

# *Empowering ARM and JSON with 'Project Bicep'*



***Esther Barthel***

*@virtuEs\_IT*

*github.com/cognitionit*

*Microsoft MVP*



***Freek Berson***

*@fberson*

*github.com/fberson*

*Microsoft MVP*

# *A special thank you to the sponsors!*

## Partner Sponsor



## Sponsors



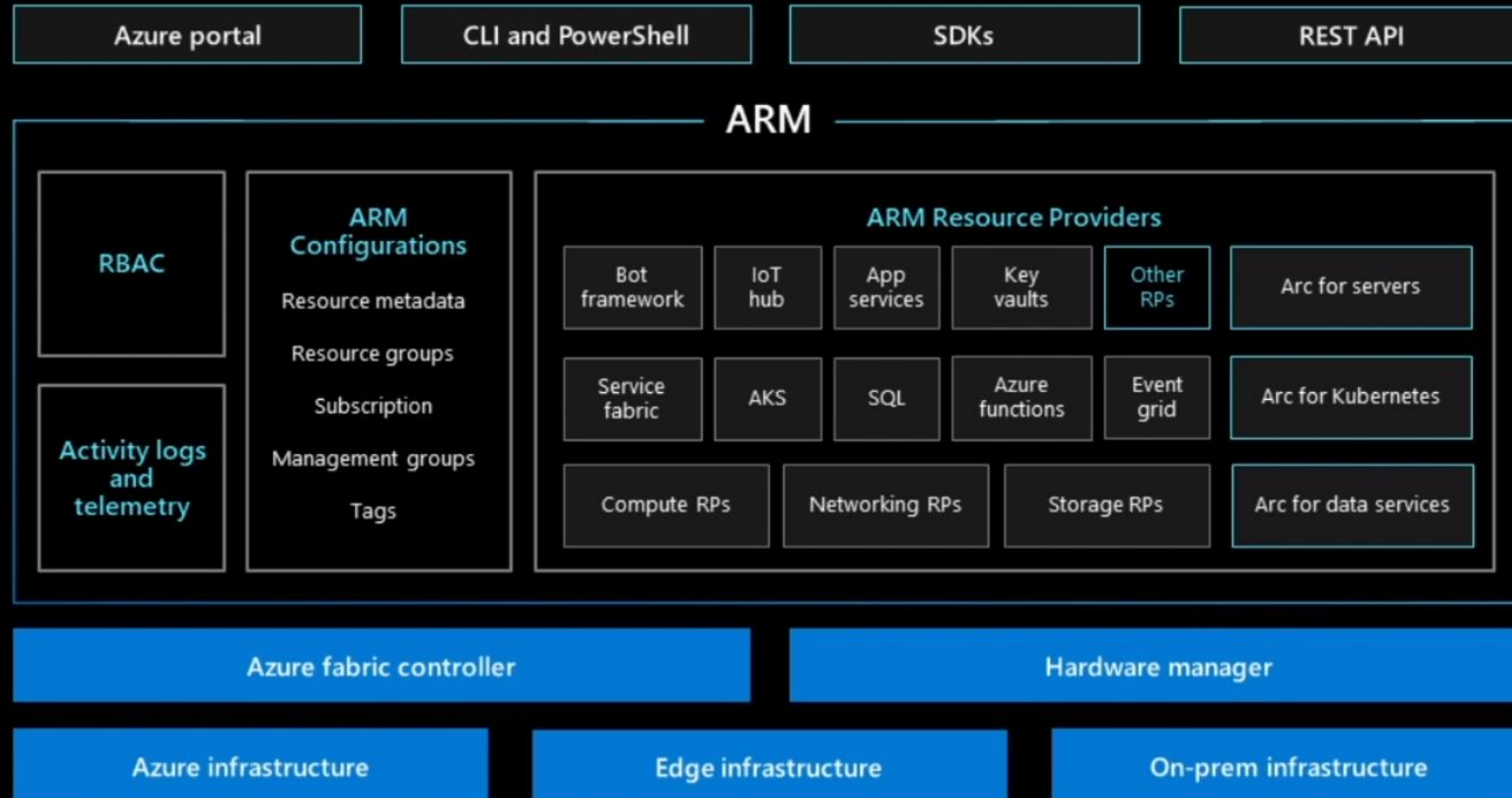
## Community Sponsors



# Agenda

- ✖ *Introduction to automation*
- ✖ *Azure Resource Manager & JSON*
- ✖ *What is Project 'Bicep'*
- ✖ *Demos, demos, demos!*

# Azure Resource Manager



# *How to get started with ARM & JSON*

... structure of an **Azure Resource Manager template**.


The template consists of **JSON** and expressions that you can use to construct values for your deployment.

