



# Empowering your WVD deployments with Project Bicep, experiences from the field!



Freek Berson - Wortell  
@fberson  
[github.com/fberson](https://github.com/fberson)  
Microsoft MVP







Thank you to our partners





# 2017, Orlando FL

## RECAP, WHAT HAS BEEN CREATED? #1/2

- 6 Managed Disks
- 6 NIC's
- 6 VM's
- 3 Availab
- Azure SC
- Static pu
- 2 load b

## RECAP, WHAT HAS BEEN CREATED? #2/2

- All VM's joined to AD Domain
- Anti-virus & Anti Malware configured
- Bginfo configured
- Full HA RDS Deployment
- 2 RD Connection Broker / Licensing
- 2 RD Gateway / Web Access
- 2 RD Session Host
- SSL Certificates configured for all roles
- Session Collection created
- Sample RemoteApp published
- RD Web Access branding
- RD CAP / RAP Policies
- Session Time Out settings configured

@fberson

TheMicrosoftPlatform.net



# What is the main driver behind automation in Azure?

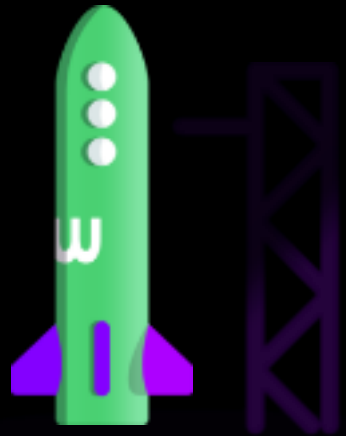


+

Having fast, and consistent yet customizable deployments that are reproducible in any Azure Subscription.



# Agenda



Quick intro into ARM, JSON & AIB

Project 'Bicep' architecture

Full WVD deployment demo!



