Mastering Bicep Infrastructure as Code for Azure!

Freek Berson
@fberson

Sponsors & organizers

Thanks to sponsors:

veeam

change cockpit

dotCLOUD
keep it simple but significant

ScriptRunner®
Your #1 for PowerShell Automation

365 | THE MICROSOFT BLOG

Thanks to organizers:
Drago Petrovic and Stoyan Chalakov.

About Freek Berson

Focus

End User Computing, Infrastructure as Code

From

The Netherlands

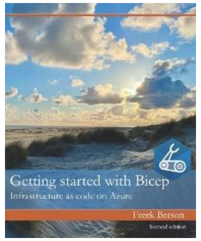
My Blog

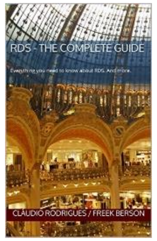
Themicrosoftplatform.net

Certifications

Books

- Getting started with Bicep
- RDS - The Complete Guide





Contact

@fberson

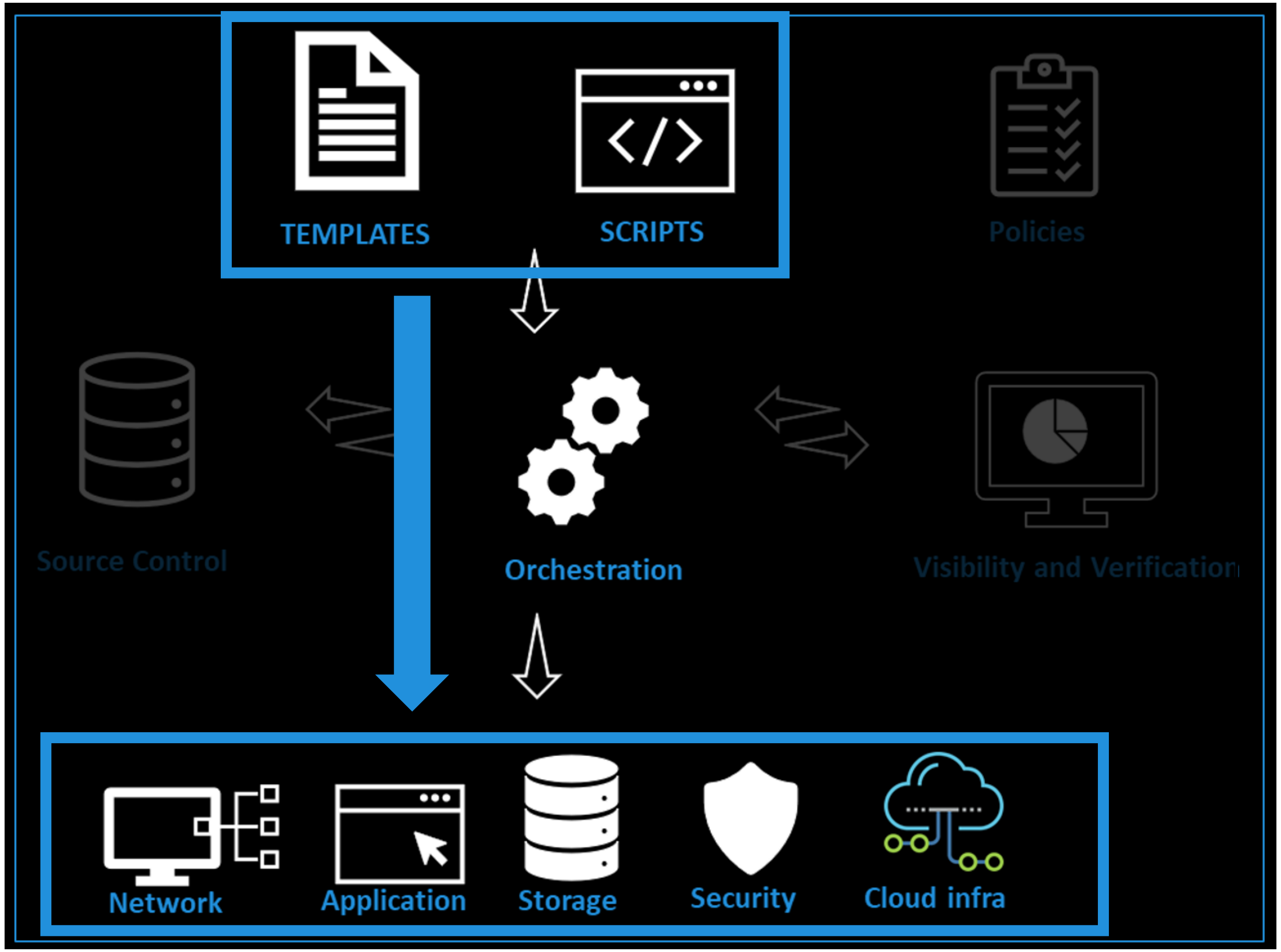
github.com/fberson

Agenda

1. Brief introduction into IaC, ARM, and Bicep
2. Demos, demos, demos ☺
3. Call to action

Infrastructure as Code (IaC)

