Microsoft

Azure DevOps Pipelines to Git Hub Actions

Matt Ruma
Cloud Solution Architect
matt.ruma@microsoft.com
www.mattruma.com

Overview

MJR068/DEMO.md at master · mattruma/MJR068 (github.com)
Script

mattruma/MJR068 (github.com) GitHub Repository

Summary - Overview (azure.com) DevOps

Azure DevOps

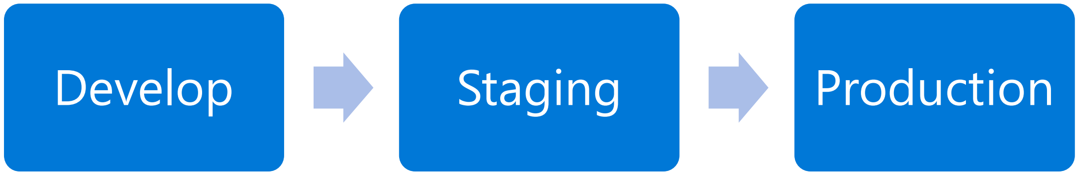
- Define pipelines using the Classic interface
- Define pipelines using YAML syntax

GitHub Actions

© Microsoft Corporation

App Architecture

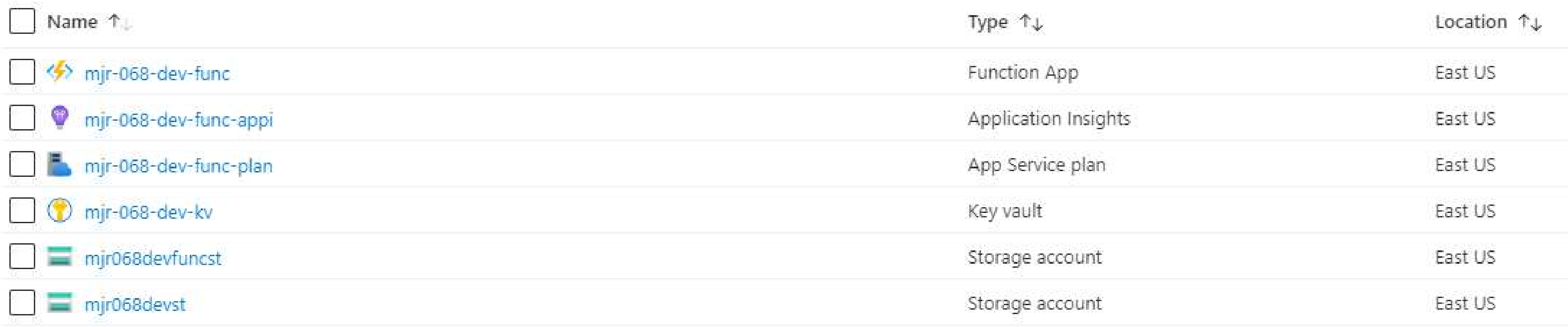
- **(2) Function Apps**
  - Timer Trigger runs every day and writes game collection from https://bgg-json.azurewebsites.net/collection/raz0rf1sh to Azure Blob Storage.
  - Http Trigger at https://mjr-068-dev-func.azurewebsites.net/api/GameListHttpTrigger returns a list of games.
- **Deployment supports (3) environments, code is promoted to each environment.**
  - Staging and Production require an approval to deploy.
  - Each environment include infrastructure deployment, code deployment and Postman integration tests.
  - Tried to have each pipeline (AzDO Classic and YAML and GitHub Actions) follow this pattern.