TECH FEST 2021

*Empowering your WVD deployments with Project Bicep!*



## Step 3. Azure DevOps



### Your subscription—your control

#### Desktops and remote apps



Full Desktop



RemoteApp



Windows 10 Enterprise  
multi-session



Windows 10 Enterprise



Windows Server  
2012 R2 and newer



Windows 7 Enterprise  
Full Desktop

#### Management and policies



Image, app, and profile  
management



User density, VM sizing,  
and scaling policies



User management and  
identity



Networking policies



## Step 2. Azure Image Builder



## Step 1. WVD deployment



### Managed by Microsoft

#### Windows Virtual Desktop Service



Clients



Diagnostics



Gateway



Management



Broker



Load balancing

#### Infrastructure



Compute

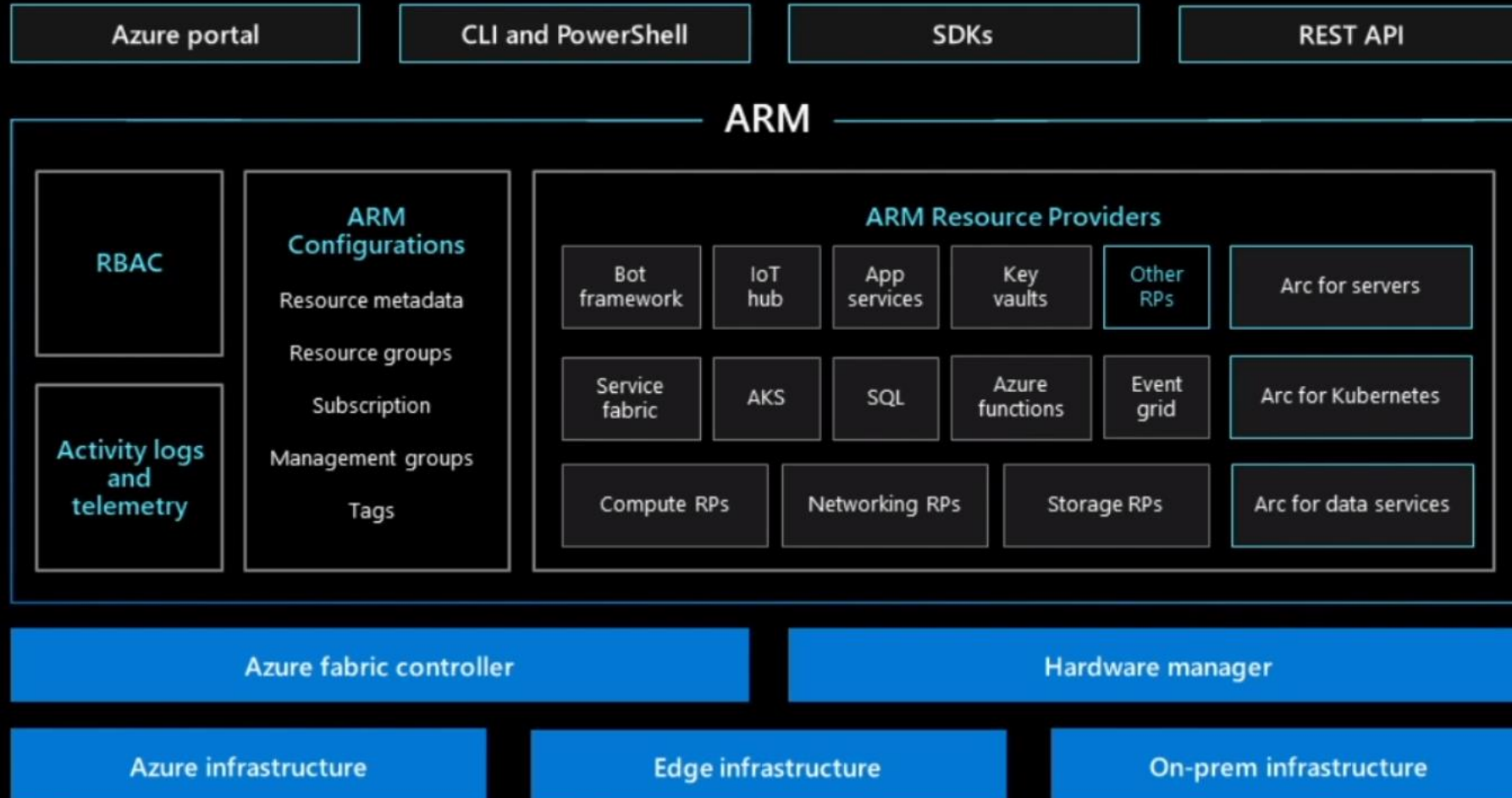


Storage

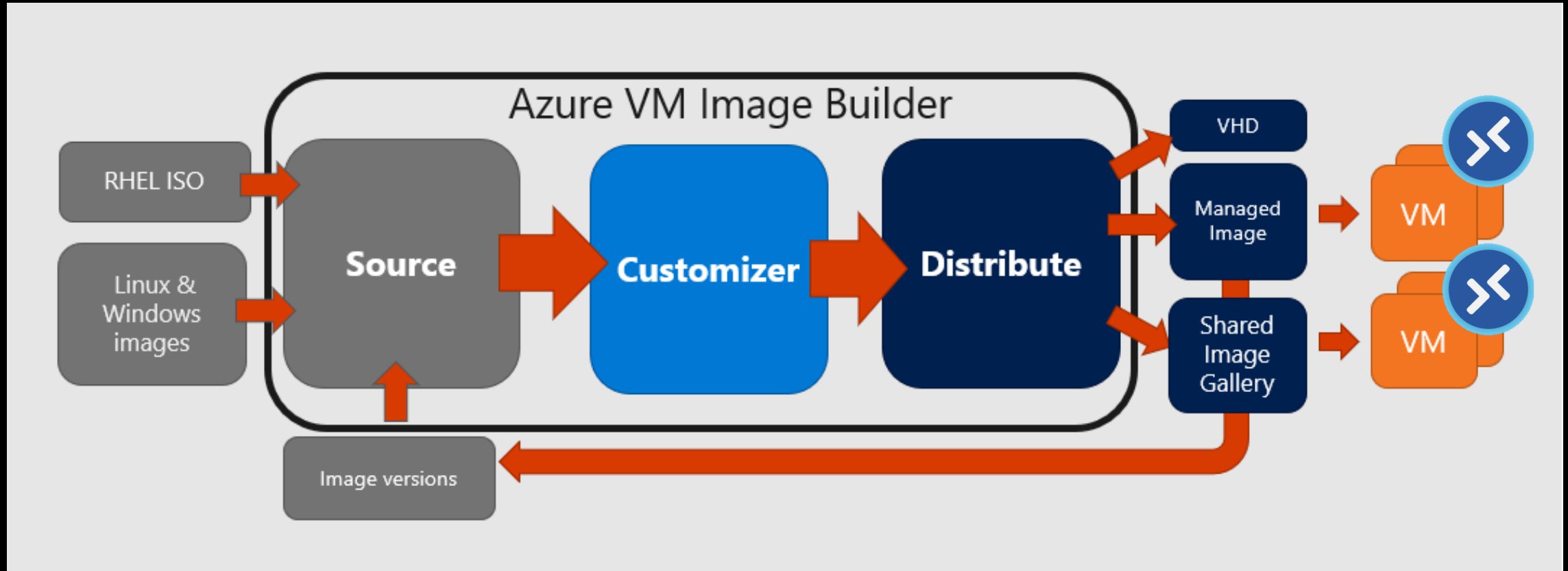


Networking

# Azure Resource Manager



# Azure Image Builder Preview





# Azure DevOps

The screenshot displays the Azure DevOps web interface. The top navigation bar shows the breadcrumb path: **WVD Host Pool Maintenance** / **Pipelines** / **Releases** / **WVD Acceptance Host Pool** / **Release-32**. The left sidebar contains a menu with the following items: **Overview**, **Boards**, **Repos**, **Pipelines** (with a sub-menu), **Environments**, **Releases**, **Library**, **Task groups**, **Deployment groups**, **Test