



ARM

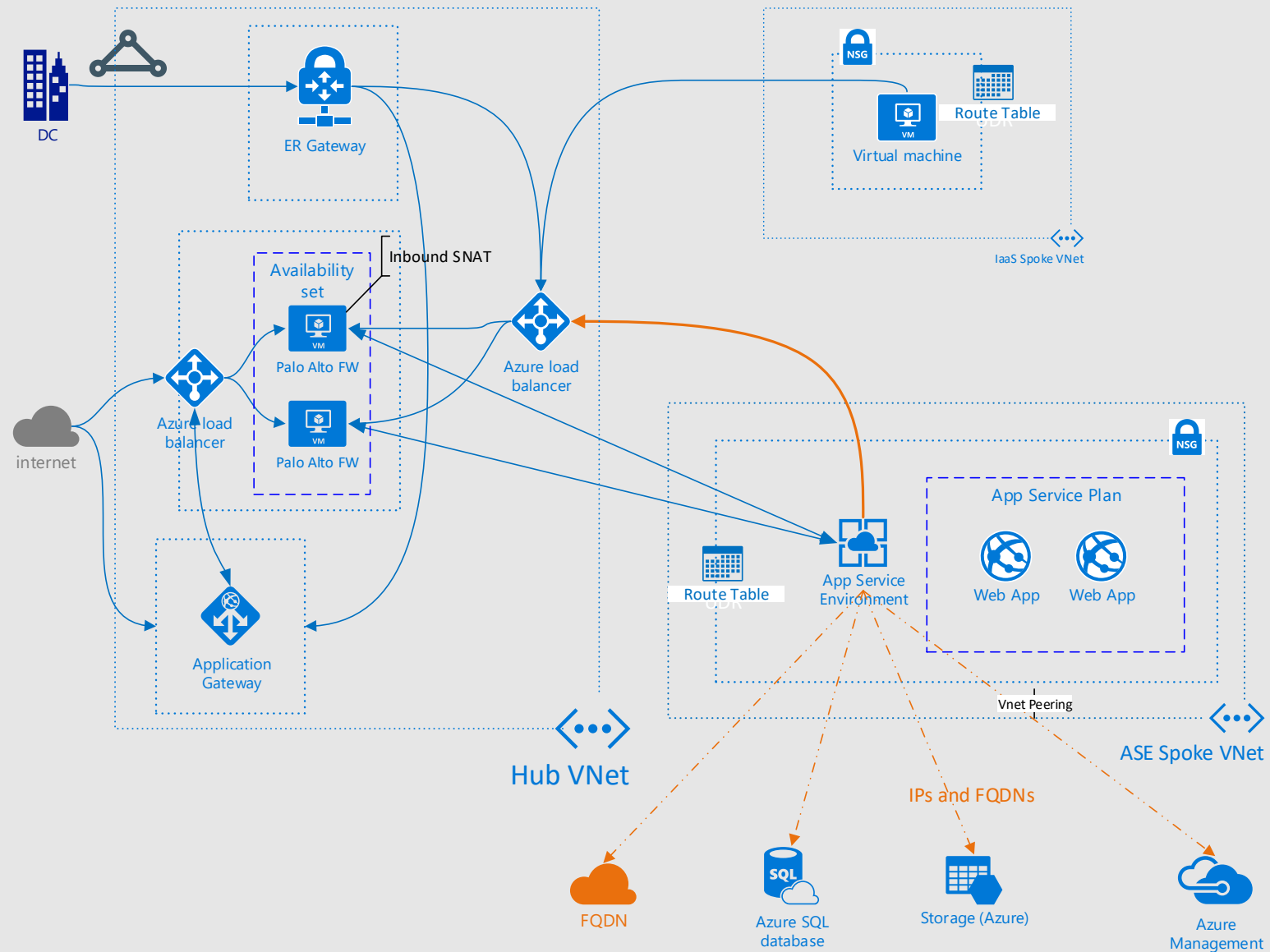
ARM Technology based development

Peter Dedrij
Partner Development Manager ISV
@PDedrij
Peter.Dedrij@microsoft.com





















You want to build this...



Which turns into this...

