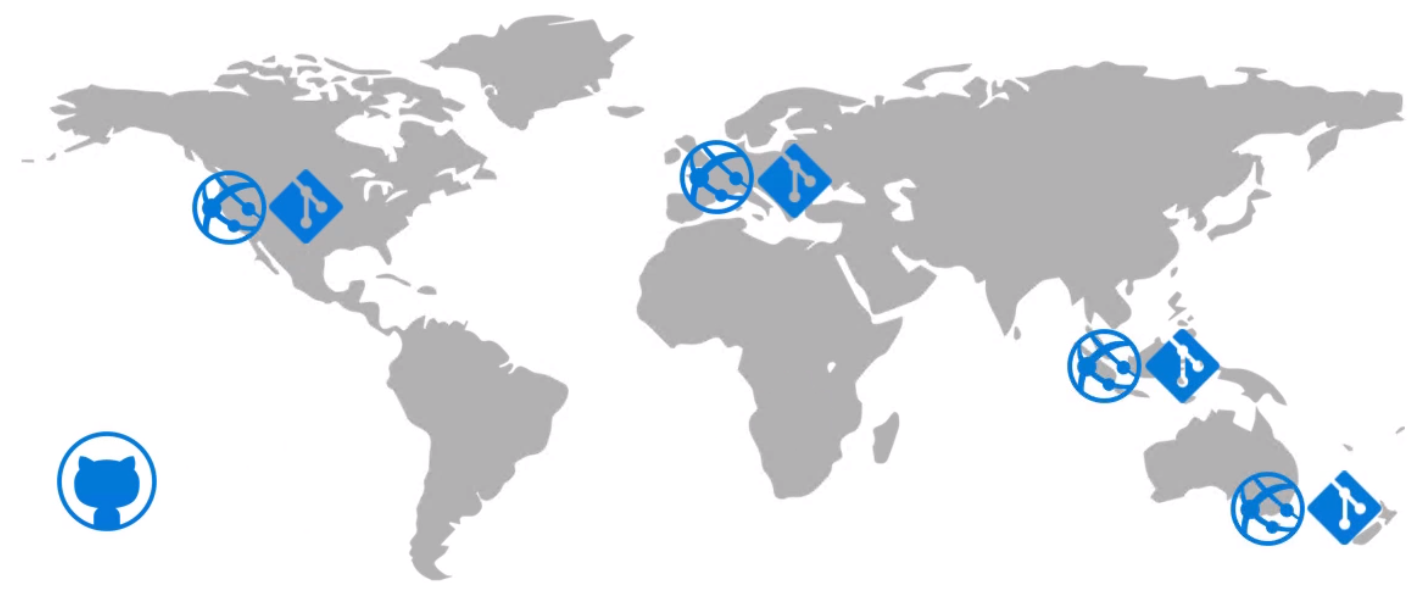
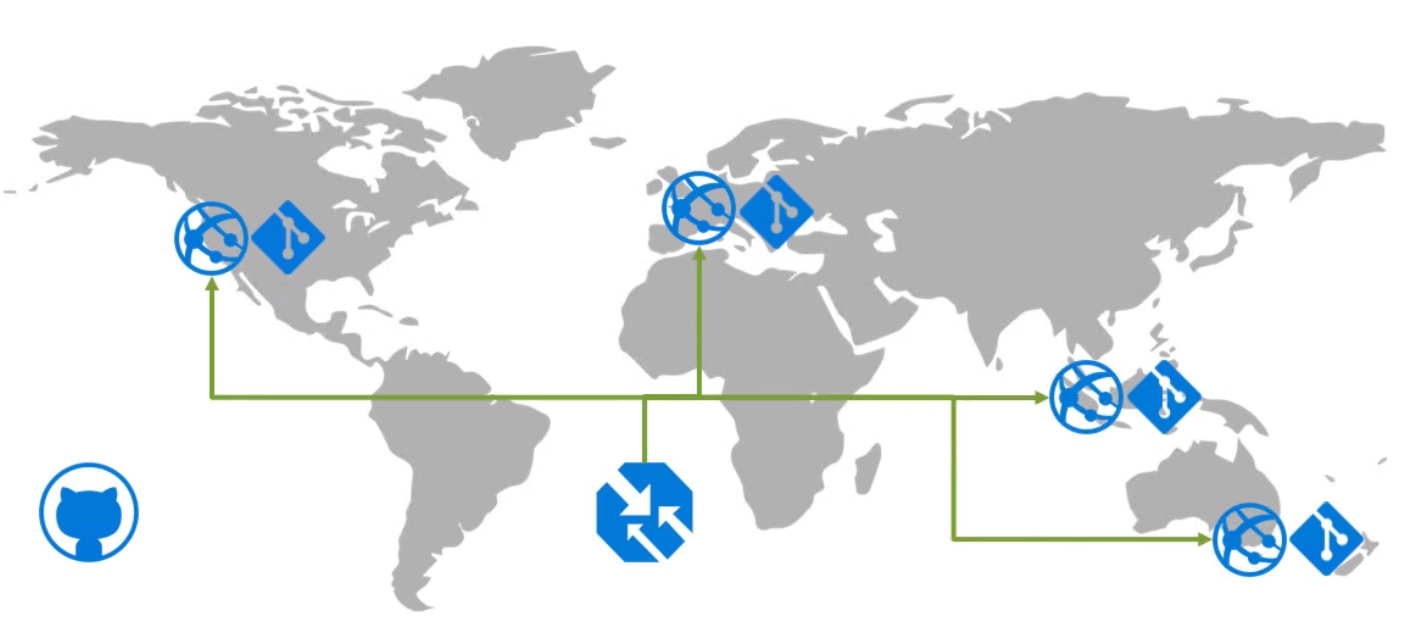