Develop → Staging → Production

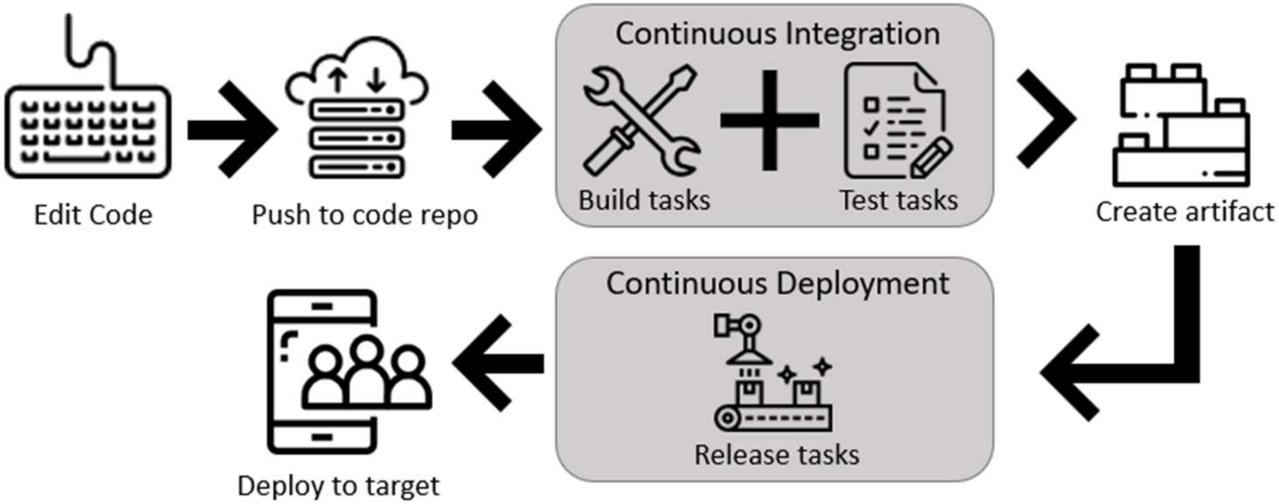
Links

- Develop [mjr-068-dev-rg - Microsoft Azure]
- Staging [mjr-068-stg-rg - Microsoft Azure]
- Production [mjr-068-prd-rg - Microsoft Azure]



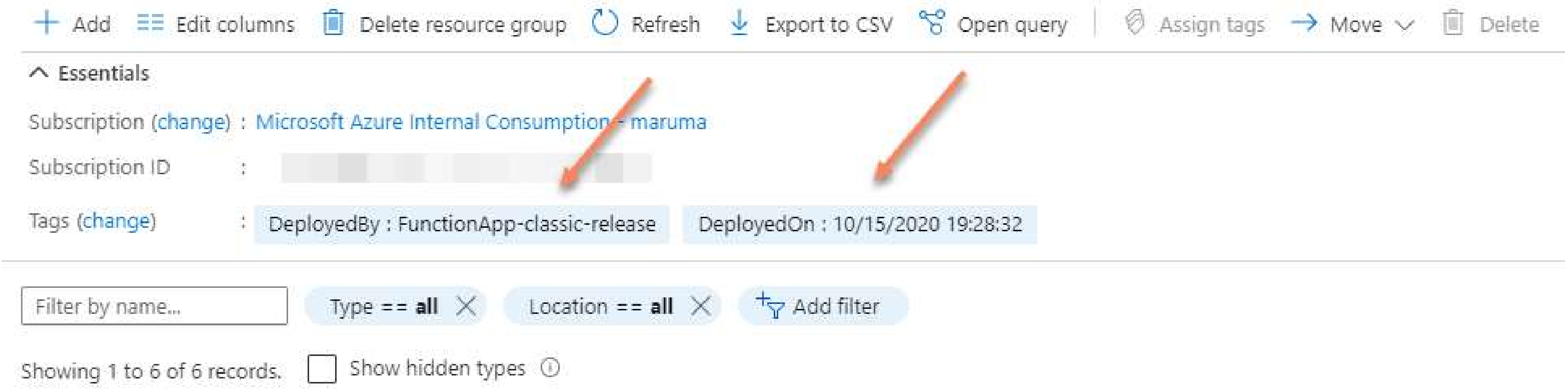
□ Name ↑↓	Type ↑↓	Location ↑↓
□ mjr-068-dev-func	Function App	East US
□ mjr-068-dev-func-appi	Application Insights	East US
□ mjr-068-dev-func-plan	App Service plan	East US
□ mjr-068-dev-kv	Key vault	East US
□ mjr068devfuncst	Storage account	East US
□ mjr068devst	Storage account	East US

Define pipelines using the Classic interface



Classic build/release

- **Build** FunctionApp-classic-build - Azure DevOps Services
  - Copy deployment files copies PowerShell scripts and ARM template.
  - Copy integration tests files copies Postman files and supporting PowerShell scripts.
  - Currently CI is turned off.
- **Release** FunctionApp-classic-release - Pipelines (azure.com)
  - Develop FunctionApp-classic-release - Pipelines (azure.com)
- **Tag the resource group with FunctionApp-classic-release and current date.**



