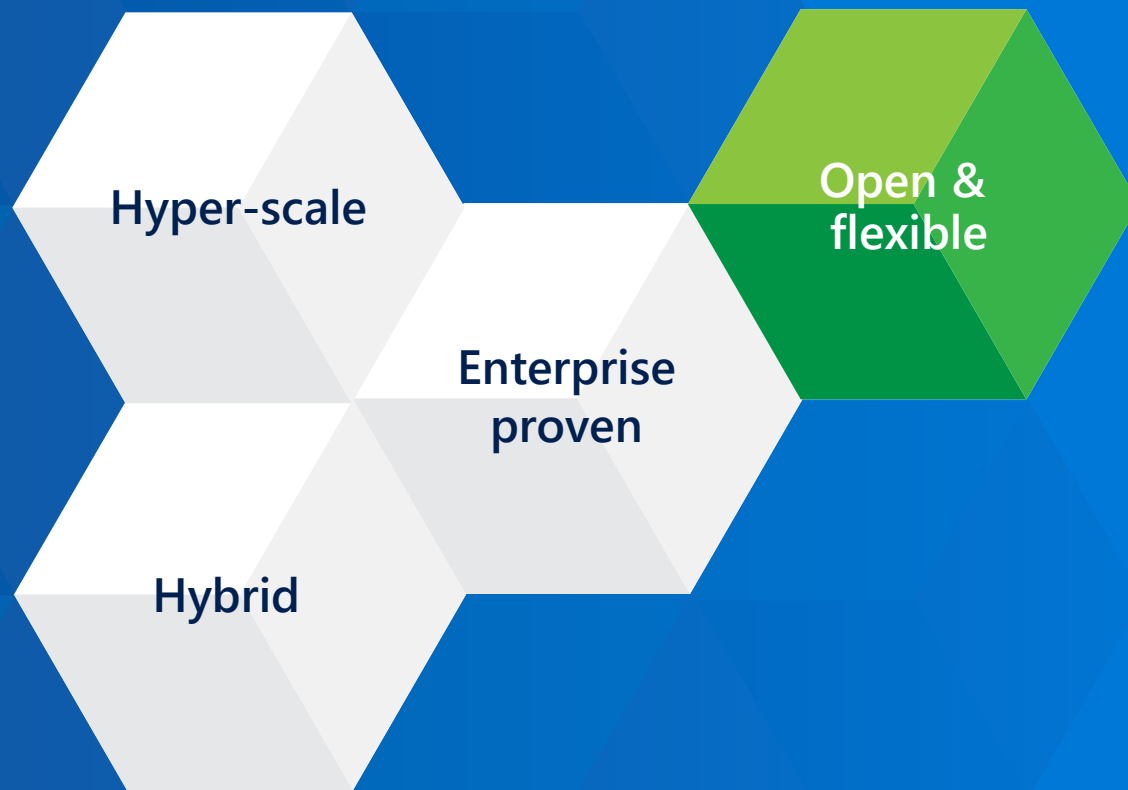
Hybrid

Azure



Hybrid





Open + Flexible

Management



Applications



App Frameworks

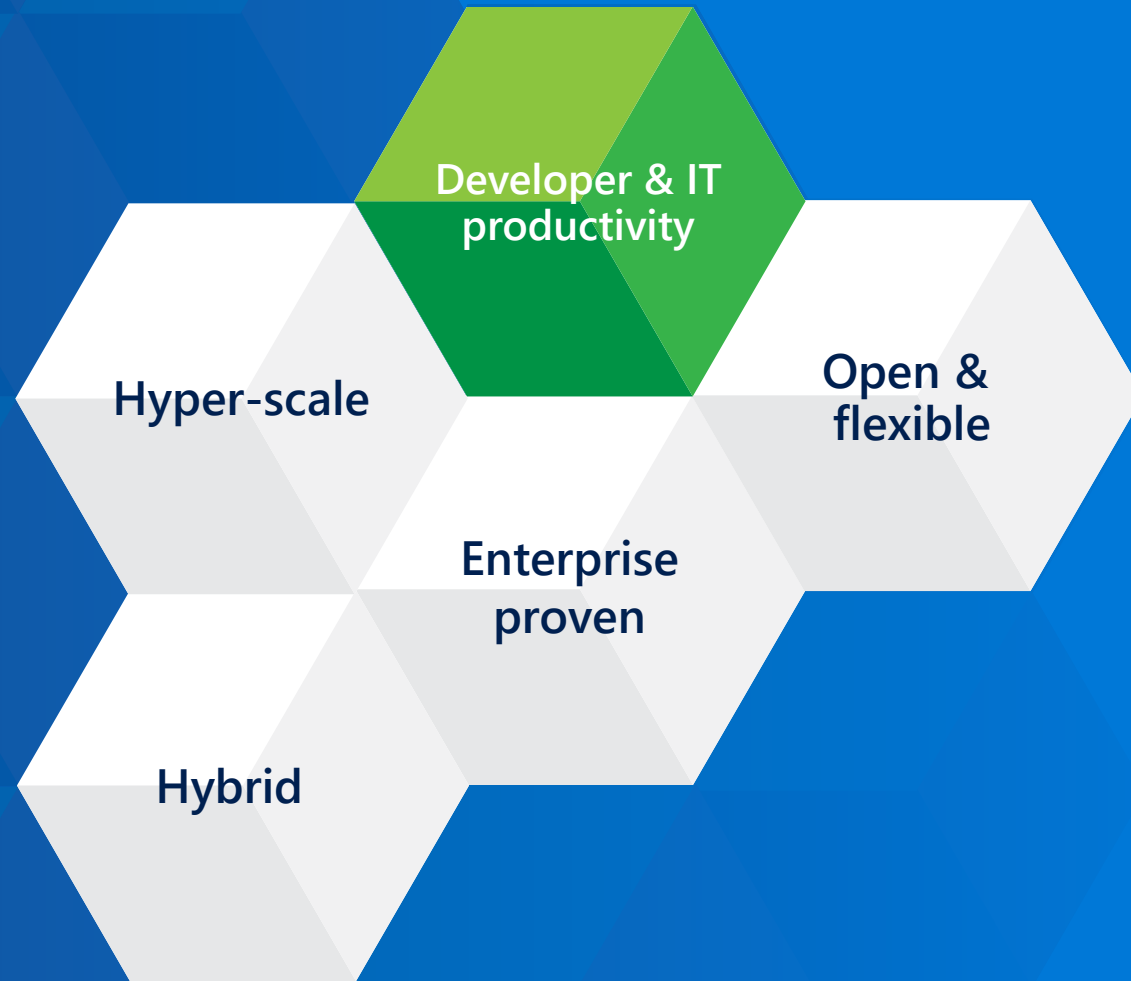


Databases & Middleware



Infrastructure