# ARM Template

## Template format

In its simplest structure, a template has the following elements:

JSON

 Copy

```
{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
  "contentVersion": "",
  "apiProfile": "",
  "parameters": {  },
  "variables": {  },
  "functions": [  ],
  "resources": [  ],
  "outputs": {  }
}
```

# ARM Template

```
"parameters": {
  "<parameter-name>" : {
    "type" : "<type-of-parameter-value>",
    "defaultValue": "<default-value-of-parameter>",
    "allowedValues": [ "<array-of-allowed-values>" ],
    "minValue": <minimum-value-for-int>,
    "maxValue": <maximum-value-for-int>,
    "minLength": <minimum-length-for-string-or-array>,
    "maxLength": <maximum-length-for-string-or-array-parameters>,
    "metadata": {
      "description"
    }
  }
}
```

```
"functions": [
  {
    "namespace": "<namespace-for-functions>",
    "members": {
      "<function-name>": {
        "parameters": [
          {
            "name": "<parameter-name>",
            "type": "<type-of-parameter-value>"
          }
        ],
        "output": {
          "type": "<type-of-output-value>",
          "value": "<function-return-value>"
        }
      }
    }
  }
],
```

```
"variables": {
  "<variable-name>": "<variable-value>",
  "<variable-name>": {
    <variable-complex-type-value>
  },
  "<variable-object-name>": {
    "copy": [
      {
        "name": "<name-of-array-property>",
        "count": <number-of-iterations>,
        "input": <object-or-value-to-repeat>
```

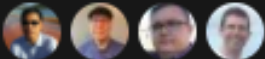
```
[
  {
    "name": "<variable-array-name>",
    "count": <number-of-iterations>,
    "input": <object-or-value-to-repeat>
```

```
"outputs": {
  "<output-name>": {
    "condition": "<boolean-value-whether-to-output-value>",
    "type": "<type-of-output-value>",
    "value": "<output-value-expression>",
    "copy": {
      "count": <number-of-iterations>,
      "input": <values-for-the-variable>
    }
  }
}
```

```
"resources": [
  {
    "condition": "<true-to-deploy-this-resource>",
    "type": "<resource-provider-namespace/resource-type-name>",
    "apiVersion": "<api-version-of-resource>",
    "name": "<name-of-the-resource>",
    "comments": "<your-reference-notes>",
    "location": "<location-of-resource>",
    "dependsOn": [
      "<array-of-related-resource-names>"
    ],
    "tags": {
      "<tag-name1>": "<tag-value1>",
      "<tag-name2>": "<tag-value2>"
    },
    "sku": {
      "name": "<sku-name>",
      "tier": "<sku-tier>",
      "size": "<sku-size>",
      "family": "<sku-family>",
      "capacity": <sku-capacity>
    },
    "kind": "<type-of-resource>",
    "copy": {
      "name": "<name-of-copy-loop>",
      "count": <number-of-iterations>,
      "mode": "<serial-or-parallel>",
      "batchSize": <number-to-deploy-serially>
    },
    "plan": {
      "name": "<plan-name>",
      "promotionCode": "<plan-promotion-code>",
      "publisher": "<plan-publisher>",
      "product": "<plan-product>",
      "version": "<plan-version>"
    }
  }
]
```

# ARM Template


## Quickstart: Create and deploy ARM templates by using the Azure portal








06/29/2020 • 6 minutes to read • 

Learn how to generate an Azure Resource Manager template (ARM template) using the Azure portal, and the process of editing and deploying the template from the portal. ARM templates are JSON files that define the resources you need to deploy for your solution. To understand the concepts associated with deploying and managing your Azure solutions, see [template deployment overview](#).



# ARM Template

 **Microsoft Azure**

      **CLOUDDEVOPSNINJA** 

Home > Windows Virtual Desktop >

## Create a host pool

**Basics** Virtual Machines Workspace Tags Review + create

### Project details

Subscription \* ⓘ

Microsoft Azure Sponsorship

Resource group \* ⓘ

Select a resource group

[Create new](#)

Host pool name \*

Location \* ⓘ

East US

Metadata will be stored in Azure geography associated with (US) East US

Validation environment ⓘ

☒ No ☐ Yes

### Host pool type

If you select pooled (shared), users will still be able to access their personalization and user data, using FSLogix.


Host pool type \*

Select a type


Review + create


Next: Virtual Machines >

# ARM Template

 Microsoft Azure


Search resources, services, and docs (G+ /)



CLOUDEDEVOPSNINJA 

Home > Windows Virtual Desktop >

## Create a host pool

 Validation passed.