"..the process of **provisioning infrastructure resources** similar to how software is deployed."



Imperative code vs Declarative code

Imperative code

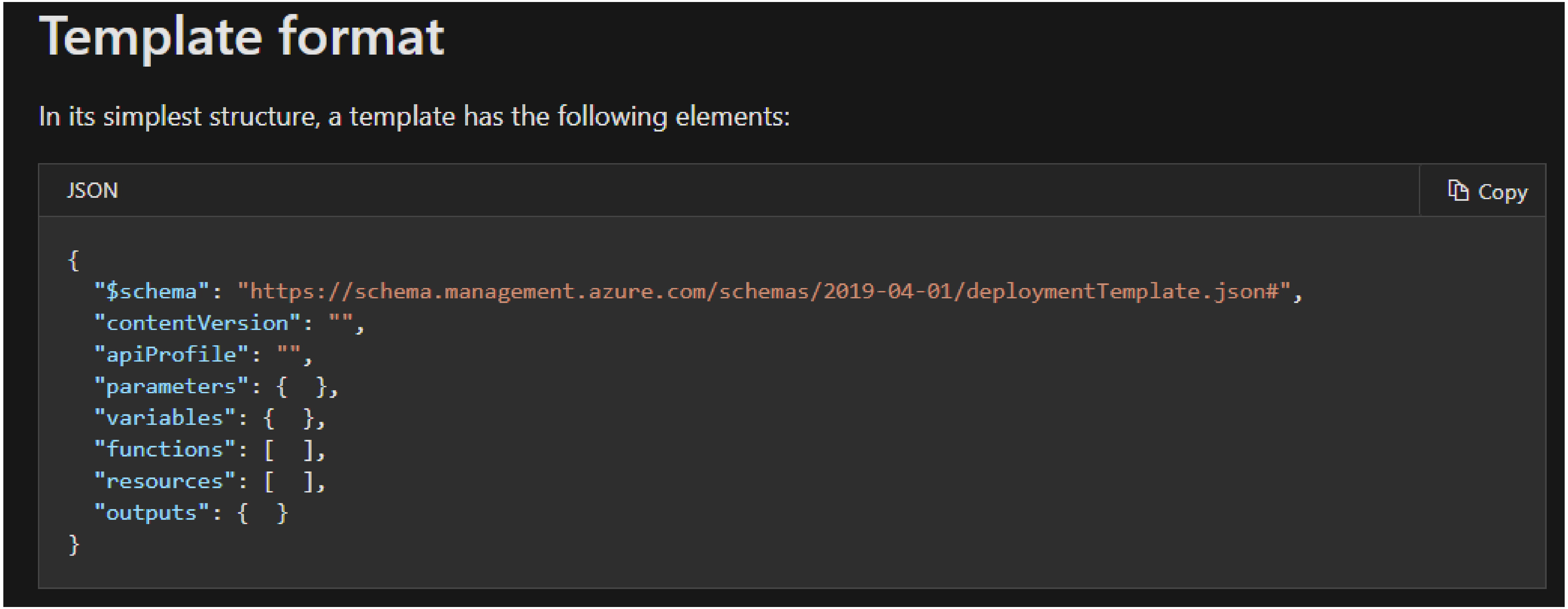
You **execute a sequence of commands**, in a specific order, to reach an end configuration.

This process defines the what and how.

Declarative code

You **specify only the end configuration**. The code doesn't define how to accomplish the task.