Plans**, and **Artifacts**. The main content area is titled **WVD Acceptance Host Pool > Release-32**. Below the title, there are tabs for **Pipeline**, **Variables**, and **History**, along with action buttons: **Deploy**, **Cancel**, **Refresh**, **Edit**, and a more options menu. The **Release** section shows a **Manually triggered** release by **Freek Berson** on **21/02/2021, 20:58**. Under the **Artifacts** section, two artifacts are listed: **\_WVD Host Pool Maint...** (ID: 49137197, branch: main) and **\_fberson\_wvd** (ID: 6ec830c68, branch: master). The **Stages** section shows a single stage named **AddHostsToExistingH...** which has **Succeeded** on **21/02/2021, 21:09**.

# Infrastructure as code

**Language** Azure Resource Manager accepts JavaScript Object Notation (JSON) templates that comply with a JSON schema. JSON is an industry standard, human readable language.

```
4. Empty.json
1  {
2    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3    "contentVersion": "1.0.0.0",
4    "parameters": {},
5    "functions": [],
6    "variables": {},
7    "resources": [],
8    "outputs": {}
9  }
```

**Tooling** Although it is possible to create a template for a complete application in simple text editor, there is sophisticated tooling available that provides you with a better authoring experience for Azure Resource Manager templates.