Basics Virtual Machines Workspace Tags Review + create

**Basics**

Subscription	Microsoft Azure Sponsorship
Resource group	rg-wvd-infra
Host pool name	hp-demo
Location	East US
Host pool type	Pooled
Max session limit	10
Load balancing algorithm	Breadth-first

**Virtual Machines**

Resource group	rg-wvd-resources
----------------	------------------

Create

< Previous

Download a template for automation

# ARM Template

The screenshot shows the Microsoft Azure portal interface. At the top, the navigation bar includes the Microsoft Azure logo, a search bar, and user information (CLOUDEDEVOPSNINJA). The breadcrumb trail indicates the path: Home > Windows Virtual Desktop > Create a host pool > Template. The 'Template' tab is active, and the 'Download' button is highlighted with a yellow box. Below the navigation bar, a blue information banner explains that ARM templates automate resource deployment. The main content area is divided into three tabs: Template, Parameters, and Scripts. The 'Template' tab is selected, displaying a JSON ARM template. On the left, a tree view shows the template's structure: Parameters (56), Variables (15), and Resources (5). The Resources section is expanded, showing four resource types: a parameter for hostpoolName, a variable for appGroupName, a deployment resource for workspace-linkedTemplate, and another deployment resource for AVSet-linkedTemplate. The JSON code is displayed in a dark-themed editor with line numbers 1 through 14.

Microsoft Azure Search resources, services, and docs (G+/)

Home > Windows Virtual Desktop > Create a host pool > Template

Download Add to library (preview) Deploy

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. [Learn more about template deployment.](#)

☒ Include parameters ⓘ

Template Parameters Scripts

> ⚙ Parameters (56)  
> 📄 Variables (15)  
✓ 🗑 Resources (5)

- 🔗 [parameters('hostpoolName')] (Microsoft.DesktopVirtualization/h
- 🔗 [variables('appGroupName')] (Microsoft.DesktopVirtualization/a
- 🔗 [concat('Workspace-linkedTemplate-', parameters('deploymentId'))] (Microsoft.Resources/deployments
- 🔗 [concat('AVSet-linkedTemplate-', parameters('deploymentId'))] (Microsoft.Resources/deployments

```
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/
deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "nestedTemplatesLocation": {
6       "type": "string",
7       "metadata": {
8         "description": "The base URI where artifacts required by this
template are located."
9       },
10      "defaultValue": "https://catalogartifact.azureedge.net/publicartifacts/
Microsoft.Hostpool-ARM-1.0.18-preview/"
11    },
12    "artifactsLocation": {
13      "type": "string",
14      "metadata": {
```

# ARM Template – Reference Guide

Microsoft

Docs

Documentation

Learn

Q&A

Code Samples

Search

Azure

Product documentation

Architecture

Learn Azure

Develop

Resources

Portal

Free account

Azure / Azure Templates

Bookmark

Share

Filter by title

Reference

AAD

Compute

Availability Sets

Cloud Services

Disk Accesses

Disk Encryption Sets

Disks

Galleries

Host Groups

Images

Proximity Placement Groups

Snapshots

Ssh Public Keys

Virtual Machines

Define resources in ARM templates

12/21/2020 • 2 minutes to read •

When creating Azure Resource Manager templates, you need to understand what resource types are available. This article provides ARM template reference documentation for the following resource types:

Learn how to create templates

For an introduction to working with templates, see [ARM template](#).

To learn about ARM templates through a guided tour, see [Deploy and manage resources in Azure by using the Azure portal](#).

Microsoft recommends that you use VS Code with the Azure Resource Manager tools extension, or the Azure CLI. For more information, see [Quickstart: Create and deploy an ARM template](#).

Microsoft.Network virtualNetworks

12/28/2020 • 12 minutes to read •

API Versions: Latest

Is this page helpful?

Yes No

In this article

Template format

Property values

Quickstart templates

## Template format

To create a Microsoft.Network/virtualNetworks resource, add the following JSON to the resources section of your template.

```
JSON
{
  "name": "string",
  "type": "Microsoft.Network/virtualNetworks",
  "apiVersion": "2020-07-01",
  "location": "string",
  "tags": {},
  "extendedLocation": {
    "name": "string",
    "type": "EdgeZone"
  },
  "properties": {
    "addressSpace": {
      "addressPrefixes": [
        "string"
      ]
    }
  }
}
```

VirtualNetworkPropertiesFormat object				Is this page helpful?
Name	Type	Required	Value	Yes No
addressSpace	object	No	The AddressSpace that contains an array of IP address ranges that can be used by subnets. - <a href="#">AddressSpace object</a>	In this article <a href="#">Template format</a> <a href="#">Property values</a> <a href="#">Quickstart templates</a>
dhcpOptions	object	No	The dhcpOptions that contains an array of DNS servers available to VMs deployed in the virtual network. - <a href="#">DhcpOptions object</a>	
subnets	array	No	A list of subnets in a Virtual Network. - <a href="#">Subnet object</a>	
ipAllocations	array	No	Array of IpAllocation which reference this VNET. - <a href="#">SubResource object</a>	