Choosing Different locations for deploying paas solutions



We will deploy application from github

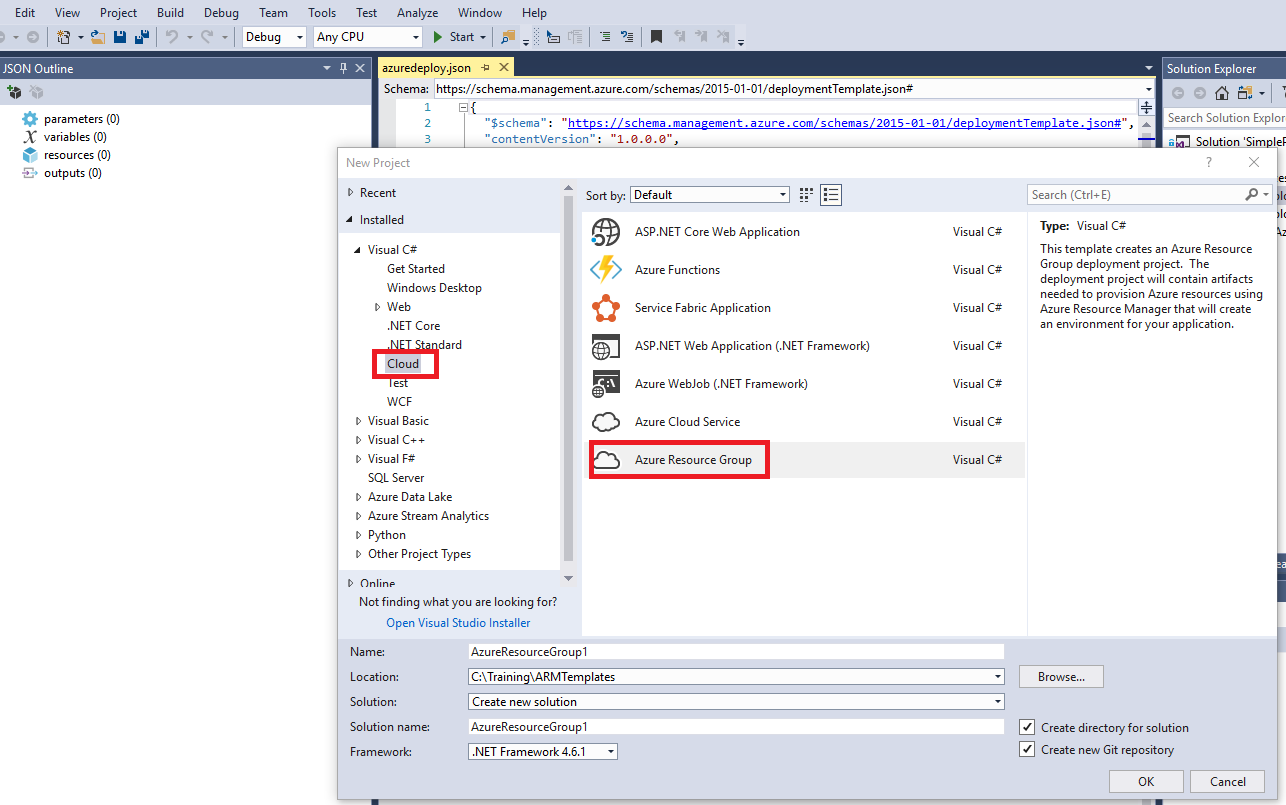


We will be managing the application using traffic manager



Steps:

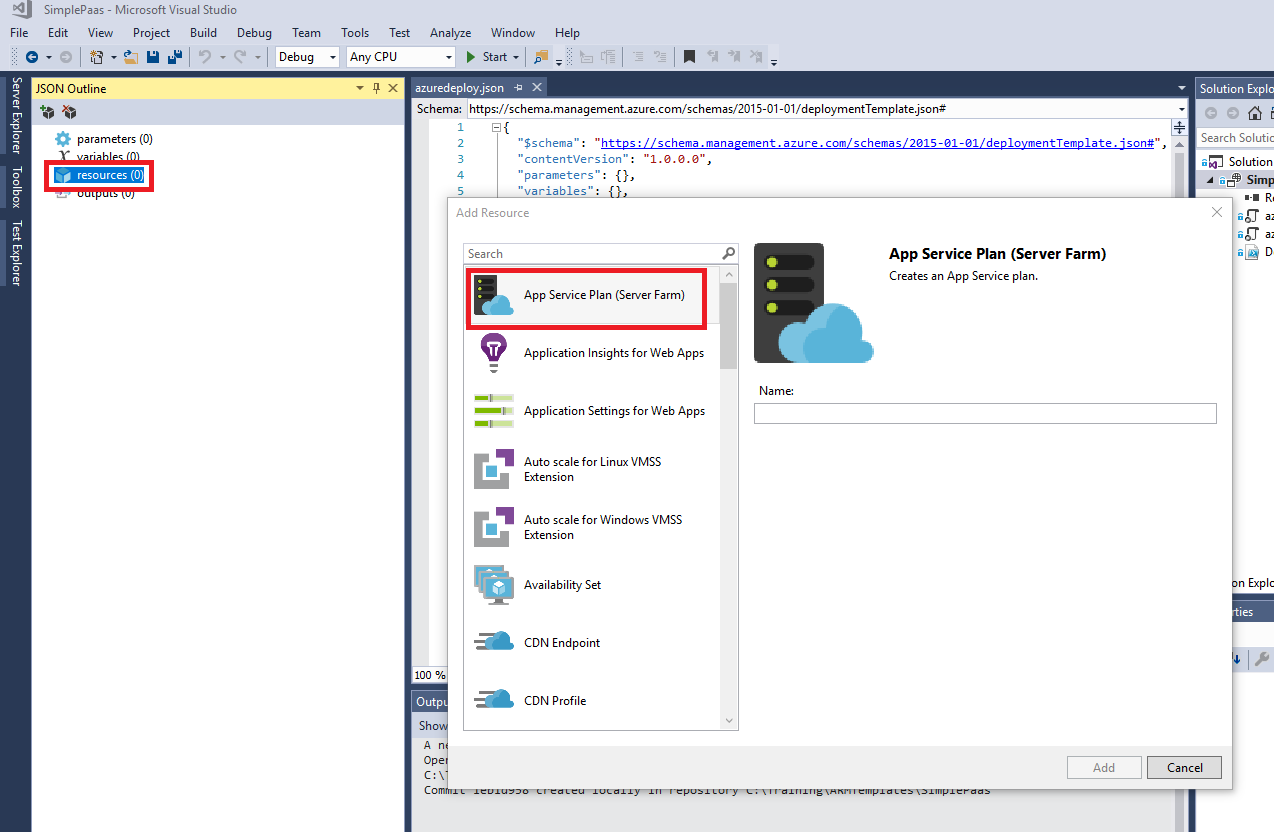
1. Create a new project from visual studio community edition. Ensure you have selected as shown in the below image



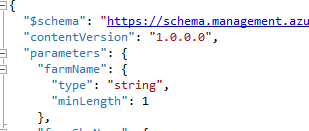
1. Select the blank template
2. Create the resource group by using azure cli as shown here

**az group create --name SamplePaas --location "Central US"**

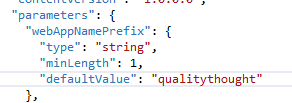
1. Create azure web farm template resource by right click on resources section & selecting App service plan



1. Now give any name & select add. In this demonstration i’m using farm as a Name
2. In the generated json in azure deploy we will be changing the farmName to webAppNamePrefix and add the following changes