ARM Template

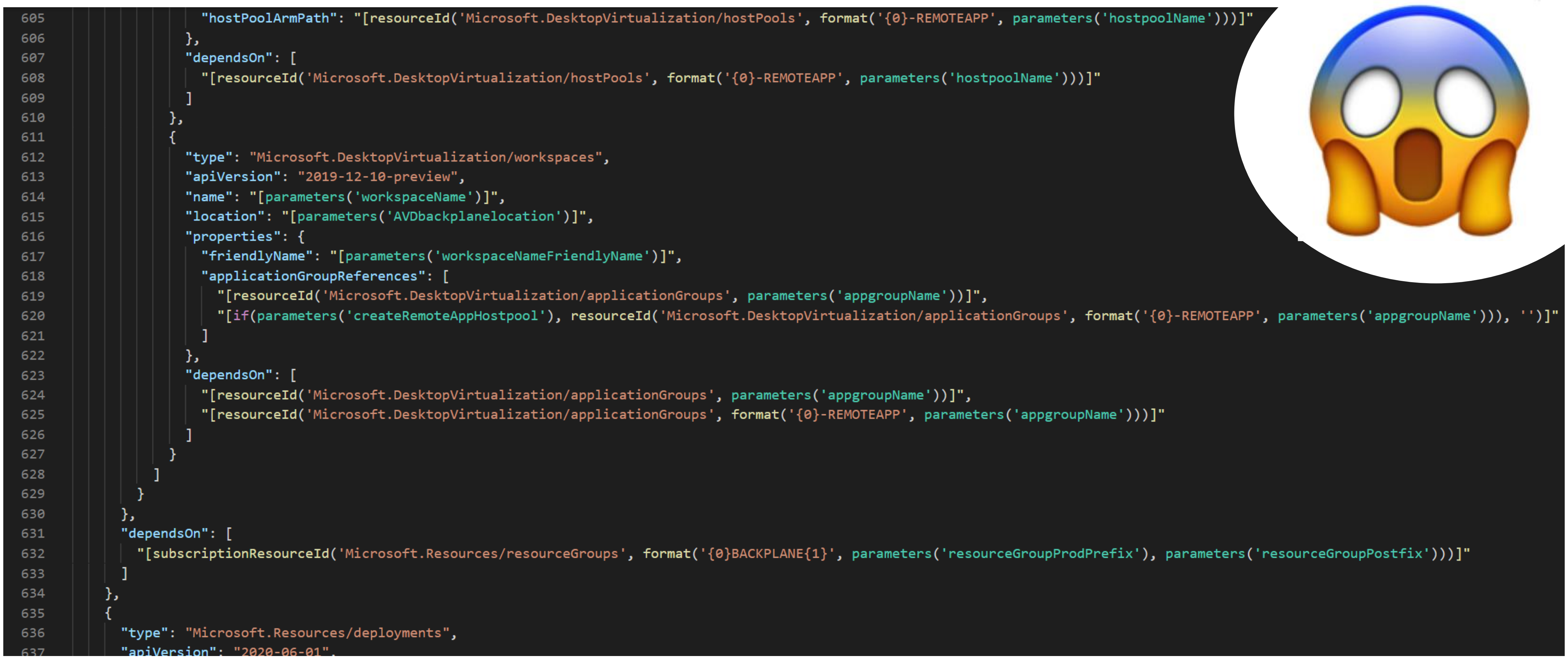


Template format

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

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

ARM Template complexity

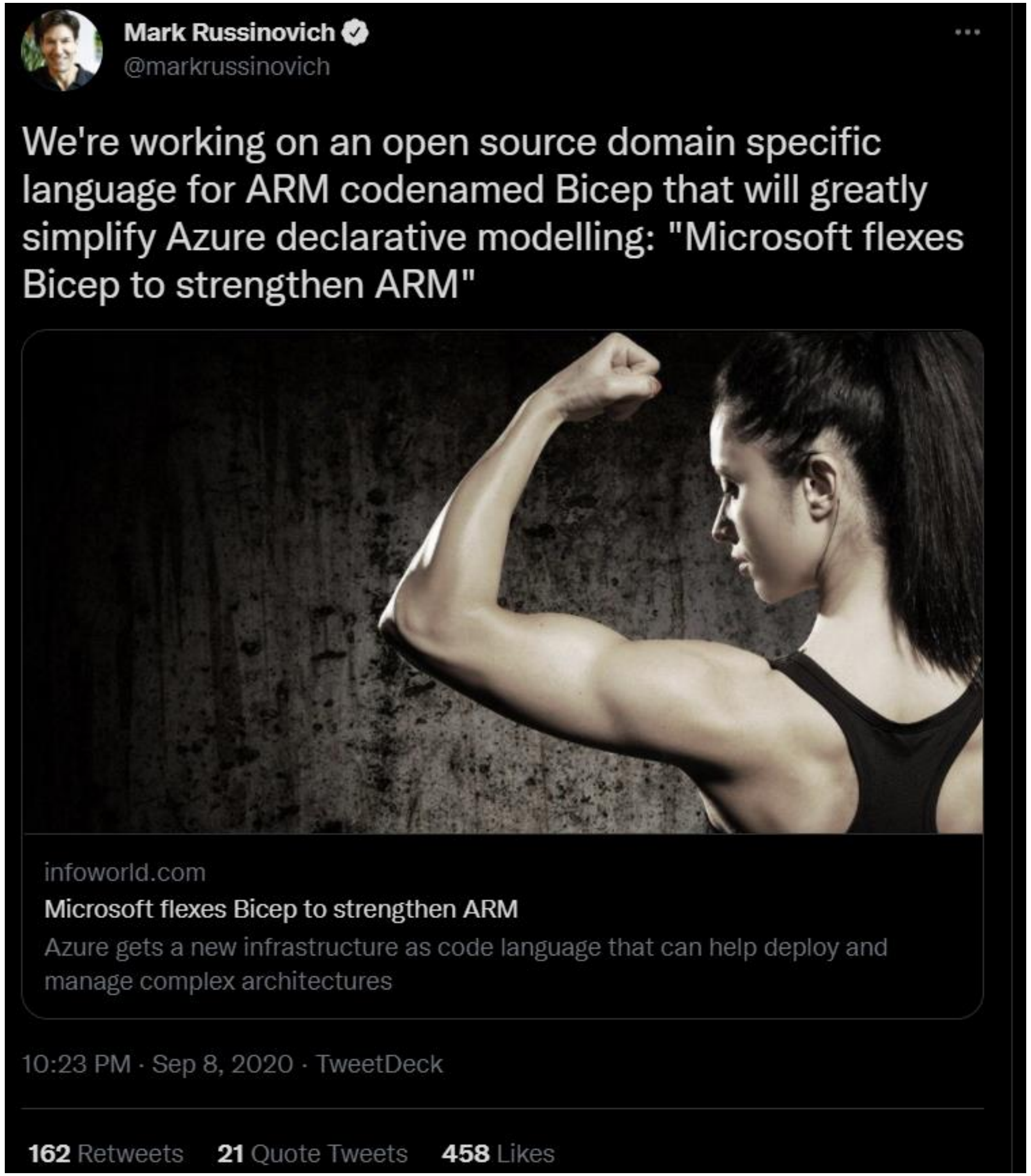


```
                "hostPoolArmPath": "[resourceId('Microsoft.DesktopVirtualization/hostPools', format('{0}-REMOTEAPP', parameters('hostpoolName')))]"
              },
              "dependsOn": [
                "[resourceId('Microsoft.DesktopVirtualization/hostPools', format('{0}-REMOTEAPP', parameters('hostpoolName')))]"
              ]
            },
            {
              "type": "Microsoft.DesktopVirtualization/workspaces",
              "apiVersion": "2019-12-10-preview",
              "name": "[parameters('workspaceName')]",
              "location": "[parameters('AVDbackplanelocation')]",
              "properties": {
                "friendlyName": "[parameters('workspaceNameFriendlyName')]",
                "applicationGroupReferences": [
                  "[resourceId('Microsoft.DesktopVirtualization/applicationGroups', parameters('appgroupName'))]",
                  "[if(parameters('createRemoteAppHostpool'), resourceId('Microsoft.DesktopVirtualization/applicationGroups', format('{0}-REMOTEAPP', parameters('appgroupName'))), '')]"
                ]
              },
              "dependsOn": [
                "[resourceId('Microsoft.DesktopVirtualization/applicationGroups', parameters('appgroupName'))]",
                "[resourceId('Microsoft.DesktopVirtualization/applicationGroups', format('{0}-REMOTEAPP', parameters('appgroupName')))]"
              ]
            }
          ]
        }
      },
      "dependsOn": [
        "[subscriptionResourceId('Microsoft.Resources/resourceGroups', format('{0}BACKPLANE{1}', parameters('resourceGroupProdPrefix'), parameters('resourceGroupPostfix')))]"
      ]
    },
    {
      "type": "Microsoft.Resources/deployments",
      "apiVersion": "2020-06-01",
```

What is 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’



Mark Russinovich
@markrussinovich

We're working on an open source domain specific language for ARM codenamed Bicep that will greatly simplify Azure declarative modelling: "Microsoft flexes Bicep to strengthen ARM"

infoworld.com
Microsoft flexes Bicep to strengthen ARM
Azure gets a new infrastructure as code language that can help deploy and manage complex architectures

10:23 PM · Sep 8, 2020 · TweetDeck

162 Retweets 21 Quote Tweets 458 Likes

Where to position Bicep?