# ARM Template – Reference Guide

```
"parameters": {  
  "dnsServer": {  
    "type": "string",  
    "metadata": {  
      "description": "The vNet DNS server to enable the domain-join of Session Hosts"  
    }  
  }  
},  
"resources": [  
  {  
    1 child: string (subnets)  
    "name": "[variables('vnetName')]",  
    "type": "Microsoft.Network/virtualNetworks",  
    "apiVersion": "2020-07-01",  
    "location": "[resourceGroup().location]",  
    "properties": {  
      "addressSpace": {"addressPrefixes": ["10.0.0.0/16"]},  
      "dhcpOptions": {"dnsServers": ["parameters('dnsServer')"]},  
    }  
  }  
]
```

# ARM Template learning path



<https://bit.ly/3qZGNj1>

**MODULE 1:** Deploy Azure infrastructure by using ARM templates

**MODULE 2:** Deploy to multiple Azure environments by using ARM template features

**MODULE 3:** Preview changes and validate Azure resources by using what-if and the ARM template test toolkit


**MODULE 4:** Automate the deployment of ARM templates by using GitHub Action

**MODULE 5:** Extend ARM templates by using deployment scripts

**MODULE 6:** Manage complex cloud deployments by using advanced ARM template features



# ARM Template – Azure Quickstart Templates



Microsoft Azure

Contact Sales

Search

My account

Portal

Esther

Overview

Solutions

Products

Documentation

Pricing

Training

Marketplace

Partners

Support

Blog

More

Free account

## Azure Quickstart Templates

Deploy Azure resources through the Azure Resource Manager with community contributed templates to get more done. Deploy, learn, fork and contribute back.

Sort by:

Date updated

Template name

Author name

Most popular

Showing all 978 templates. Refine

Deploy a simple Windows VM

This template allows you to deploy a simple Windows VM using a few different options for the Windows version, using the latest patched version. This will deploy an A2 size VM in the resource group location and return the FQDN of the VM.

by Brian Moore

Last updated: 9/18/2020

Deploy to Azure

Browse on GitHub

Parameters

PARAMETER NAME	DESCRIPTION
OSVersion	The Windows version for the VM. This will pick a fully patched image of this given Windows version.
vmSize	Size of the virtual machine.
location	Location for all resources.
vmName	Name of the virtual machine.

Use the template

PowerShell

```
New-AzResourceGroup -Name <resource-group-name> -Location <resource-group-location> #use this command when you need to create a new resource group for your deployment
New-AzResourceGroupDeployment -ResourceGroupName <resource-group-name> -TemplateUri https://raw.githubusercontent.com/Azure/azure-quickstart-templates/master/101-vm-simple-windows/azuredeploy.
```

Install and configure Azure PowerShell

Resource Types:

All

Microsoft.Aad (1)

Microsoft.Analysisservices (1)

Microsoft.ApiManagement (11)

Microsoft.Appconfiguration (2)

Microsoft.Appplatform (1)

Microsoft.Attestation (1)

Microsoft.Authorization (15)

# ARM Template – Automatic deployments

```
# ARM Template file
$jsonARMTemplateFile = ".\ARM-WVDNewHostpool.template.json"
$jsonARMPParameterFile = ".\ARM-WVDNewHostpool.parameter.json"


# Create WVD Hostpool, based on ARM Template
New-AzResourceGroupDeployment -ResourceGroupName "rg-wvd-infra" `
    -TemplateFile $jsonARMTemplateFile `
    -TemplateParameterFile $jsonARMPParameterFile `
    -administratorAccountPassword $secureDomainAdminPassword `
    -vmAdministratorAccountPassword $secureLocalAdminPassword `
    -Verbose
```



# ARM Template – Automatic deployments

✖ Deployment failed. [Click here for details](#) →

## Your deployment failed


 Deployment name: ARM-WVDN  
Subscription:  
Resource group: rg-wvd-infra

Deployment details (Download)

Resource
✖ vmCreation-linkedTemplate-
✓ AVSet-linkedTemplate-
✓ Workspace-linkedTemplate-
✓ wvd-hp-demo-DAG
✓ wvd-hp-demo

✖ Deployment failed. [Click here for details](#) →

## Your deployment failed

 Deployment name: vmCreation-linkedTemplate-  
Subscription:  
Resource group: rg-wvd-resources

Start time: 1/27/2021, 5:37:38 PM  
Correlation ID: d26c3483-452c-462e-8838-8ba39d8490be

Deployment details (Download)