To



1. We will be adding new parameter to handle count of web apps

"webAppCount": {

"type": "int",

"defaultValue": 4,

"minValue": 1

}

1. After correcting the parameter name in resources section adding copy section to handle multiple resource creation
2. Changing the resource name parameter to reflect the prefix-<count> using expression "name": **"[concat(parameters('webAppNamePrefix'),'-',copyIndex(),'-', uniqueString(resourceGroup().id))]"**,
3. We will be using the same name in display name as well as properties section
4. Add a parameter for web app locations

"webAppLocations": {

"type": "array",

"defaultValue": ["Central US", "Australia Southeast", "Central India", "West Europe"]

}

1. Modify the location property of webapp resource to take locations

"location": "[parameters('webAppLocations')[copyIndex()]]",

1. Modify the count to take length of webAppLocations to make it much more flexible

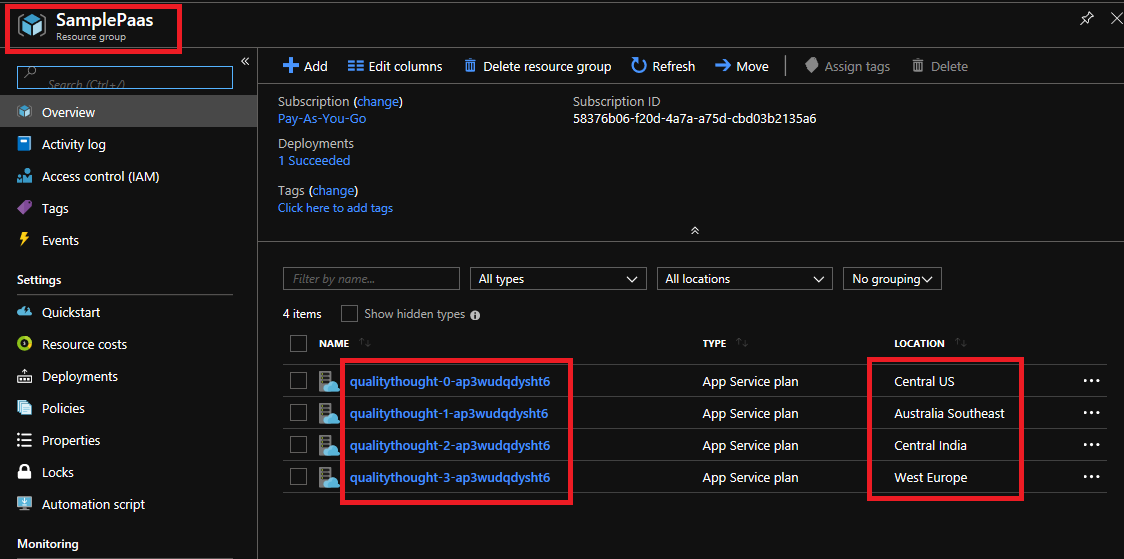
"count": "[length(parameters(webAppLocations))]",

1. Now you can remove WebAppCount Parameter
2. Deploy the template using the following command

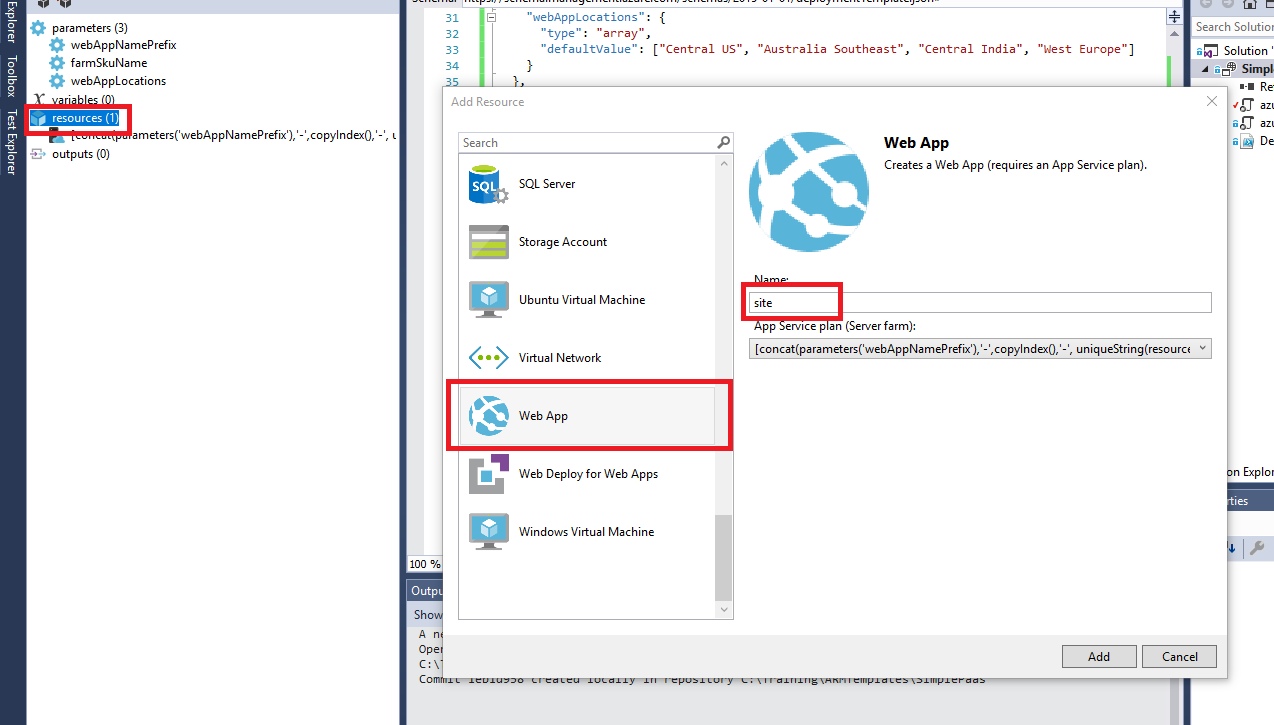
**az group deployment create --name PaasDeployment --resource-group Sam**