# JSON / ARM challenges

```
262 "apiVersion": "[variables('apiVersion')]",
263 "type": "Microsoft.Compute/virtualMachines",
264 "name": "[concat(variables('azureVMName'), '-', copyindex(parameters('sequenceStartNumberWVDHost')))]",
265 "comments": "This resources creates Virtual Machines that wil host the RDSH role",
266 "tags": {
267   "displayName": "WVD Session Host Virtual Machines"
268 },
269 "location": "[resourceGroup().location]",
270 "copy": {
271   "name": "[concat(variables('azureVMName'), 'vm-loop')]",
272   "count": "[parameters('numberOfInstancesWVD')]"
273 },
274 "dependsOn": [
275   "[concat('Microsoft.Network/networkInterfaces/', variables('azureVMName'), '-', copyindex(parameters('sequenceStartNumberWVDHost')), variables('networkAdapterNamePostFix'))]",
276   "[concat('Microsoft.Compute/availabilitySets/', parameters('existingAvailabilitySetName'))]"
277 ],
278 "properties": {
279   "licenseType": "Windows_Client",
280   "hardwareProfile": {
281     "vmSize": "[parameters('virtualMachinesSizeWVD')]"
282   },
283   "availabilitySet": {
284     "id": "[resourceId('Microsoft.Compute/availabilitySets', parameters('existingavailabilitySetName'))]"
285   },
286   "osProfile": {
287     "computerName": "[concat(parameters('hostNamePrefixWVD'), '-', copyindex(parameters('sequenceStartNumberWVDHost')))]",
288     "adminUsername": "[concat(parameters('hostNamePrefixWVD'), '-', copyindex(parameters('sequenceStartNumberWVDHost')), '-adm')]",
289     "adminPassword": "[parameters('localAdminPassword')]",
290     "windowsConfiguration": {
291       "timeZone": "[variables('vmTimeZone')]"
292     }
293   },
294   "storageProfile": {
295     "osDisk": {
296       "name": "[concat(variables('azureVMName'), '-', copyindex(parameters('sequenceStartNumberWVDHost')), '-', variables('virtualmachineosdisk').diskName)]",
297       "managedDisk": {
298         "storageAccountType": "[variables('storage').type]"
299       },
300       "osType": "windows",
301       "caching": "[variables('virtualmachineosdisk').cacheOption]",
302       "createOption": "[variables('virtualmachineosdisk').createOption]"
303     },
304     "imageReference": {
```





# What is Microsoft 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."



Empowering your WVD deployments with Project Bicep!

# Project 'Bicep'

Simple declarative language to provision infrastructure to Azure.