Resource	Type	Status	Operation details
✖ wvd-sh-0-nic	Microsoft.Network/networkl...	BadRequest	<a href="#">Operation details</a>
✖ wvd-sh-1-nic	Microsoft.Network/networkl...	BadRequest	<a href="#">Operation details</a>
✓ NSG-linkedTemplate	Microsoft.Resources/deploy...	OK	<a href="#">Operation details</a>

# ARM Template – Automatic deployments

```
1 {  
2   "code": "DeploymentFailed",  
3   "message": "At least one resource deployment operation  
failed. Please list deployment operations for details.  
Please see https://aka.ms/DeployOperations for usage  
details.",  
4   "details": [  
5     {  
6       "code": "InvalidResourceReference",  
7       "message": "Resource /subscriptions/  
                                /resourceGroups/  
rg-wvd-resources/providers/Microsoft.Network/  
virtualNetworks/vnet-wvd-resource/subnets/default  
referenced by resource /subscriptions/  
                                /resourceGroups/  
rg-wvd-resources/providers/Microsoft.Network/  
networkInterfaces/wvd-sh-0-nic was not found. Please make  
sure that the referenced resource exists, and that both  
resources are in the same region."  
8     },
```

# ARM Template – Troubleshooting

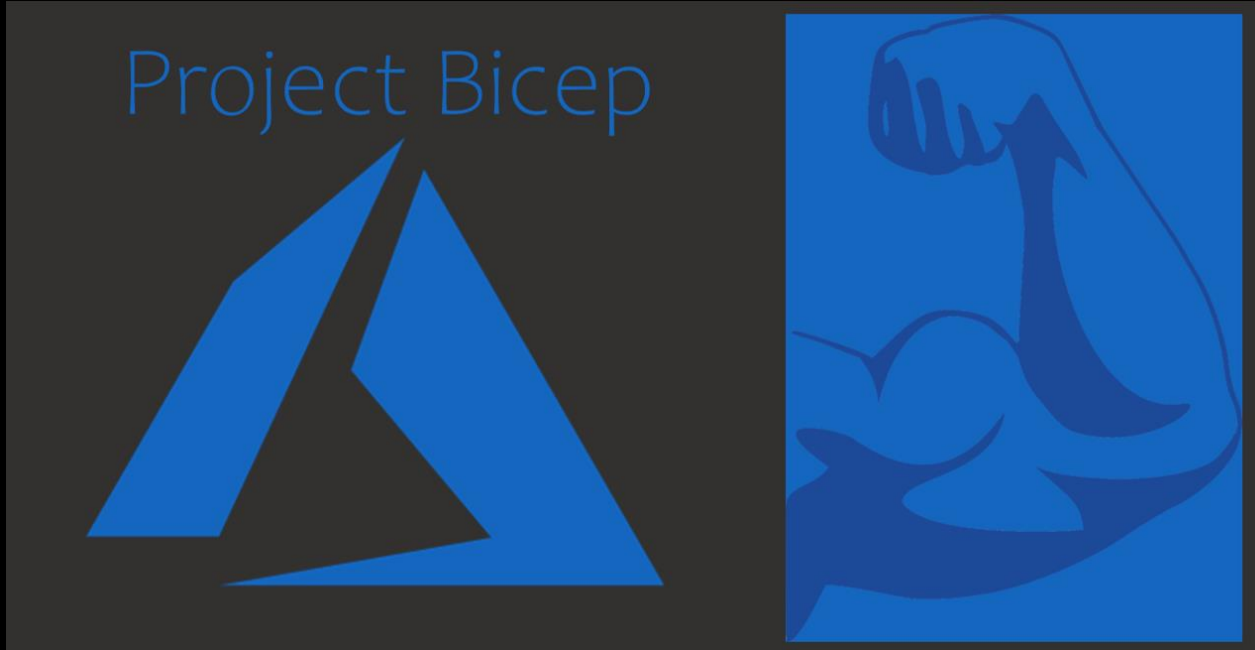
```
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": { ...
324 },
325 "variables": { ...
346 },
347 "resources": [
348   {
349     "apiVersion": "2018-05-01",
350     "name": "[concat('UpdateHostPool-', parameters('deploymentId'))]",
351     "type": "Microsoft.Resources/deployments",
352     "resourceGroup": "[parameters('hostpoolResourceGroup')]",
353     "condition": "[not(empty(parameters('hostpoolResourceGroup')))]",
354     "properties": { ...
369   }
370 },
371 { ...
403 }
404 {
405   "apiVersion": "2018-05-01",
406   "name": "[concat('vmCreation-linkedTemplate-', parameters('deploymentId'))]",
407   "resourceGroup": "[parameters('vmResourceGroup')]",
408   "dependsOn": [
409     "[concat('AVSet-linkedTemplate-', parameters('deploymentId'))]"
410   ],
411   "type": "Microsoft.Resources/deployments",
412   "properties": { ...
522 }
523 },
524 ],
525 "outputs": {
526   "rdshVmNamesObject": {
527     "value": "[variables('rdshVmNamesOutput')]",
528     "type": "object"
529   }
530 }
531 }
```