<input type="checkbox"/>	 Firewall-UDR	Route table	North Central US
<input type="checkbox"/>	 gomezhub3rgdiag175	Storage account	North Central US
<input type="checkbox"/>	 HubVNet	Virtual network	North Central US
<input type="checkbox"/>	 HubVNet-Gwy	Virtual network gateway	North Central US
<input type="checkbox"/>	 HubVNet-Gwy-IP	Public IP address	North Central US
<input type="checkbox"/>	 HubVNet-LocalGateway	Local network gateway	North Central US
<input type="checkbox"/>	 OnPrem-AllowRDP-Subnet	Network security group	South Central US
<input type="checkbox"/>	 OnPrem-AllowWeb-Subnet	Network security group	South Central US
<input type="checkbox"/>	 OnPrem-Connection	Connection	North Central US
<input type="checkbox"/>	 OnPremVNet	Virtual network	South Central US
<input type="checkbox"/>	 OnPremVNet-Gwy	Virtual network gateway	South Central US
<input type="checkbox"/>	 OnPremVNet-Gwy-IP	Public IP address	South Central US
<input type="checkbox"/>	 OnPremVNet-LocalGateway	Local network gateway	North Central US
<input type="checkbox"/>	 OnPremWebVM	Virtual machine	South Central US
<input type="checkbox"/>	 OnPremWebVM_disk2_64a38c9a8bad48b89e8c70c77384dd2c	Disk	South Central US
<input type="checkbox"/>	 OnPremWebVM_OsDisk_1_e9ba2fd1d5174797aceb2b1a136b1752	Disk	South Central US

What are ARM Templates?

Declarative files for creating Azure resources in a reliable, repeatable and auditable way.



Infrastructure as Code

Define Azure resources using text files.



Declarative Syntax

Declare how the resources should be and Azure Resource Manager "makes it so".



JSON

ARM Templates are JSON format text files. Edit them in Visual Studio Code (or other text editors). Version control them.



Meta-Language

Contains some programming language constructs such as functions and loops.



<https://docs.microsoft.com/en-us/azure/templates/>

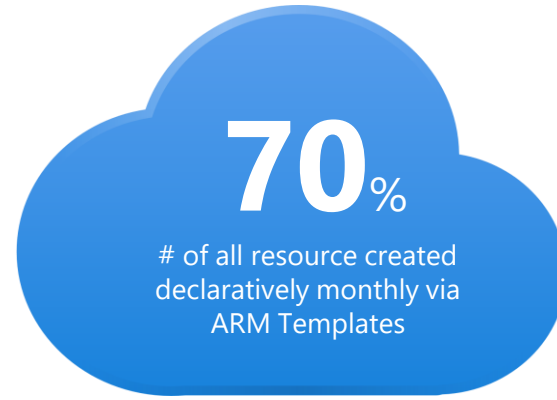
```
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": { ...
45 },
46   "variables": {
47     "webSiteName": "[concat('webSite', uniqueString(resourceGroup().id))]"
48   },
49   "resources": [
50     {
51       "type": "Microsoft.Web/serverfarms", // dependency for a web site
52       "apiVersion": "2015-08-01",
53       "name": "[parameters('hostingPlanName')]",
54       "location": "[resourceGroup().location]",
55       "tags": {
56         "environment": "Production" // change this if not prod
57       },
58       "sku": {
59         "name": "[parameters('skuName')]",
60         "capacity": "[parameters('skuCapacity')]"
61       },
62       "properties": {
63         "name": "[parameters('hostingPlanName')]"
64       }
65     },
66     {
```

IT as Code narrowing the gap between developers and IT



Cloud Center of Excellence (CCOE)

- Provision Tenant Infra
- Assign Policies
- Assign RBAC
- Create Subscriptions



Provision and manage the lifecycle of resources in a declarative way



App Team/DevOps

- Create Pipelines
- Deploy App Infra
- Deploy Apps

Declarative Approaches

1. Infra as Code
2. Policy as Code
3. Config as Code
4. Role based access as Code

Capabilities Provided



Day One and @ Scale

Resource Provider coverage from day one and ability to do large scale multi-region deployments



Simplify Authoring

- Automated Template generation via Azure Portal
- VSCode extension with intellisense, snippets etc.



Environment Setup

- Tenant, MG and Subscription level deployments
- One-click deployment solutions(Blueprints) to help meet regulations such as ISO, CIS, PCI, FedRAMP