Subscription (change) : Microsoft Azure Internal Consumption - maruma

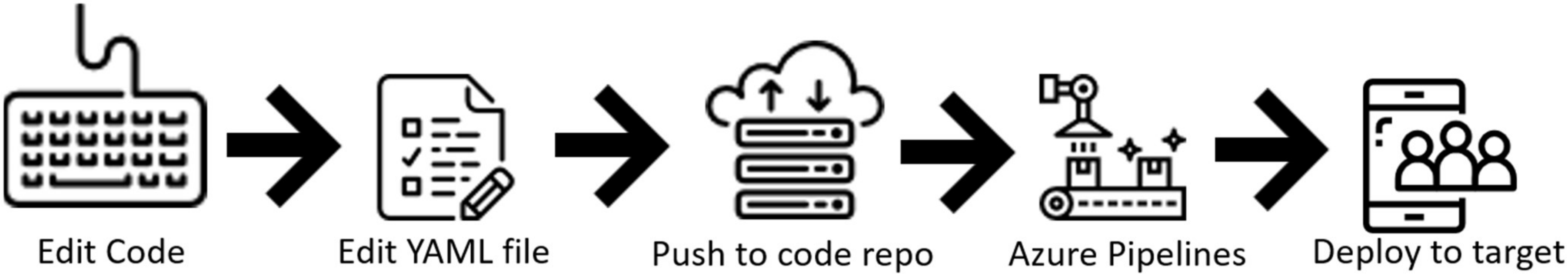
Subscription ID : [redacted]

Tags (change) : DeployedBy : FunctionApp-classic-release DeployedOn : 10/15/2020 19:28:32

Filter by name... Type == all Location == all Add filter

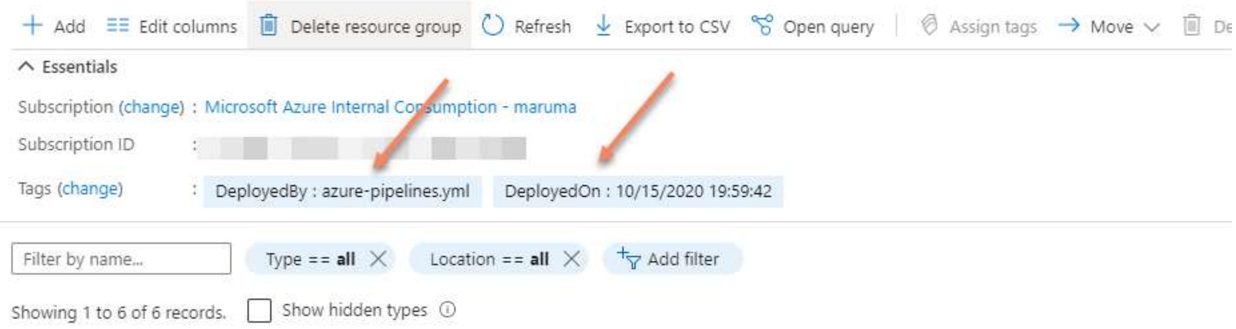
Showing 1 to 6 of 6 records. □ Show hidden types

Define pipelines using YAML syntax



YAML pipeline

- **Environments** [Environments - Pipelines (azure.com)]
  - For the approval process.
- **Variable Groups** [Library - Pipelines (azure.com)]
  - For variables in Key Vault, could have used these in Classic Pipelines, as well.
- **Pipelines** [Pipelines - Run 20201015.1 (azure.com)]
  - Streamline with templates, a lot of redundancy.
  - CI is turned off by setting branch to none.



Subscription (change) : Microsoft Azure Internal Consumption - maruma

Subscription ID : [redacted]

Tags (change) : DeployedBy : azure-pipelines.yml DeployedOn : 10/15/2020 19:59:42

Filter by name...