## Challenges:

- 531 lines of code
- complex JSON formatting
- external dependencies:
  - linked templates



# What is Project 'Bicep'



*"..Bicep is a **Domain Specific Language (DSL)** for deploying Azure resources declaratively. It aims to **drastically simplify the authoring experience** with a cleaner syntax and better support for modularity and code re-use. Bicep is a transparent abstraction over ARM and ARM templates.*

# Project 'Bicep'

*Simple declarative language to provision infrastructure to Azure.*

## Intuitive

Easy to read and to author

## Transpiles to ARM Templates

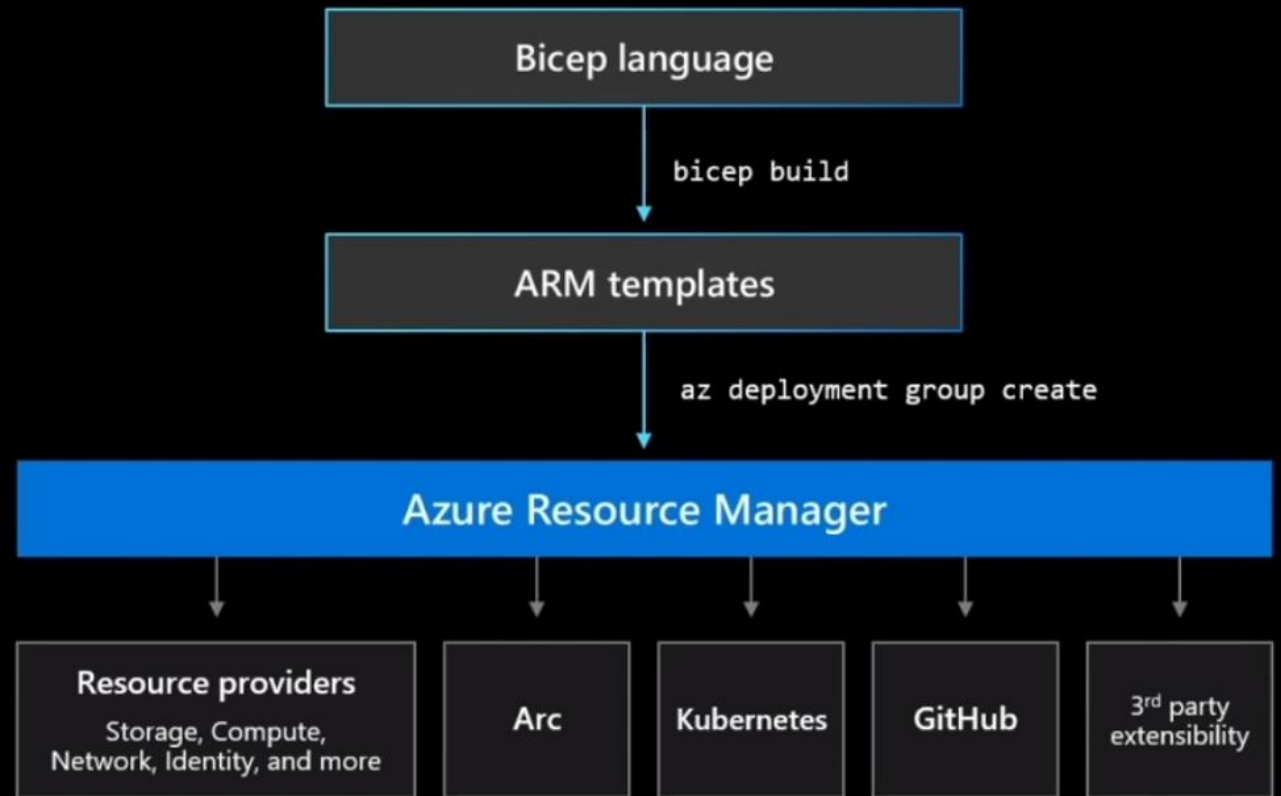
Leverage ARM template knowledge and investments