## Intuitive

Easy to read and to author

## Transpiles to ARM Templates

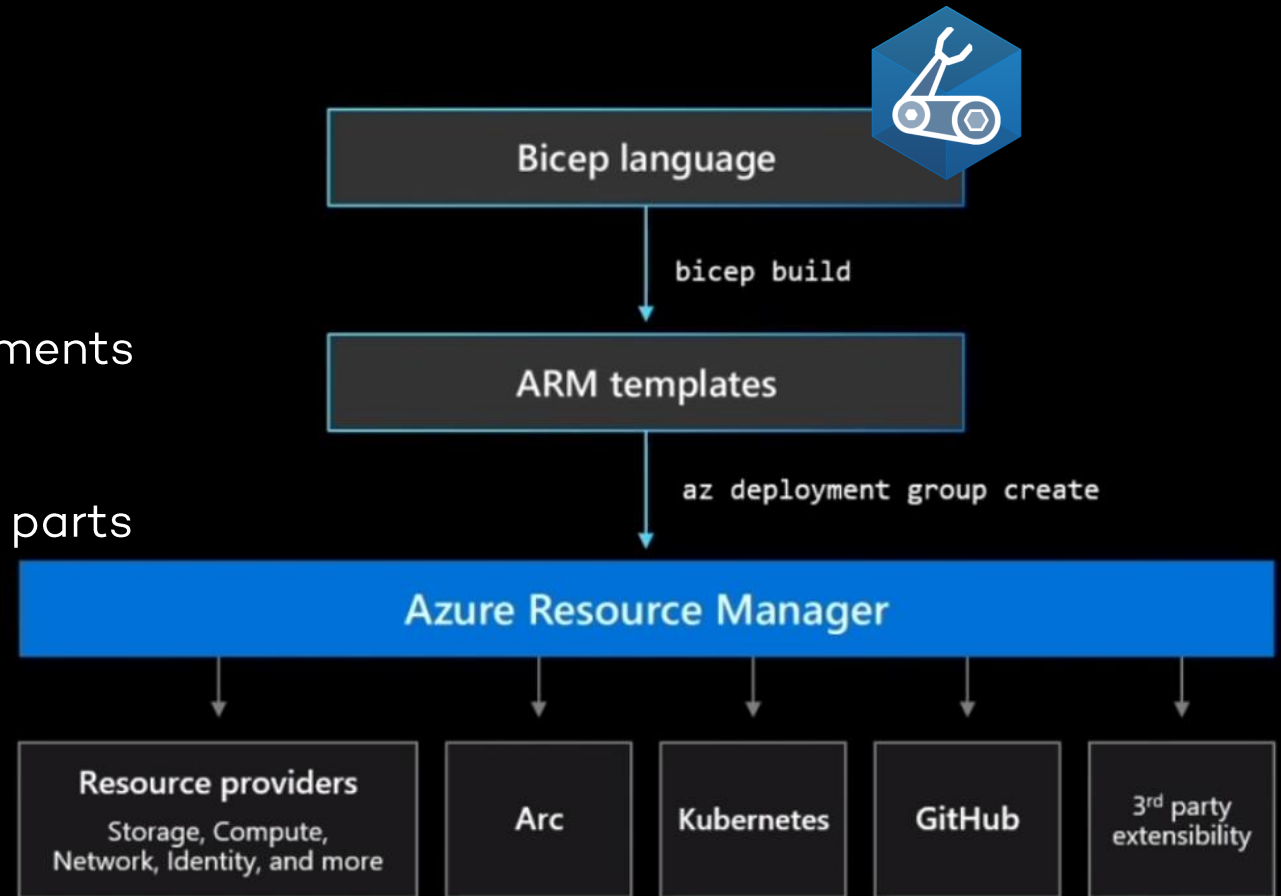
Leverage ARM template knowledge and investments