Dev + IT productivity



Visual Studio



GitHub



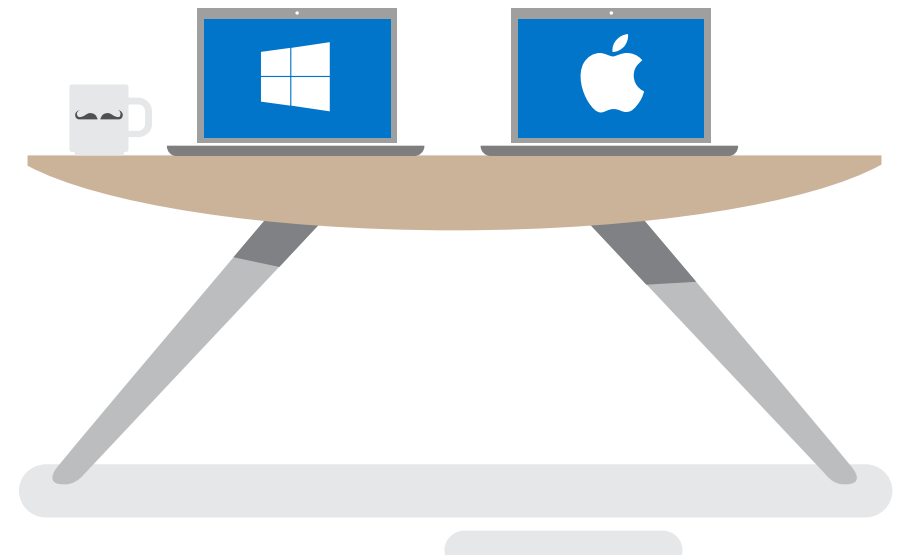
Eclipse

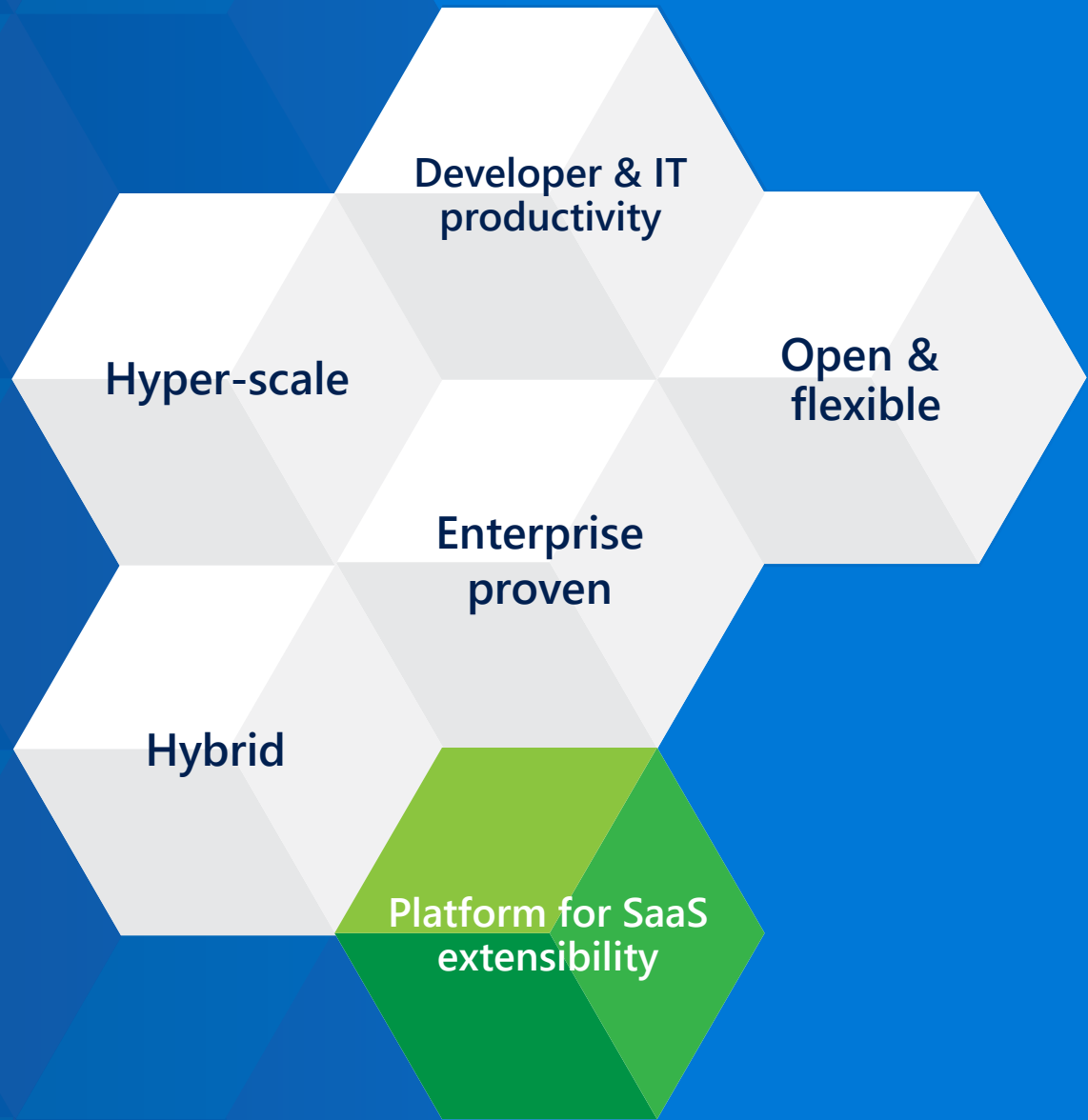


Chef + Puppet

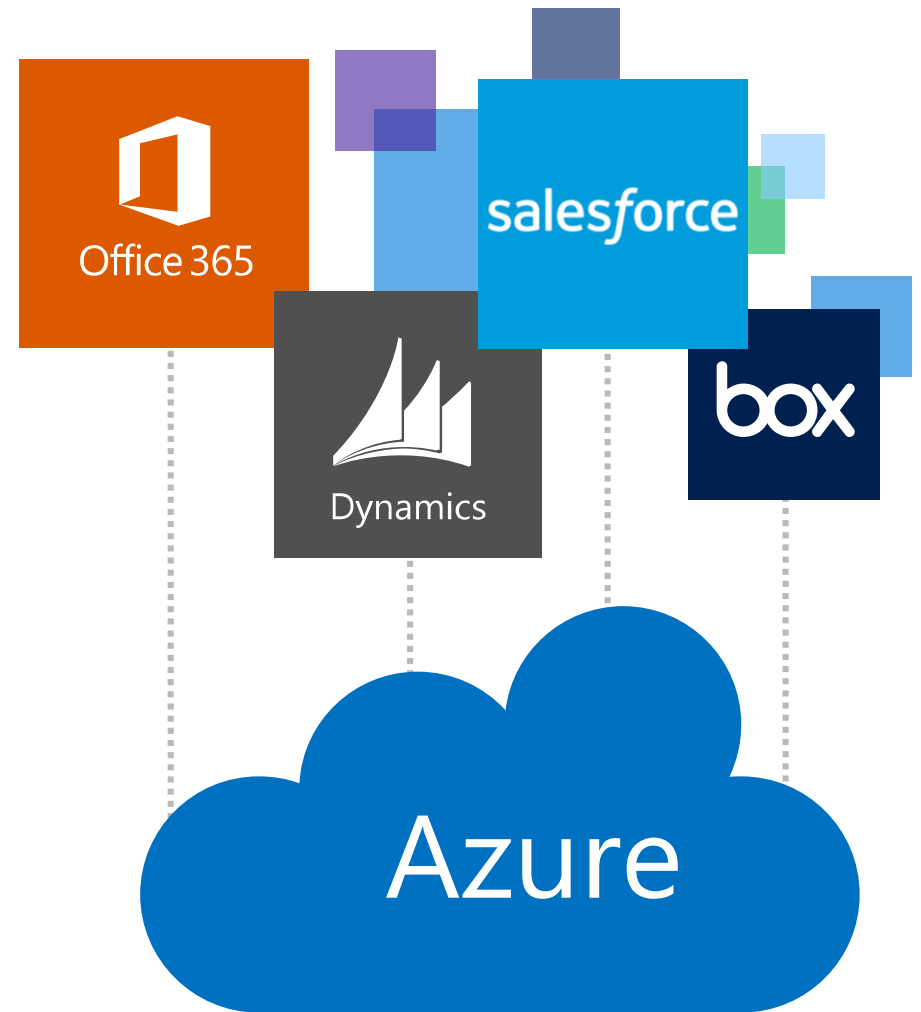


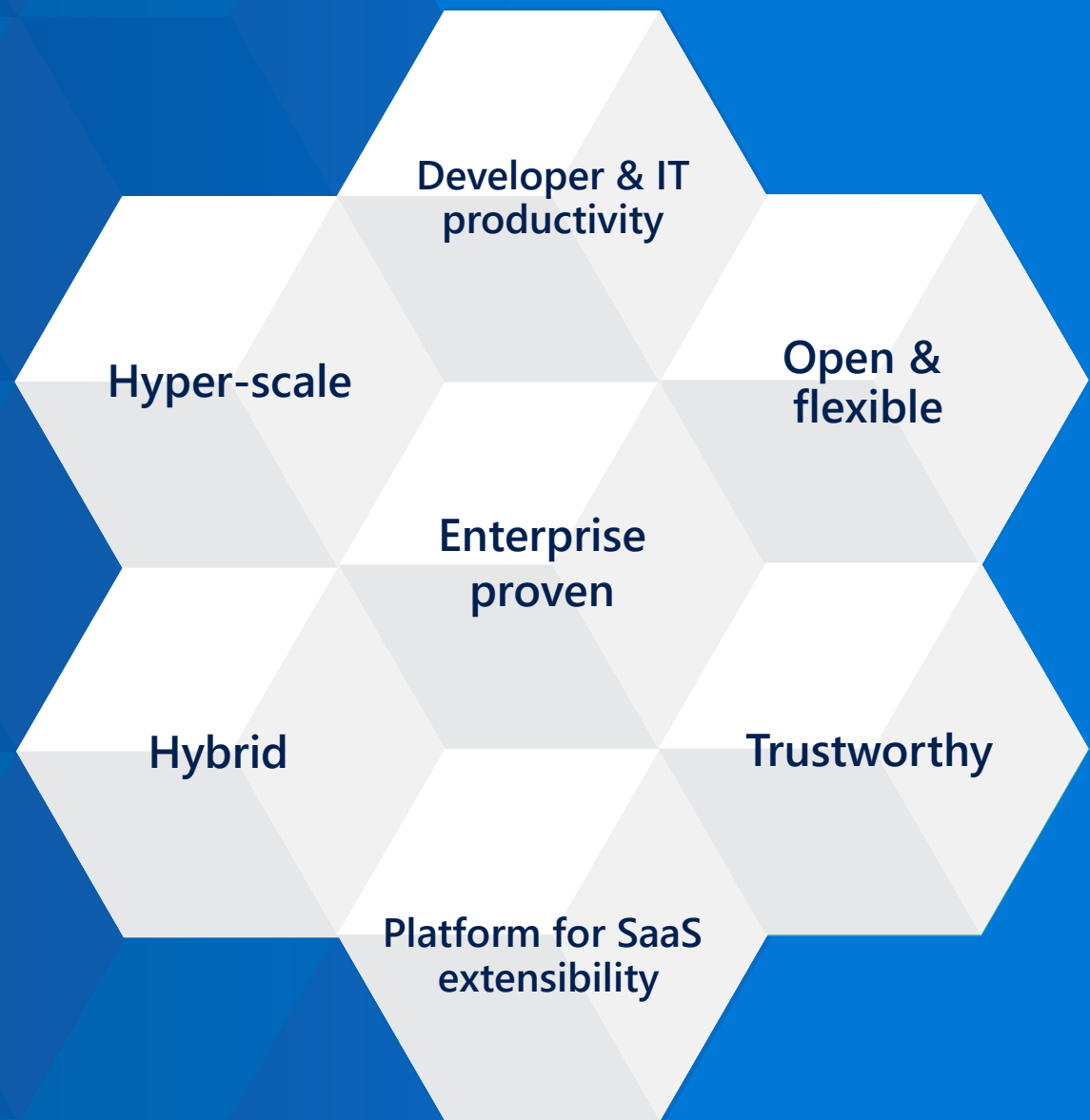