## Modular

Abstract common blocks of code into reusable parts

## Open Source

Transparency and community



# Installing project 'Bicep'

## 1. Install the Bicep CLI (required)

```
# Create the install folder
$installPath = "$env:USERPROFILE\.bicep"
$installDir = New-Item -ItemType Directory -Path $installPath -Force
$installDir.Attributes += 'Hidden'
# Fetch the latest Bicep CLI binary
(New-Object Net.WebClient).DownloadFile("https://github.com/Azure/bicep/releases/latest/download/bicep-win-x64.exe", "$installPath\bicep.exe")
# Add bicep to your PATH
$currentPath = (Get-Item -path "HKCU:\Environment" ).GetValue('Path', '', 'DoNotExpandEnvironmentNames')
if (-not $currentPath.Contains("%USERPROFILE%.bicep")) { setx PATH ($currentPath + ";%USERPROFILE%.bicep") }
if (-not $env:path.Contains($installPath)) { $env:path += ";$installPath" }
# Verify you can now access the 'bicep' command.
bicep --help
# Done!
```

## 2. Install the Bicep VS Code extension(optional)

```
# Fetch the latest Bicep VSCode extension
$vsixPath = "$env:TEMP\vscode-bicep.vsix"
(New-Object Net.WebClient).DownloadFile("https://github.com/Azure/bicep/releases/latest/download/vscode-bicep.vsix", $vsixPath)
# Install the extension
code --install-extension $vsixPath
# Clean up the file
Remove-Item $vsixPath
# Done!
```

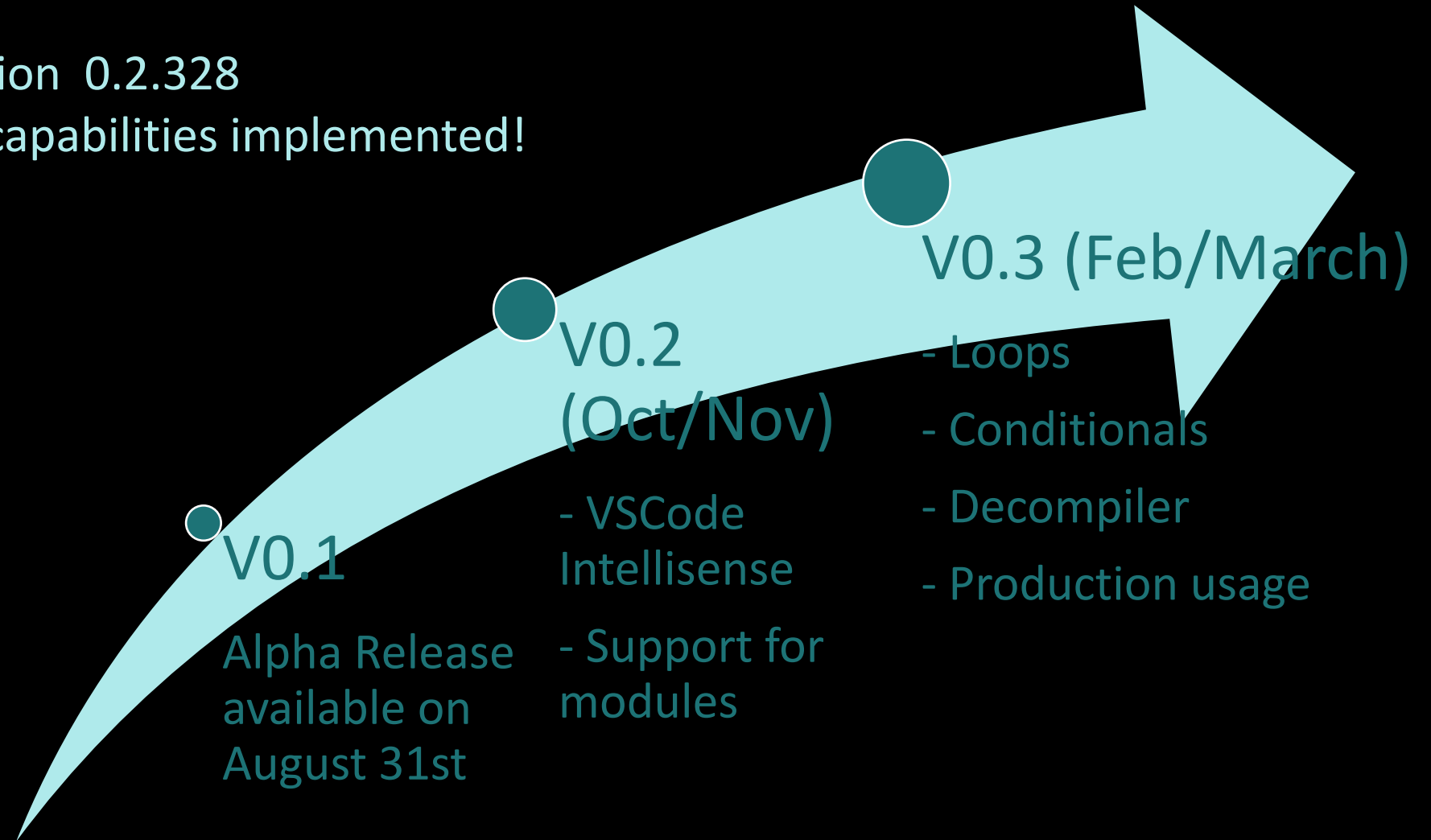
# Demo



# Road map

Current release: CLI version 0.2.328

~90% of ARM template capabilities implemented!