Intuitive

Easy to read and to author

Transpiles to ARM Templates

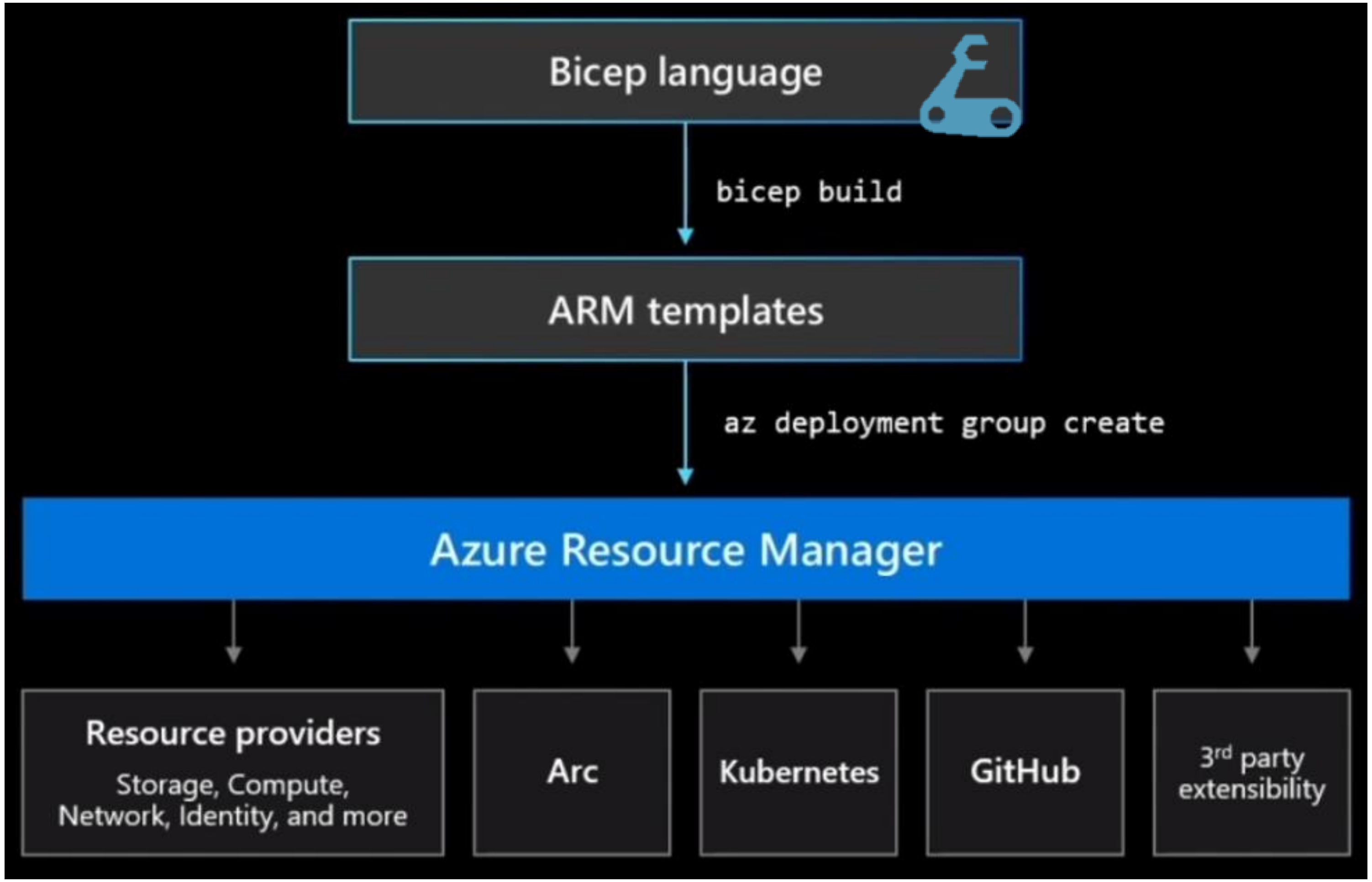
Leverage ARM template knowledge and investments