Type == all

Location == all

Showing 1 to 6 of 6 records. □ Show hidden types

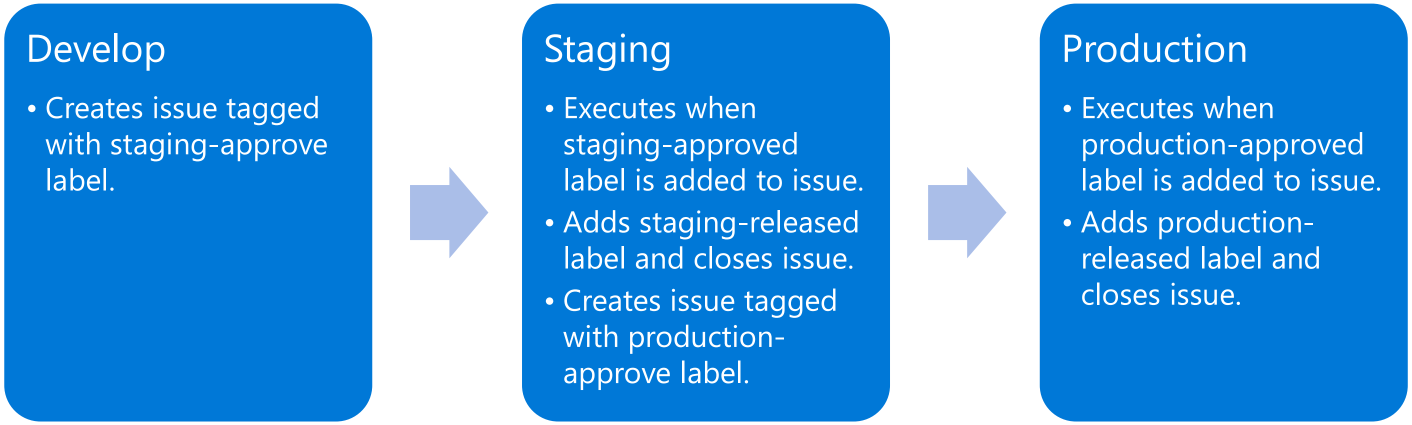
GitHub actions

- **Automate, customize, and execute your software development workflows right in your repository with GitHub Actions.**
- **You can discover, create, and share actions to perform any job you'd like, including CI/CD, and combine actions in a completely customized workflow.**
- **The GitHub Actions workflow = AzDO YAML pipeline**
  - Both use YAML syntax
  - AzDO YAML supports stages, jobs and steps.
  - GitHub Actions workflow YAML supports jobs and steps.
  - No built-in approval process, like AzDO for promoting code to different environments.
- **Nowhere near as streamlined as I would like it, more of a brute force conversion.**

GitHub actions

- **Secrets** Secrets (github.com)
  - Some will still come from Key Vault, this is what is needed to execute the GitHub actions workflows.
- **Workflows** MJR068/.github at master · mattruma/MJR068
  - Store in .github folder.
  - Workflow for each environment.
- **Custom Actions** MJR068/.github/actions at master · mattruma/MJR068
  - Created custom actions for my approval workflow.
    - add-label-to-issue MJR068/.github/actions/add-label-to-issue at master · mattruma/MJR068
    - check-issue MJR068/.github/actions/check-issue at master · mattruma/MJR068
    - close-issue MJR068/.github/actions/close-issue at master · mattruma/MJR068
    - create-issue https://github.com/mattruma/MJR068/tree/master/.github/actions/create-issue
  - Similar custom actions were available online but wanted experience on how to do it myself.
  - Consolidate into two actions, create and update.

Makeshift GitHub actions approval process



Develop

- Creates issue tagged with staging-approve label.

Staging

- Executes when staging-approved label is added to issue.
- Adds staging-released label and closes issue.
- Creates issue tagged with production-approve label.

Production

- Executes when production-approved label is added to issue.
- Adds production-released label and closes issue.