# Breaking changes planned

	BEFORE		AFTER
Param Decorators <a href="#">Issue #64</a>	<pre>param authType string {   default: 'password'   allowed: [     'sshPublicKey'     'password'   ] }</pre>	➔	<pre>@allowed([   'sshPublicKey'   'password' ) param authType string = 'password'</pre>
Outputs <a href="#">Issue #569</a>	<pre>param adminUsername string ... output username string = adminUsername</pre>	➔	<pre>param adminUsername string ... output adminUsername as string</pre>

# Call to action: aka.ms/bicep

Install guides, tutorials, example code & playgrounds!

The screenshot shows the GitHub repository for Azure/bicep. The repository has 85 watchers, 827 stars, and 120 forks. The main branch is 'main'. The repository is described as 'Bicep is a declarative language for describing and deploying Azure resources'. The repository includes a README, a LICENSE, and a list of releases. The latest release is v0.2.328 (alpha), released yesterday. The repository also has a list of contributors and a list of languages used in the project.

file	description	last commit
.config	official build prereqs (#1021)	2 months ago
.github	Stop triggering build on tags (#1345)	3 days ago
pipelines	Fix official VSIX version to unblock publishing to VS gallery (#1365)	yesterday
vscode	Fix launch.json (#1219)	last month
Formula	Update homebrew to use v0.2.212 (#1192)	last month
docs	Filter available types by target scope (#1321)	9 days ago
scripts	Script to populate existing release with signed build artifacts (#1344)	3 days ago
src	Refactor scope naming (#1368)	15 hours ago
gitattributes	Run VSIX tests on all platforms (#930)	2 months ago
gitignore	Update CONTRIBUTING.md with instruction to run Bicep VSCode extensi...	2 months ago
gitmodules	Switched from submodule to type nugets (#987)	2 months ago
Bicep.sln	Detecting duplicate resource and module names (#1204)	11 days ago
Bicep.sln.DotSettings	Parse errors no longer spill over multiple new lines (#77)	6 months ago
CODE_OF_CONDUCT.md	Initial CODE_OF_CONDUCT.md commit	9 months ago
CONTRIBUTING.md	Detecting duplicate resource and module names (#1204)	11 days ago
LICENSE	Updating LICENSE to template content	9 months ago
NuGet.config	Replace expressions nugget package with stripped down library (#182)	5 months ago
README.md	Added link to Bicep PowerShell Module in readme (#1333)	4 days ago
SECURITY.md	Initial SECURITY.md commit	9 months ago
SetBaseline.ps1	Fixed baseline script (#1108)	last month
azure-pipelines.yml	Updated to .net 5 (#911)	2 months ago

The screenshot shows the Azure/bicep GitHub repository page, specifically the 'examples' directory. The repository has 85 watchers, 827 stars, and 120 forks. The main branch is 'main'. The repository is described as 'Bicep is a declarative language for describing and deploying Azure resources'. The repository includes a README, a LICENSE, and a list of releases. The latest release is v0.2.328 (alpha), released yesterday. The repository also has a list of contributors and a list of languages used in the project.

file	description	last commit
main.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
main.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago
readme.md	Added readme.md (#1047)	2 months ago
wvd-LogAnalytics.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-LogAnalytics.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-backplane-module.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-backplane-module.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-fileservices-module.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-fileservices-module.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-monitor-diag.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-monitor-diag.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-network-module.bicep	multi-module WVD deployment with some prereqs (#1010)	2 months ago
wvd-network-module.json	multi-module WVD deployment with some prereqs (#1010)	2 months ago

The screenshot shows the Azure/bicep GitHub repository page, specifically the 'examples' directory. The repository has 85 watchers, 827 stars, and 120 forks. The main branch is 'main'. The repository is described as 'Bicep is a declarative language for describing and deploying Azure resources'. The repository includes a README, a LICENSE, and a list of releases. The latest release is v0.2.328 (alpha), released yesterday. The repository also has a list of contributors and a list of languages used in the project.

page_type	languages	description	products
resources	md, json, bicep	Multi-module Bicep project that deploys a WVD environment in Azure including some prerequisites that WVD generally needs.	azure, windows-virtual-desktop

Multi-module Bicep project for WVD



*Esther Barthel*  
*@virtuEs\_IT*  
*github.com/cognitionit*



*Freek Berson*  
*@fberson*  
*github.com/fberson*