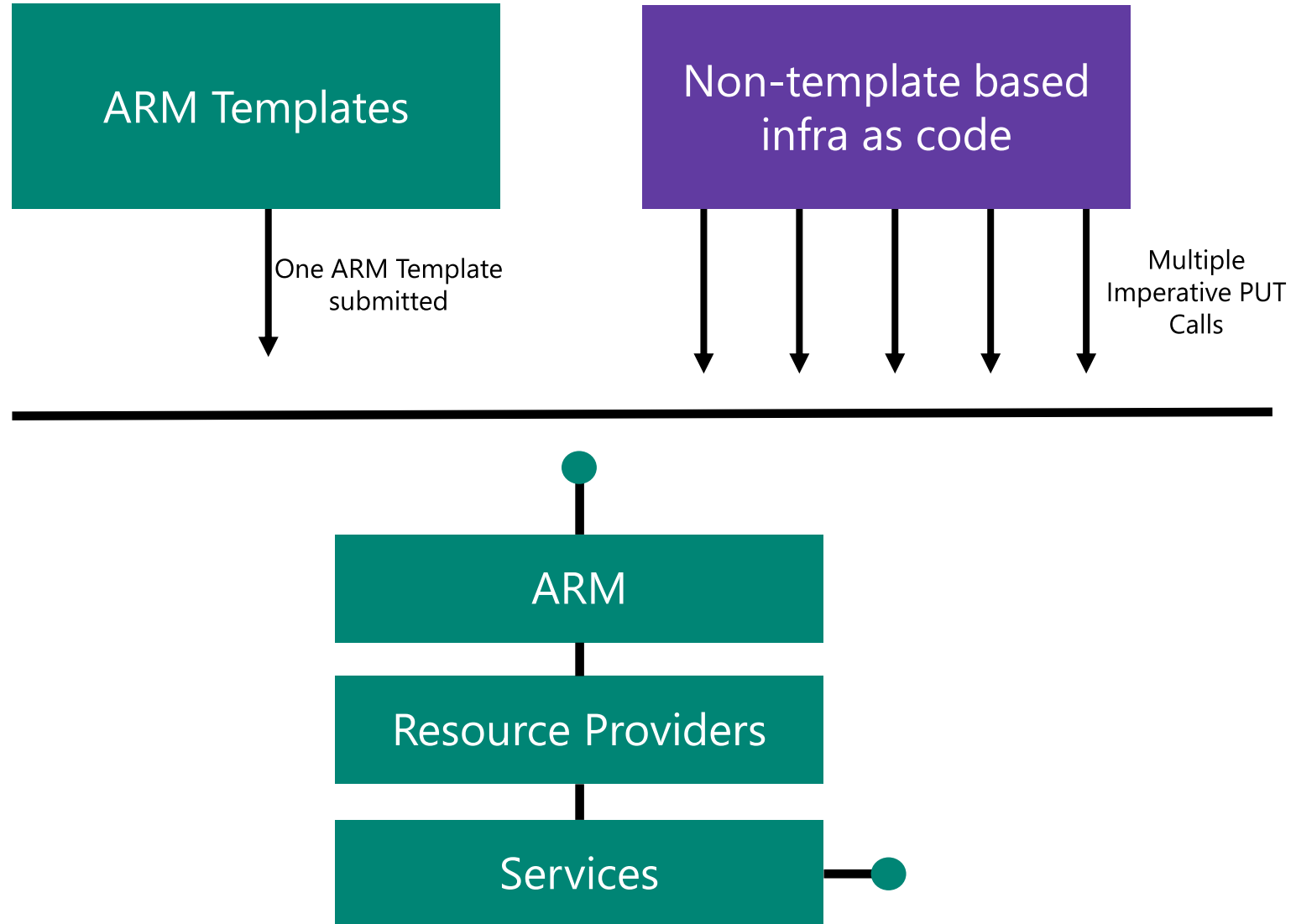


Integrated

- Azure Policy the remediations
- AzDevOps and GitHub tasks
- Provisioning flexibility with Terraform, Ansible, SNOW

Benefits of using ARM Templates vs. other Infra as Code solutions

- 1 Resource Provider coverage from day one**
When a new Azure Resource is released there is an ARM Template available
- 2 Deployments are a **tracked object** in the Azure Portal**
View the ARM Template deployment in the AzPortal, not the case for other infra as code solutions
- 3 Pre-Flight Checks**
ARM Deployments have pre-flight checks to make sure the template will deploy successfully
- 4 Azure Policy Remediations**
If a customer is using Azure Policy the remediations performed for non-compliant resources are done through ARM Templates.
- 5 Concurrent Executions**
3rd party IaC [solutions](#) do not allow current runs, plans or applies. No state management in ARM
- 6 Blueprint\Service Catalog**
Ability to publish approved set of templates and reference architectures across your organization. ISO, NIST, HIPAA etc.



Features of ARM Templates

Parameterized

Parameters can be used to configure resource attributes at deployment time. Allows generalization and re-use of ARM Templates.

Testable

Templates can be validated prior to deployment.

Modular

Templates can be broken into smaller, re-usable components and **linked** together at deployment time by using **Deployment** resources.

Templates can also be **nested** inside other templates.

Version Control

Using ARM Templates with version control allows your infrastructure to be reviewable, traceable and auditable.

Idempotent

Resources will only be changed or created if they have drifted out of state or need to be updated or created.



<https://docs.microsoft.com/en-us/azure/azure-resource-manager/template-best-practices>

Benefits

Deploy @ scale, manage, and monitor
Standardisation of customer environment
No dependency on customer knowledge
Comply to industry standards
Enforced security, privacy, ...