Modular

Abstract common blocks of code into reusable parts

Open Source

Transparency and community

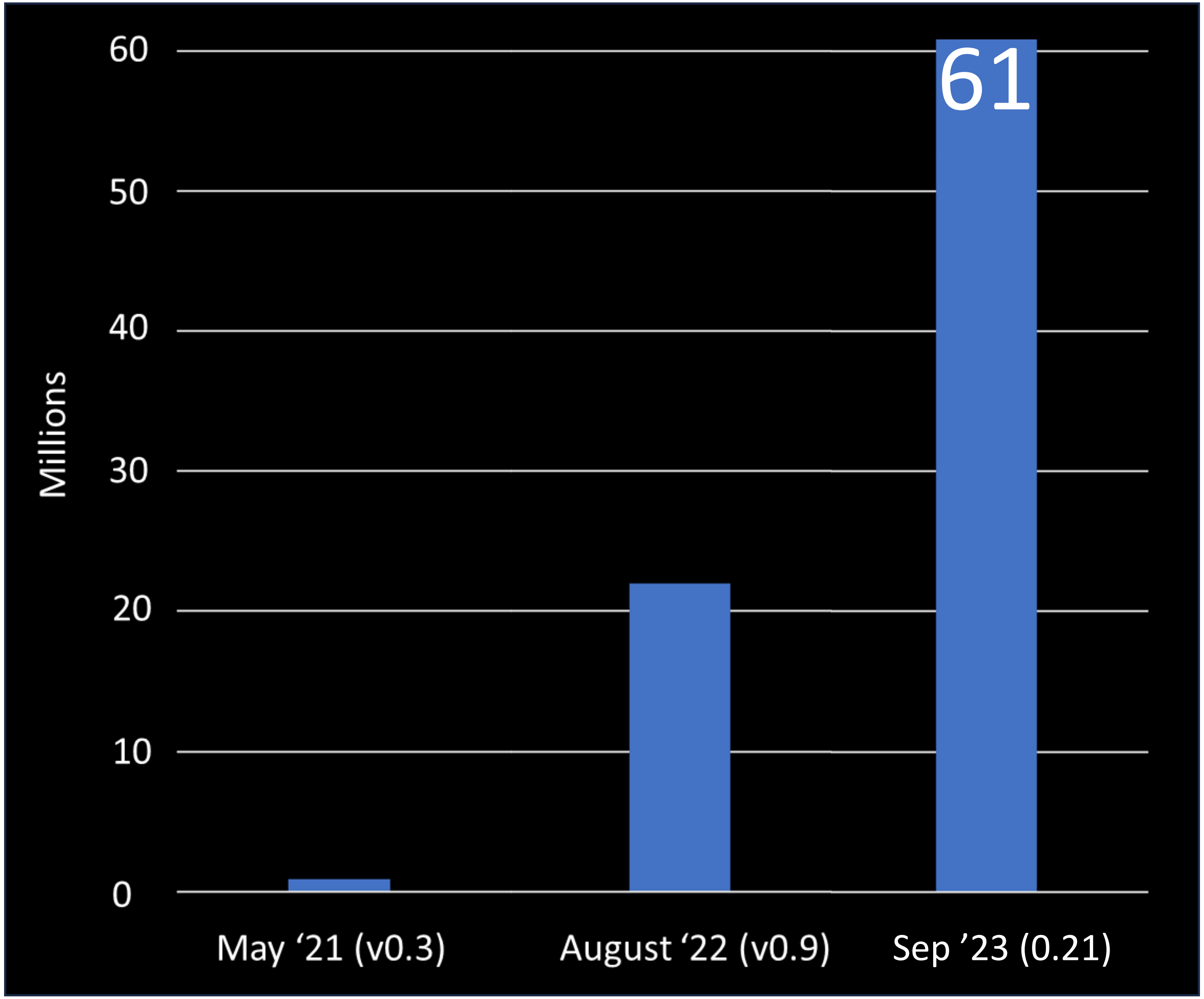


```
bicep build
```

```
az deployment group create
```

How well is Bicep adopted?

Number of resources deployed using Bicep in last 30 days



Demos

1. Creating a Bicep template from scratch
2. Optimize authoring experience with VSCode
3. Modules
4. Large scale use case
5. Deployment stacks

Call to action!

- Bicep MS Docs:
  Aka.ms/bicep
- Bicep Monthly Community call
  surveymonkey.com/r/ARMnews
- Bicep GitHub location
  github.com/Azure/bicep
- Bicep Learning path
  docs.microsoft.com/en-us/azure/azure-resource-manager/bicep/learn-bicep

cloud8

Mastering Bicep – Infrastructure as Code for Azure!

Freek Berson
@fberson