## Modular

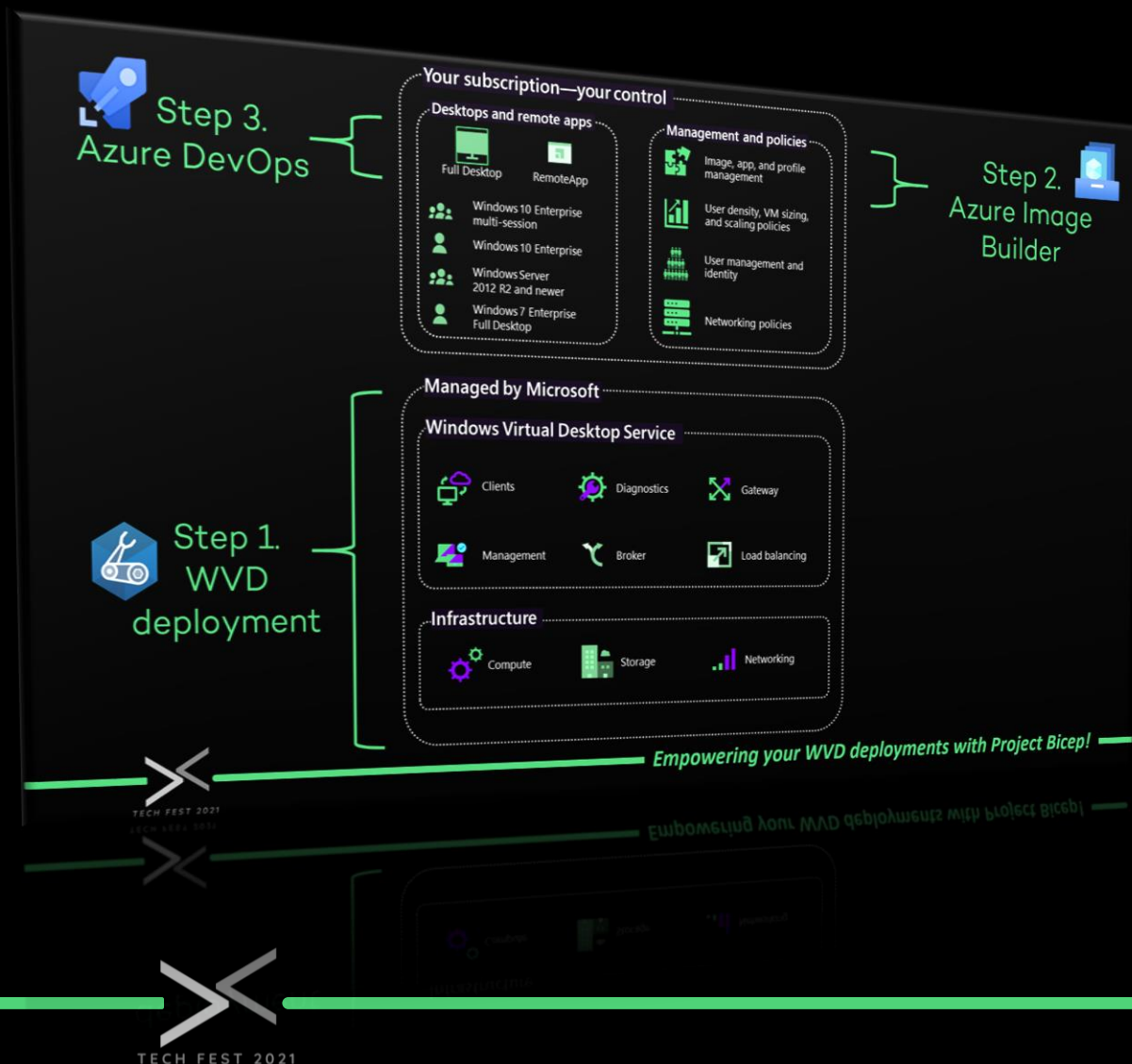
Abstract common blocks of code into reusable parts