**plePaas --template-file .\azuredeploy.json**

1. You can cross check in azure portal . Navigate to Resource Groups ⇒ SamplePaas



1. Now as the farms are ready , we will be creating Web Site Template resources , Deploy and Validate the deployment
2. To add the Web app, Navigate to visual studio right click on resources section & add Web app as shown below



1. Remove the variable sitename created and ensure to add the resource name of web app to the following

"name": "[concat(parameters('webAppNamePrefix'),'-site-',copyIndex(),'-', uniqueString(resourceGroup().id))]",

1. Also use the same section in properties section of web site as well
2. Change the depends on section to farmcopy

"dependsOn": [

"farmCopy"

]

1. Add copy section to web site resource

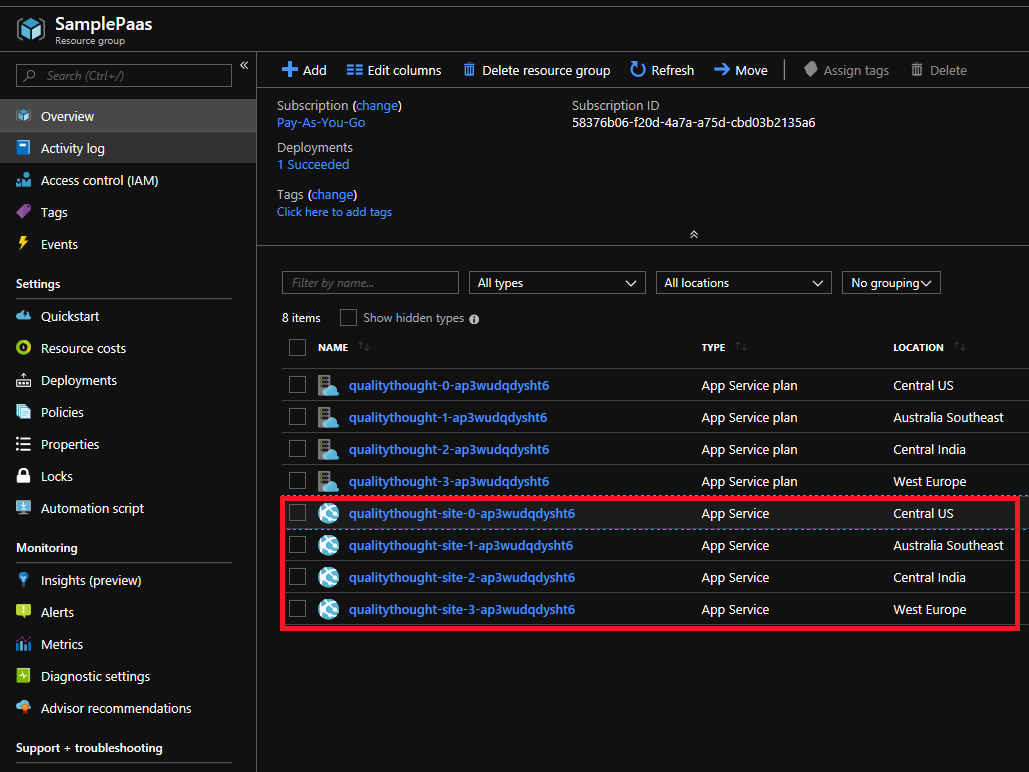
"copy": {

"count": "[length(parameters('webAppLocations'))]",

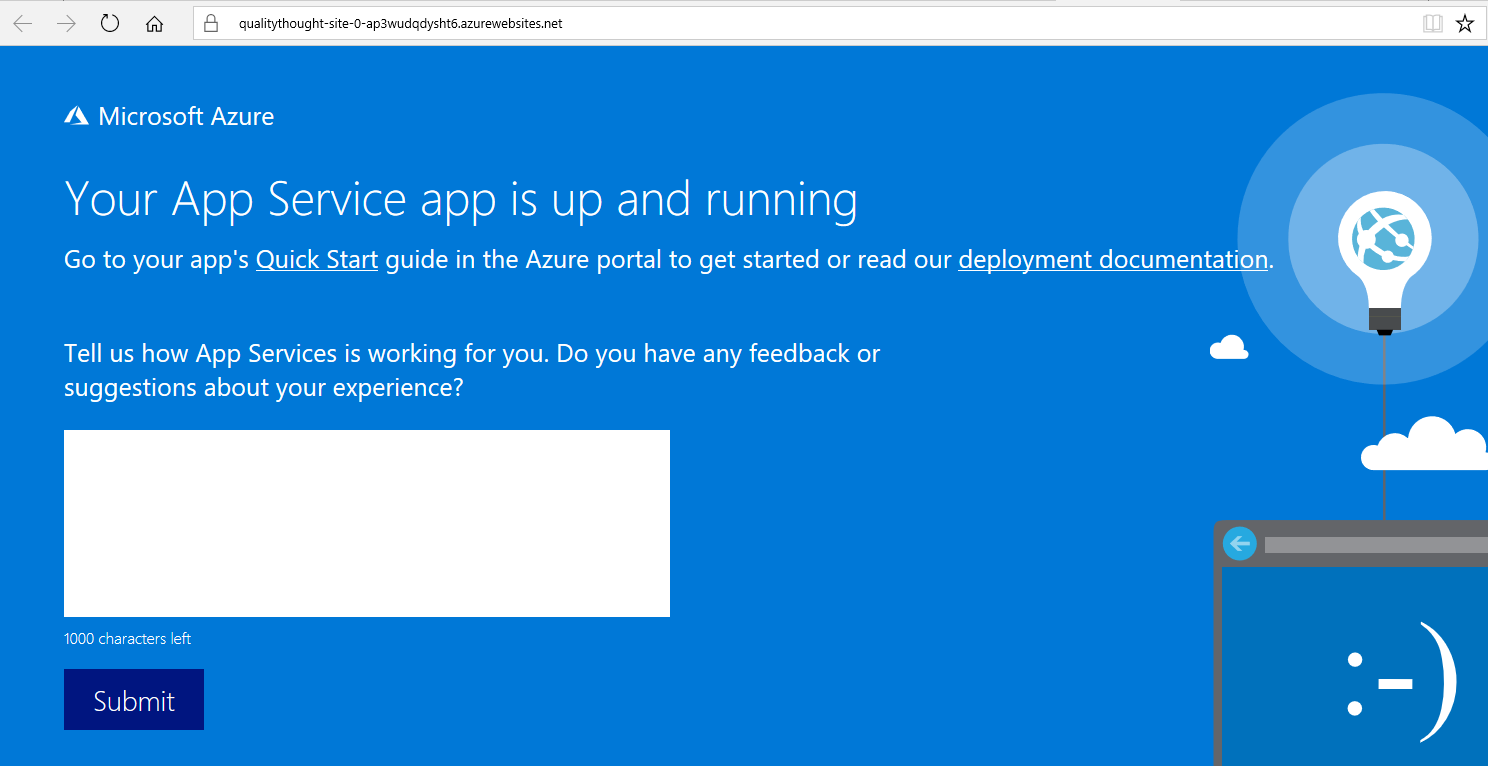
"name": "siteCopy"

}

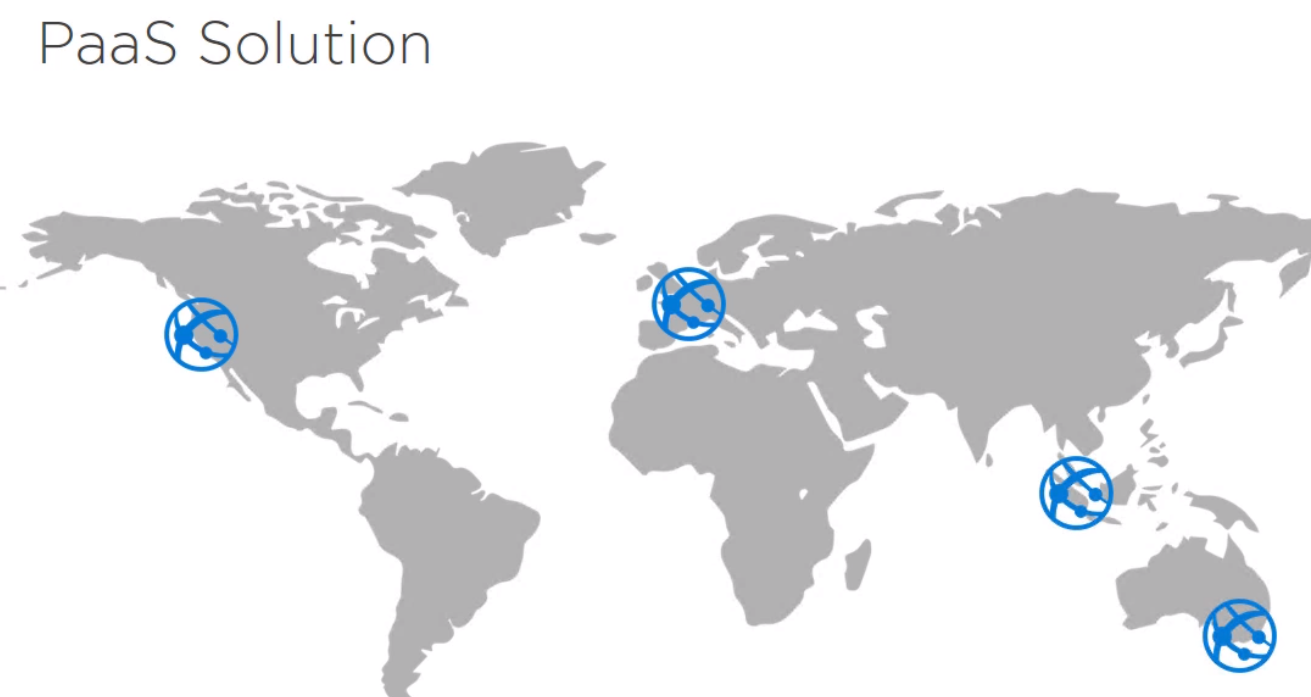
1. Re run the deployment as shown in step **#15**
2. Check in the portal



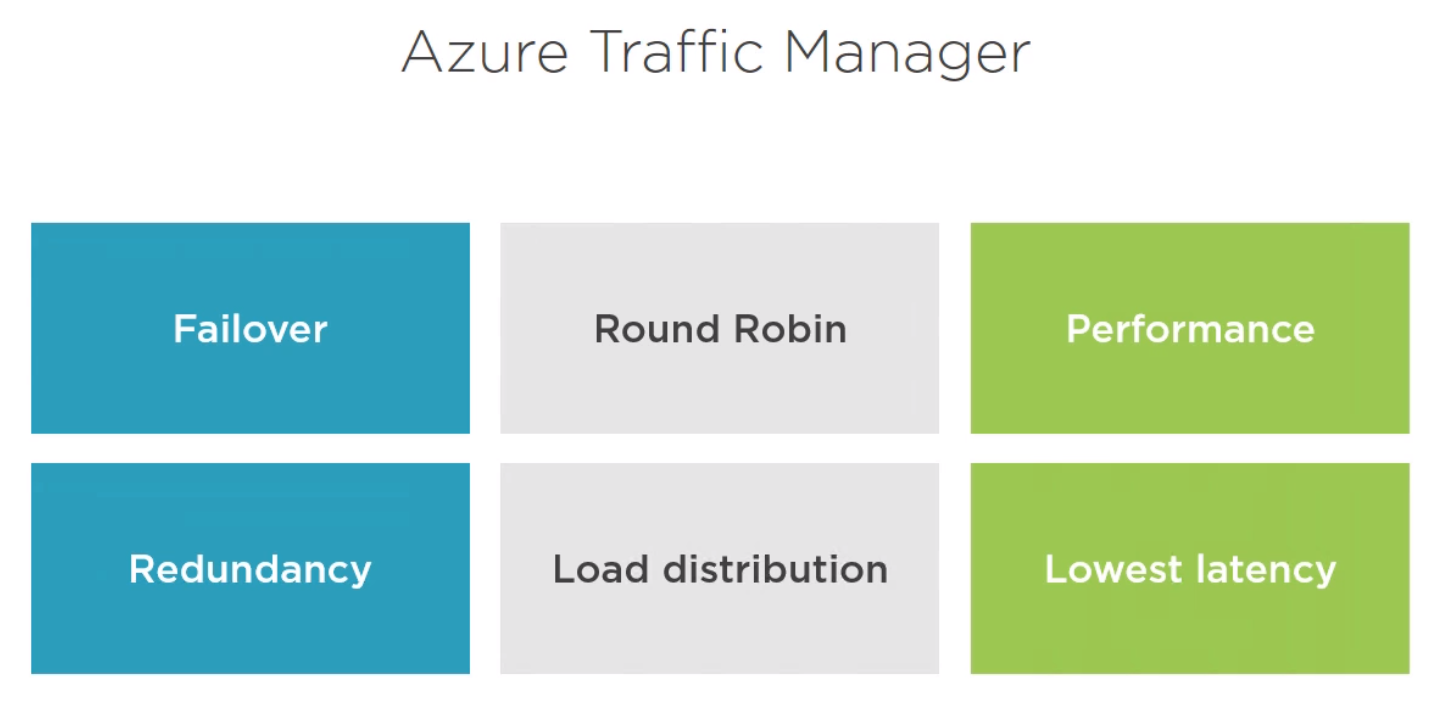
1. Click on any app service and click the url