Powershell





SaaS extensibility





Trustworthy

More compliance certifications than any other cloud





30

Azure regions
Hyper-scale
around the world

Latest launch was in September 2016

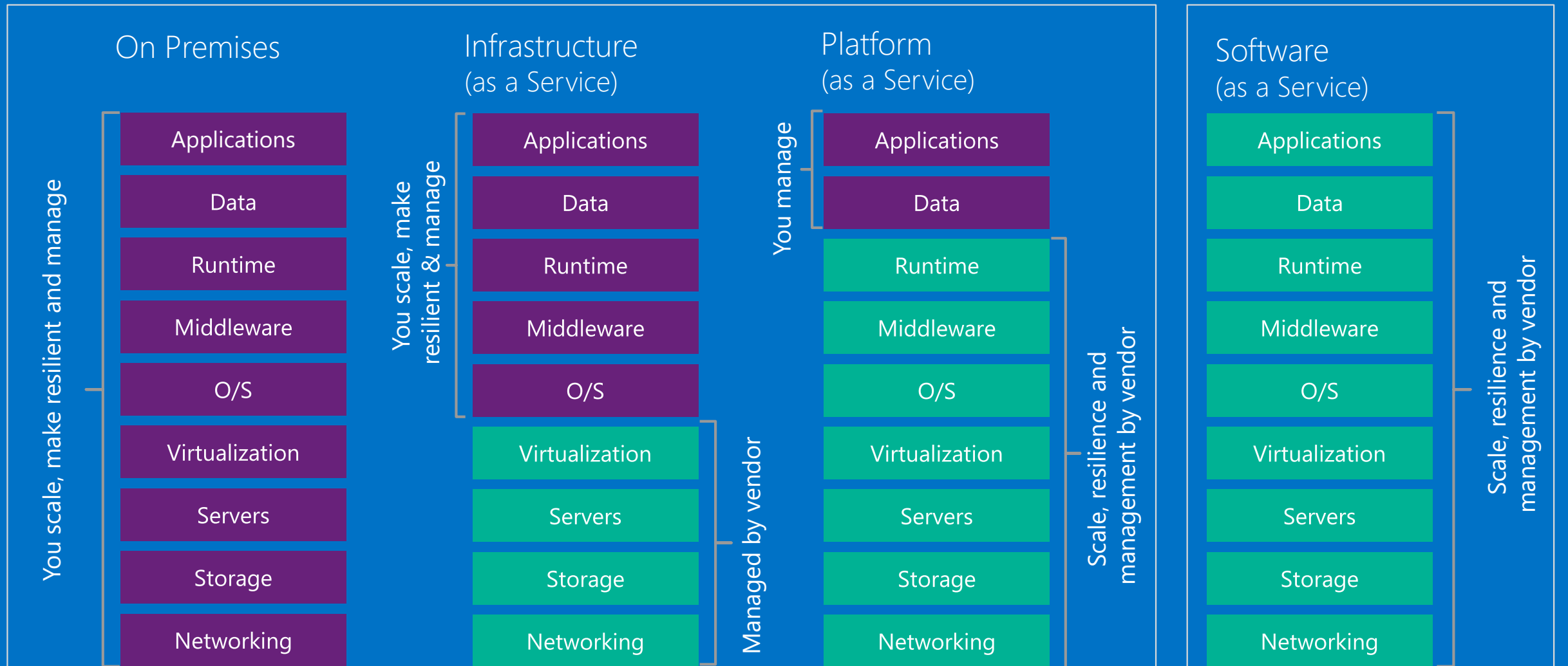
UK West, UK South

GENERALLY AVAILABLE

8 new regions announced: US DoD East, US DoD Central, US Gov Arizona, US Gov Texas, France Central, France South, Korea Central, Korea South

Why the cloud?

Microsoft Azure



Azure App Service

Build and scale great cloud apps



.NET, Java, Node.js, PHP, Python

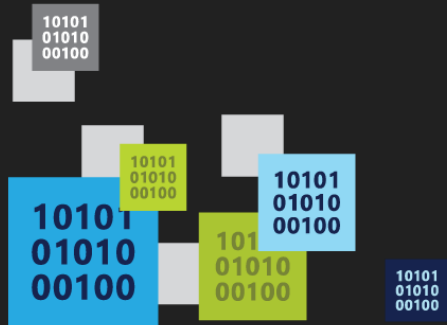
Auto patching

Auto scale

Integration with existing apps

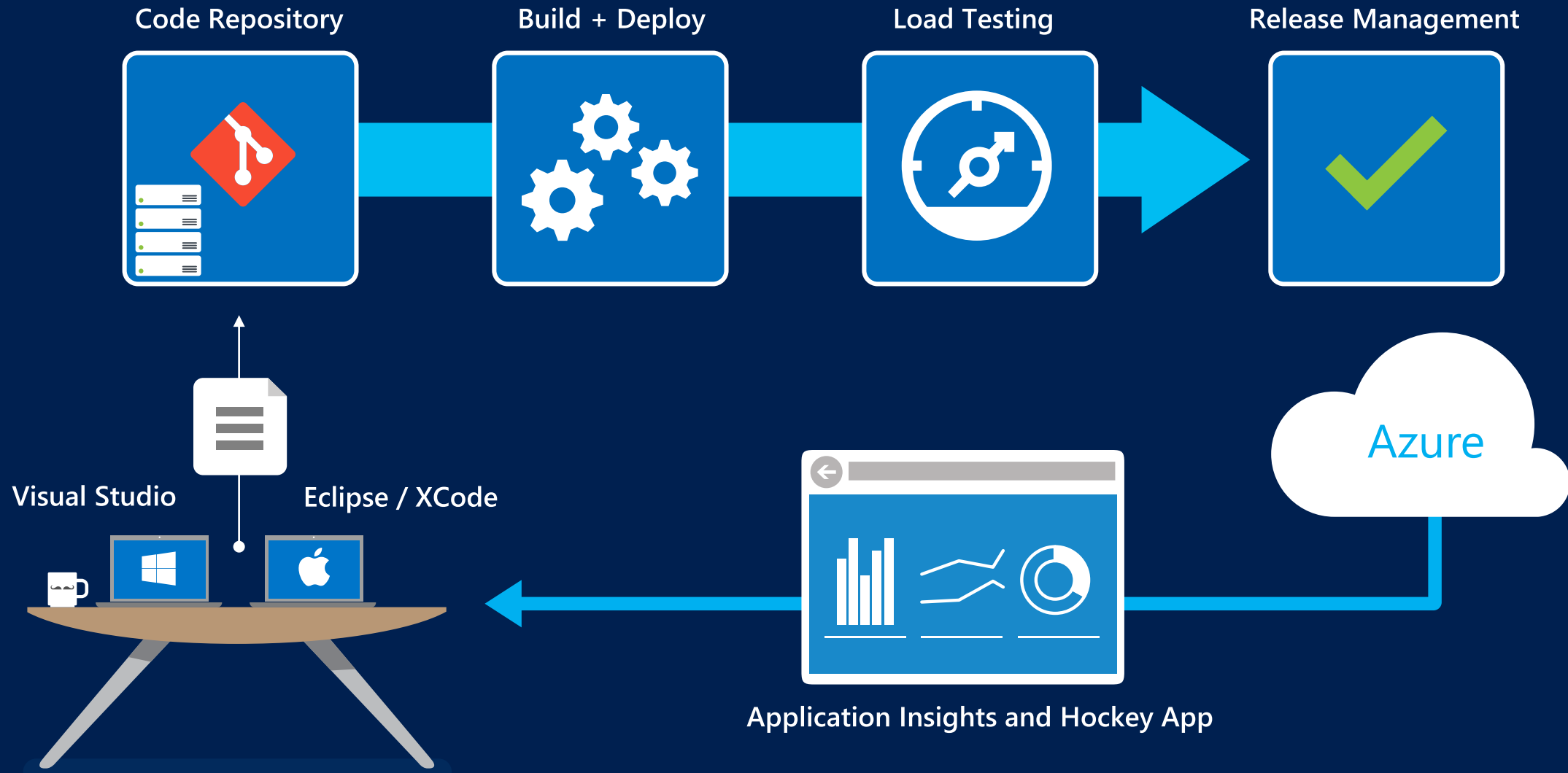
Continuous deployment

Demo: Azure Portal

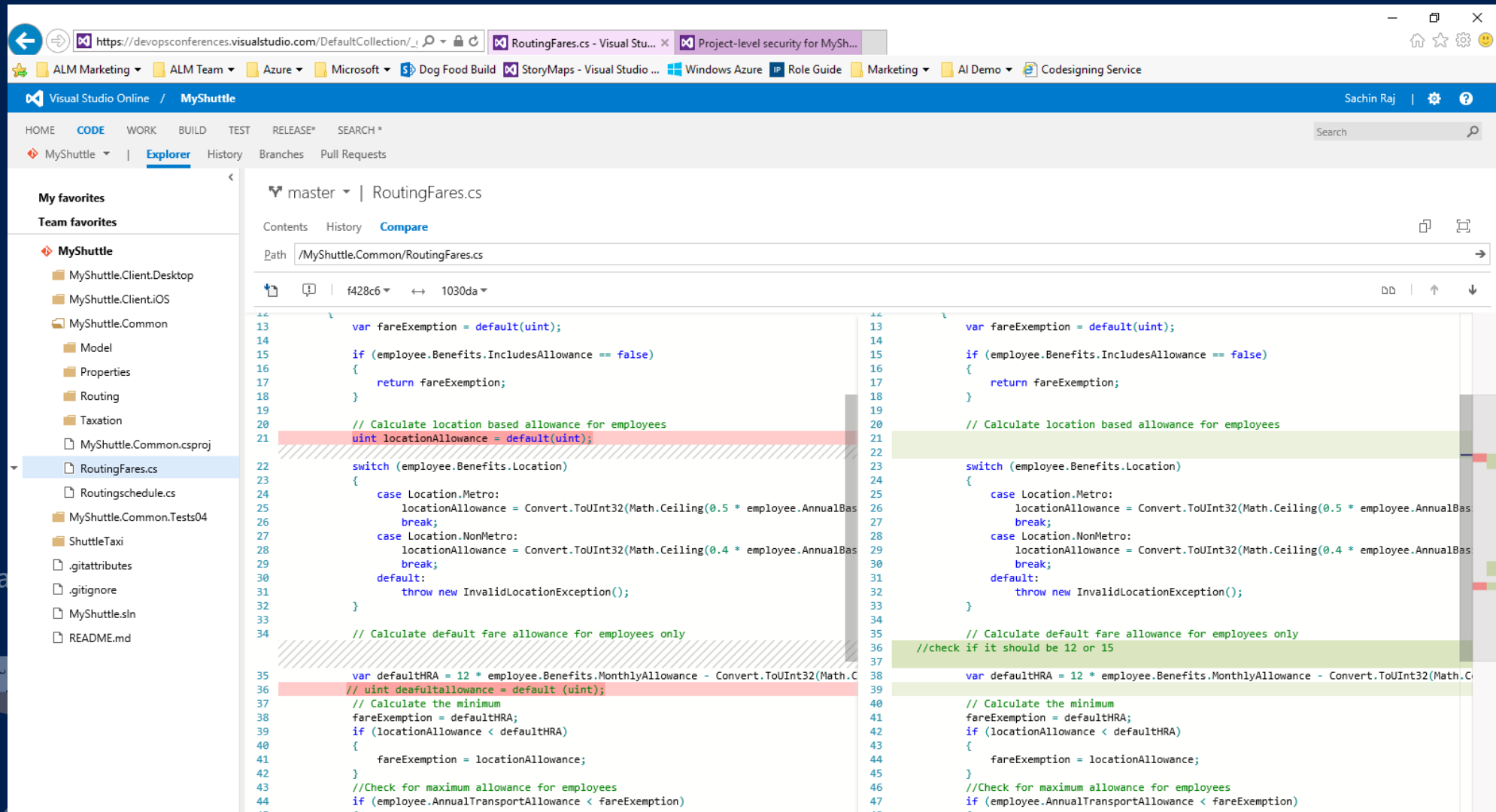


Agility

Visual Studio Team Services: DevOps



VS Team Services: Code Repository



The screenshot displays the Visual Studio Online (VSO) interface. The top navigation bar includes tabs for 'My Shuttle' and 'RoutingFares.cs - Visual Studio...'. The left sidebar shows the 'My Shuttle' project structure with folders like 'My Shuttle.Client.Desktop', 'My Shuttle.Client.iOS', 'My Shuttle.Common', 'Model', 'Properties', 'Routing', 'Taxation', and 'My Shuttle.Common.csproj'. The main editor area shows a comparison of the 'RoutingFares.cs' file between two versions (f428c6 and 1030da). The code is written in C# and includes comments and logic for calculating location-based allowance for employees. The comparison highlights differences between the two versions, with the left version being the baseline and the right version showing changes. The code includes a switch statement for 'employee.Benefits.Location' and a default case for 'InvalidLocationException()'. The comparison shows that the right version has a new case for 'Location.Metro' and a new case for 'Location.NonMetro'.

```
13 var fareExemption = default(uint);
14
15 if (employee.Benefits.IncludesAllowance == false)
16 {
17     return fareExemption;
18 }
19
20 // Calculate location based allowance for employees
21 uint locationAllowance = default(uint);
22
23 switch (employee.Benefits.Location)
24 {
25     case Location.Metro:
26         locationAllowance = Convert.ToInt32(Math.Ceiling(0.5 * employee.AnnualBas
27         break;
28     case Location.NonMetro:
29         locationAllowance = Convert.ToInt32(Math.Ceiling(0.4 * employee.AnnualBas
30         break;
31     default:
32         throw new InvalidLocationException();
33 }
34
35 // Calculate default fare allowance for employees only
36
37 var defaultHRA = 12 * employee.Benefits.MonthlyAllowance - Convert.ToInt32(Math.C
38 // uint defaultallowance = default(uint);
39 // Calculate the minimum
40 fareExemption = defaultHRA;
41 if (locationAllowance < defaultHRA)
42 {
43     fareExemption = locationAllowance;
44 }
45 //Check for maximum allowance for employees
46 if (employee.AnnualTransportAllowance < fareExemption)
47 {
48     fareExemption = employee.AnnualTransportAllowance;
49 }
```

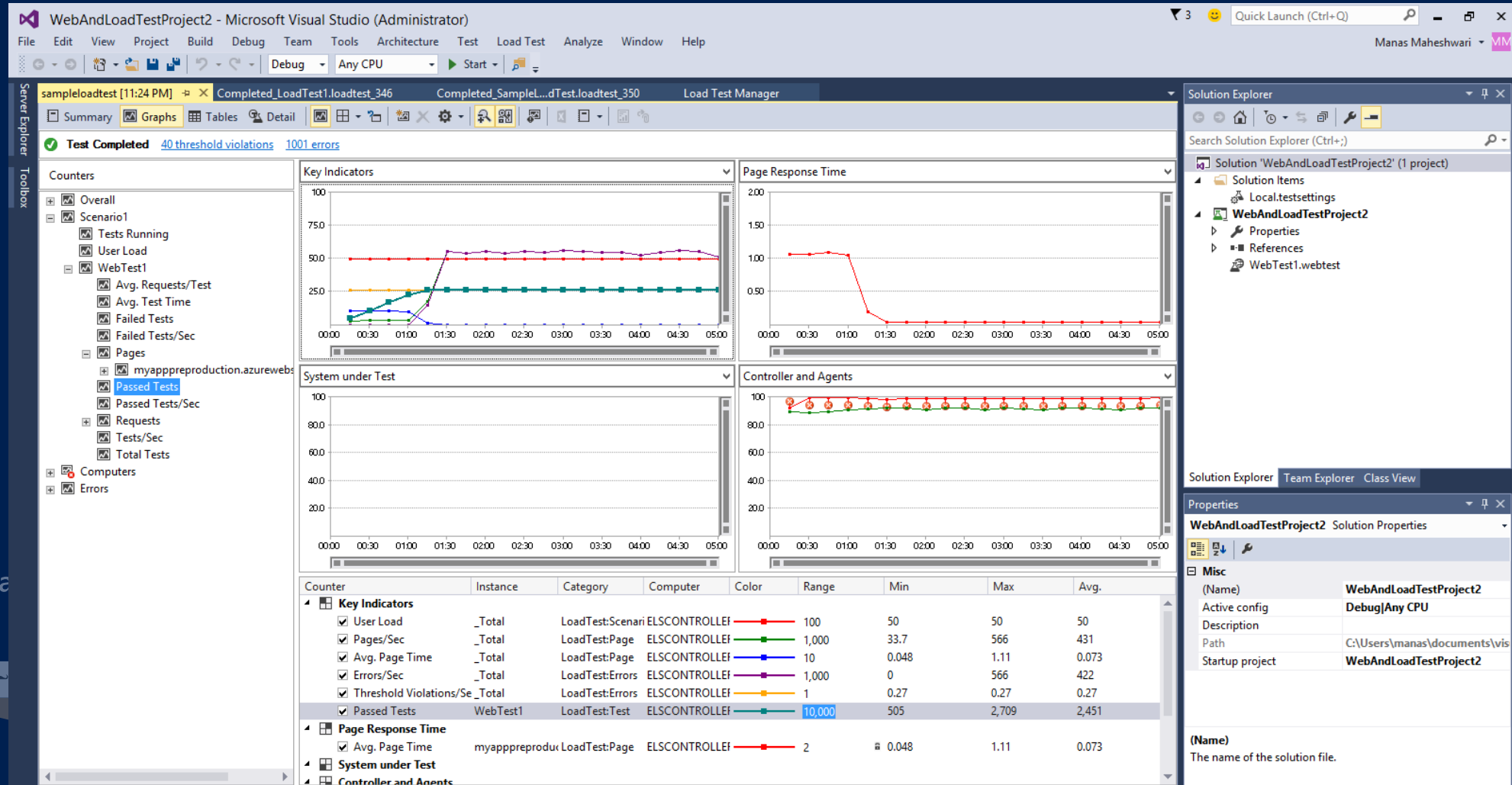

VS Team Services: Build + Deploy

The screenshot displays the Visual Studio Online (VS Team Services) interface for a build definition named 'MyShuttle'. The browser address bar shows the URL: <https://devopsconferences.visualstudio.com/DefaultCollection/MyShuttle>. The user 'Sachin Raj' is logged in.

The interface is divided into several sections:

- Left Sidebar (Explorer):** Contains a search bar and a list of build definitions. Under 'Team favorites', 'MyShuttle Java' is listed with a status of 'Completed 3...'. Under 'Build definitions', 'MyShuttle-Android' is selected.
- Definitions / MyShuttle-Android | Builds:** The main area shows the build definition configuration. It includes tabs for 'Build', 'Options', 'Repository', 'Variables', 'Triggers', 'General', 'Retention', and 'History'. The 'Build' tab is active, showing a list of build steps: 'Xamarin License', 'Android Build', 'Android Signing' (selected), 'NuGet Installer', 'MSBuild', 'Xamarin License', 'Xamarin Test Cloud', and 'Publish Build Artifacts'.
- Right Panel (Configuration):** Displays the configuration for the selected 'Android Signing' step, titled 'Signing and aligning APK file(s) \MyShuttle.Android\Myshuttle.apk'. It includes fields for 'APK Files' (set to '\MyShuttle.Android\Myshuttle.apk') and three sections of options:
 - Jarsigner Options:** Includes checkboxes for 'Sign the APK' (checked), 'Keystore File' (set to '.\keystore'), 'Keystore Password', 'Alias', 'Key Password', and 'Jarsigner Arguments' (set to '-verbose -sigalg MD5withRSA -digestalg SHA1').
 - Zipalign Options:** Includes checkboxes for 'Zipalign' (checked) and 'Zipalign Location'.
 - Control Options:** Includes checkboxes for 'Enabled' (checked), 'Continue on error', and 'Always run'.

VS Team Services: Load Testing



Visual

Application Insights and Hockey App

VS Team Services: Release Management

The screenshot displays the Visual Studio Online Release Management interface. The top navigation bar includes links for HOME, CODE, WORK, BUILD, TEST, and RELEASE*. The left sidebar shows the 'Release Definitions' section with a list of definitions: MyShuttle CRM, MyShuttle IT, MyShuttle Plat, MyShuttle Web, MyShuttle-Temp, PartsUnlimited, and Tracking Service. The main area, titled 'All release definitions', shows a grid of release definition cards. Each card displays the environment (Dev, QA, Prod), the release name, the version, the date, and a bar chart representing the release history. The cards are organized by release definition, with each definition having multiple environment cards. For example, 'MyShuttle CRM' has Dev (CRM 2.0), QA (CRM 1.7), and Prod (CRM 1.5) environments. The 'MyShuttle Plat' definition has a Dev environment (Release.20150915155247) and a QA environment. The 'MyShuttle Web' definition has Dev (Release.20150529002320), QA (Release.20150529002320), and Prod (Release.20150527153631) environments. The 'MyShuttle-Temp' definition has a Default Environment (Release.20150714142139).