## Open Source

Transparency and community



# Demo

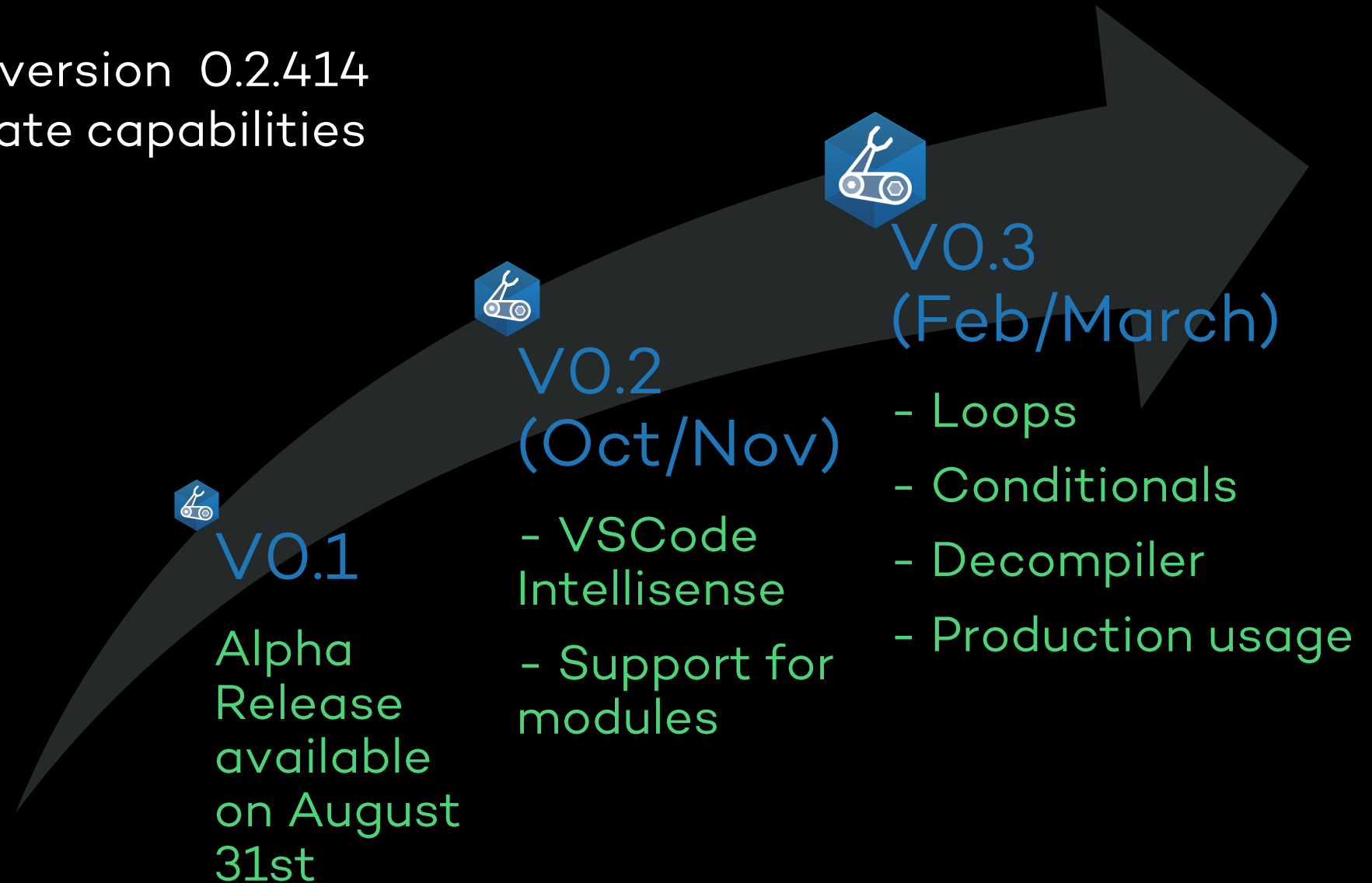


- Vnet with 2 Subnets
- Peering to ADDS vnet
- WVD Workspaces, hostpools and Appgroups
- WVD Template Host VM
- Azure File Services for Profile Management
- Azure Keyvault with secrets
- Azure Log analytics & diagnostics settings
- Full Azure Image Builder setup Preview
- Template Specs Preview



# Road map

Current release: CLI version 0.2.414  
~90% of ARM template capabilities implemented!



# 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. It includes a file browser on the left with folders like .config, github, pipelines, vscode, Formula, docs, scripts, and src. The main content area shows a list of recent commits, including one by anthony-c-martin titled 'Refactor scope naming (#1368)'. The right sidebar contains an 'About' section describing Bicep as a declarative language for describing and deploying Azure resources, a 'Releases' section for v0.2.328 (alpha), and a 'Contributors' section.

This screenshot shows a specific example within the Azure/bicep repository: 'bicep/docs/examples/201/wvd-backplane-with-network-and-storage-and-monitoring/'. It displays a list of files including main.bicep, main.json, and readme.md. Below the file list is a table summarizing the project's components:

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

Below the table, it identifies the project as a 'Multi-module Bicep project for WVD'.

The screenshot shows the Bicep Playground interface with example code for deploying a Windows Virtual Desktop (WVD) environment. The code includes parameters for tenant ID, subscription ID, and resource group, and defines resources for the WVD backplane, network, storage, and monitoring components.



# THANK YOU!



Freek Berson - Wortell  
@fberson  
[github.com/fberson](https://github.com/fberson)  
Microsoft MVP







Thank you for being part of WVD Tech Fest!  
Please evaluate your session below:

