Automating Deployment with Azure Resource Manager

Mark Heath
MICROSOFT MVP

@mark_heath www.markheath.net

Overview

Automating app deployment

- Reliable and repeatable
- Production and test deployments

Automating app upgrade

- Azure functions code
- Static web content

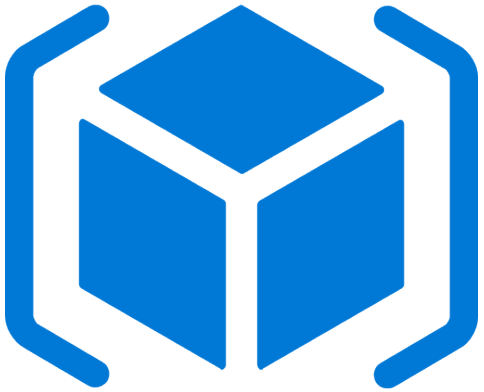
Azure Resource Manager templates

- Define resources in JSON
- Automate with PowerShell

Deploy functions code from Git

Update blob storage with AzCopy

Azure Resource Manager (ARM)



Automate deployment of app resources

- e.g. Storage Accounts, Function Apps

Organize resources into Resource Groups

- Clean up easily by deleting the group

Control access to resources

- Protect the production environment

Define the infrastructure in JSON

- ARM Template

ARM Templates

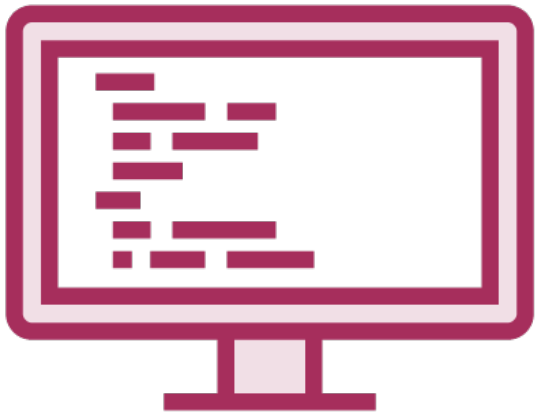
Define resources

Define configuration and settings

Parameters and variables

- Deploy multiple versions of your template

Creating ARM Templates



Generate Templates from the Azure Portal

- Includes sample deployment code

Create Templates in Visual Studio

- Needs the Azure SDK
- Create a Deployment Project

resources.azure.com

- See the full ARM template for your resources

Azure quick-start templates

- https://github.com/Azure/azure-quickstart-templates

Updating Application Code

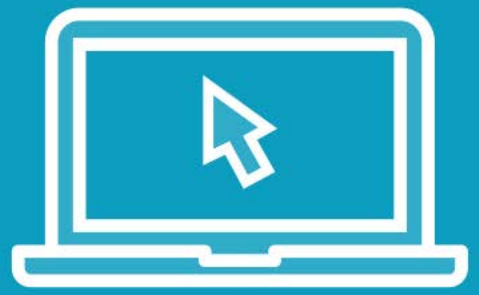
Blob storage container contents

- Upload manually with Azure Storage Explorer
- AzCopy command line utility

Azure Functions code

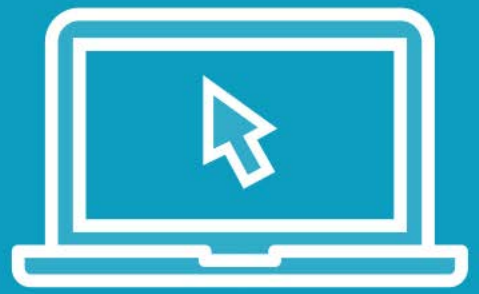
- MsDeploy, FTP, Kudu
- Git repository
- Configure in ARM template

Demo



Deploy an Azure Functions app with an ARM Template

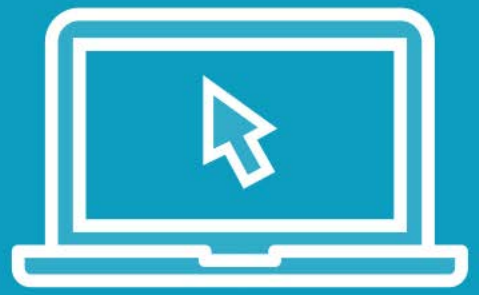
Demo



Update function source code with Git

- Push commit to GitHub
- Sync from GitHub

Demo



Update blob container contents with AzCopy

Summary

Azure Resource Manager

ARM templates

- Describe infrastructure
- Deploy with PowerShell
- Can be parameterized

Automated updates

- Azure Function Apps Git synchronization
- AzCopy for blob storage containers

Deployment environments

- Production and test

Monitoring applications