Visual Studio Online / MyShuttle

HOME CODE WORK BUILD TEST RELEASE* SEARCH *

Search work items

This is the preview version of Release Management service. Refer to [documentation](#) for more information.

Search release definitions...

Release Definitions

- All release definitions
- MyShuttle CRM
- MyShuttle IT
- MyShuttle Plat
- MyShuttle Web
- MyShuttle-Temp
- PartsUnlimited
- Tracking Service

All release definitions

Releases Overview

MyShuttle CRM

- Artifacts (1)
- Vijay Machiraju

MyShuttle IT

- Artifacts (0)
- Vijay Machiraju

MyShuttle Plat

- Artifacts (0)
- Sachin Raj

MyShuttle Web

- Artifacts (1)
- Vijay Machiraju

MyShuttle-Temp

- Artifacts (0)

Dev CRM 2.0 4/27/2015

QA CRM 1.7 4/27/2015

Prod CRM 1.5 4/27/2015

QA ITWeb 3.4 4/25/2015

Prod ITWeb 3.2 4/25/2015

Dev Release.20150915155247 9/15/2015

QA

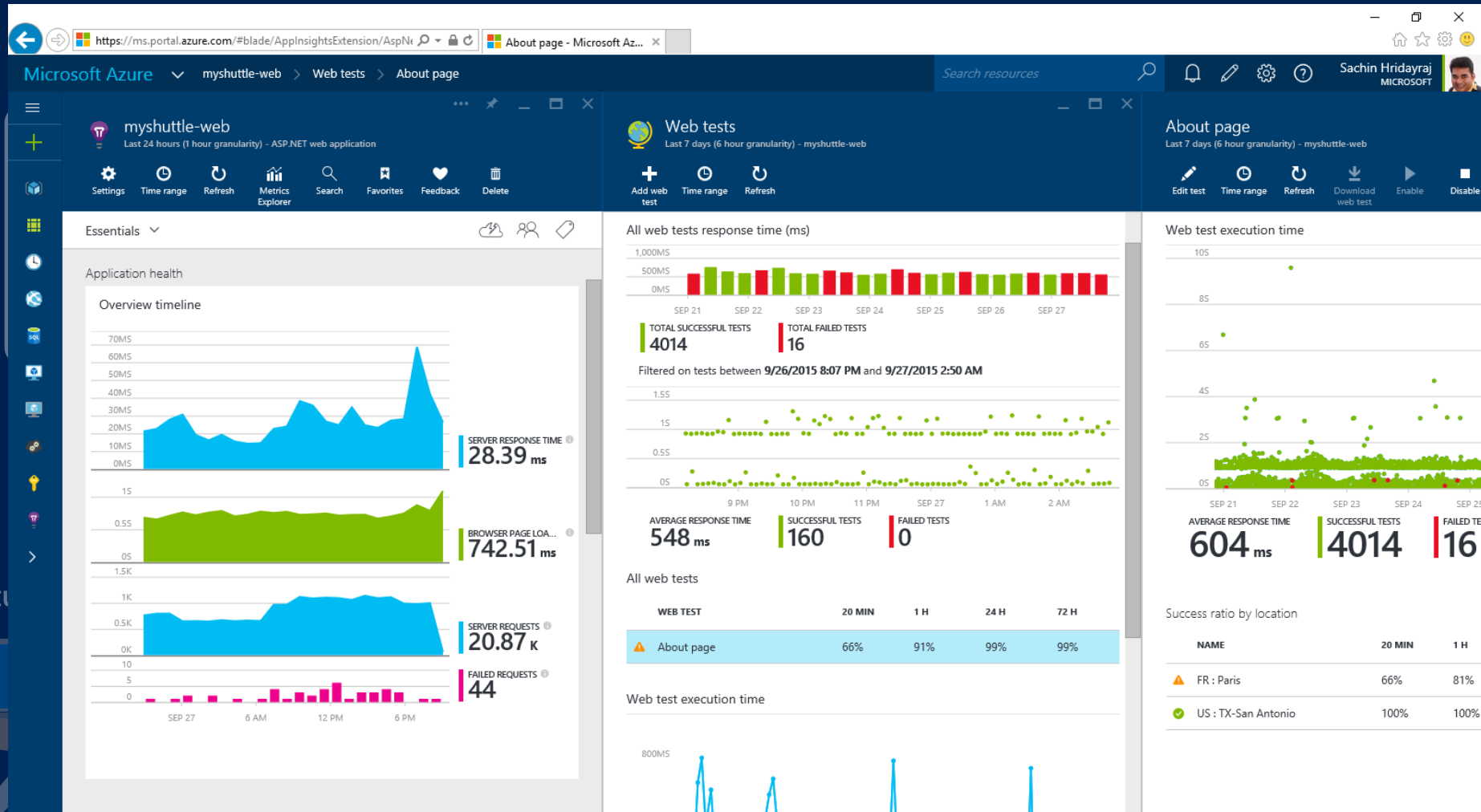
Dev Release.20150529002320 5/29/2015

QA Release.20150529002320 5/29/2015

Prod Release.20150527153631 5/27/2015

Default Environment Release.20150714142139

VS Team Services: Application Insights



Visual Studio

Application Insights and Hockey App

The Cloud is changing the world



¡Gracias!