1. As of now we have app services distributed across 4 regions/locations.



1. In order to provide the unified interface we will be deploying the traffic manager
2. Azure Traffic manager can be used with various options. We would be using traffic manager for lowest latency & Performance



1. We will do the following

Create Traffic Manager template resource & Deploy & Validate

1. Since the template is not available in resources we need to add the resource for traffic manager manually as specified <https://docs.microsoft.com/en-us/azure/templates/Microsoft.Network/trafficmanagerprofiles?toc=%2Fen-us%2Fazure%2Fazure-resource-manager%2Ftoc.json&bc=%2Fen-us%2Fazure%2Fbread%2Ftoc.json>
2. Try to create the resource by typing the following

{

"name": "[concat(parameters('webAppNamePrefix'),'-tm')]",

"type": "Microsoft.Network/trafficManagerProfiles",

"apiVersion": "2015-11-01",

"location": "global",

"dependsOn": [

"siteCopy"

],

"properties": {

"profileStatus": "Enabled",

"trafficRoutingMethod": "Performance",

"dnsConfig": {

"relativeName": "[concat(parameters('webAppNamePrefix'),'-tm')]",

"ttl": "30"

},

"monitorConfig": {

"protocol": "HTTP",

"port": "80",

"path": "/"

},

"endpoints": [

{

"name": "[concat(parameters('webAppNamePrefix'),'-endpoint-','0')]",

"type": "Microsoft.Network/trafficManagerProfiles/azureEndpoints",

"properties": {

"targetResourceId": "[concat(parameters('webAppNamePrefix'),'-site-','0','-', uniqueString(resourceGroup().id))]",

"endpointStatus": "Enabled"

}

},

{

"name": "[concat(parameters('webAppNamePrefix'),'-endpoint-','1')]",

"type": "Microsoft.Network/trafficManagerProfiles/azureEndpoints",

"properties": {

"targetResourceId": "[concat(parameters('webAppNamePrefix'),'-site-','1','-', uniqueString(resourceGroup().id))]",

"endpointStatus": "Enabled"

}

},

{

"name": "[concat(parameters('webAppNamePrefix'),'-endpoint-','2')]",

"type": "Microsoft.Network/trafficManagerProfiles/azureEndpoints",

"properties": {

"targetResourceId": "[concat(parameters('webAppNamePrefix'),'-site-','2','-', uniqueString(resourceGroup().id))]",

"endpointStatus": "Enabled"

}

},

{

"name": "[concat(parameters('webAppNamePrefix'),'-endpoint-','3')]",

"type": "Microsoft.Network/trafficManagerProfiles/azureEndpoints",

"properties": {

"targetResourceId": "[concat(parameters('webAppNamePrefix'),'-site-','3','-', uniqueString(resourceGroup().id))]",

"endpointStatus": "Enabled"

}

}

]

}

}

1. Deploy the resource