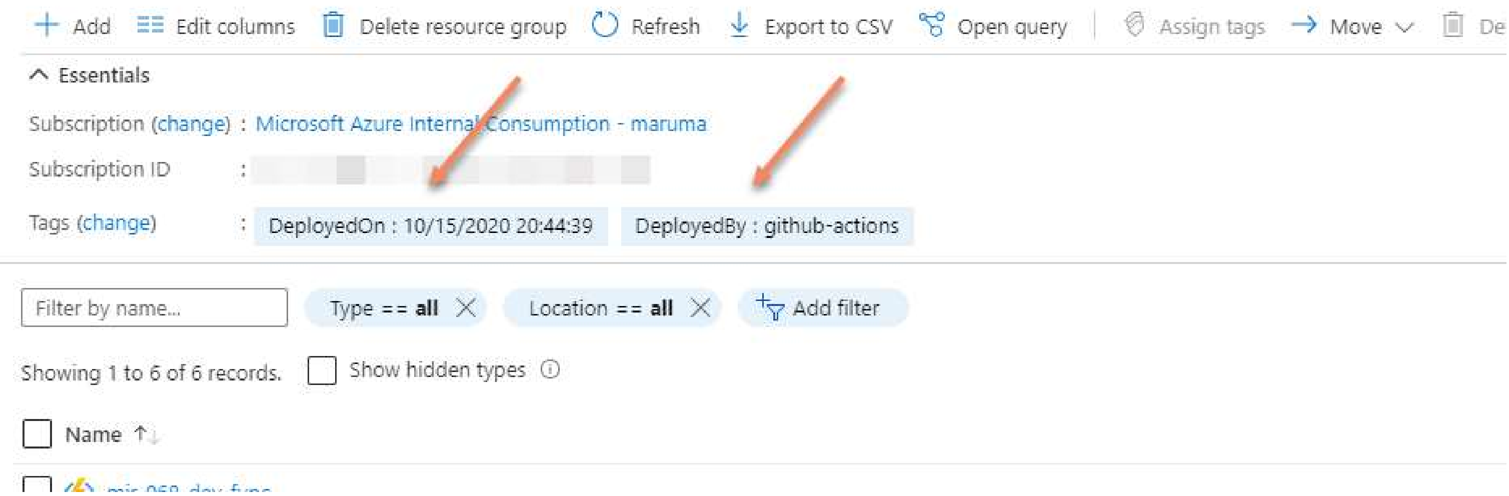
- **In one way, robust, for how easy it is to create a Custom Action.**
- **Not very streamlined, lots of opportunity for missed deployments.**
- **Better approval process hopefully coming soon.**

GitHub actions workflow

- **See MJR068/develop.yml at master · mattruma/MJR068 (github.com)**
- **Set to allow for manually trigger.**
- **Service Principal variable is empty, gets populated in script.**
  - I would probably lean towards making it a parameter that is passed in.
- **Lot of "hard-coded" parameters in here, desperately needs some streamlining.**
- **The Azure/login@v1 action takes forever.**
  - Issue tracked at Login consistently takes 30s-60s to complete · Issue #20 · Azure/login (github.com).
  - Once that is fixed, will be comparable to the speed of the AzDO Classic and YAML pipelines.
- **Custom actions Creating a JavaScript action - GitHub Docs**
  - require('@actions/core');
  - require('@actions/github');
  - new github.GitHub(core.getInput('token'));

GitHub Actions workflow

- Develop GitHub actions workflow [Actions · mattruma/MJR068 (github.com)]
- Develop run [github actions · mattruma/MJR068@ba6ee3b]
- Issues [Issues · mattruma/MJR068 (github.com)]



Subscription (change) : Microsoft Azure Internal Consumption - maruma

Subscription ID : [redacted]

Tags (change) : DeployedOn : 10/15/2020 20:44:39 DeployedBy : github-actions

Showing 1 to 6 of 6 records. □ Show hidden types

□ Name ↑↓

□ mjr-068-dev-func

Closing thoughts

- **AzDO the more mature product.**
- **Github actions moving fast to catch up with AzDO.**
  - Missing approval/gate process.
  - Lacks the granular permissions that AzDO gives at least in the cloud offering, maybe available in GitHub Enterprise, not sure.
  - Missing stages, just supports jobs and steps.

> We spent some time deliberating about it and I think the biggest conclusion we came to is we can’t effectively run 2 products having this sort of internal competition between two things.
>
> So we are going to move towards having one in the end.
>
> — Sasha Rosenbaum, Senior Program Manager, Azure DevOps

Thank You

ευχαριστώ Salamat Po متشكرم شكراً Grazie

благодаря ありがとうございます Kiitos Teşekkürler 谢谢

ขอบคุณครับ Obrigado شکریہ Terima Kasih Dziękuję

Hvala Köszönöm Tak Dank u wel дякую Tack

Mulţumesc спасибо Danke Cám ơn Gracias

多謝晒 Ďakujem תודה நன்றி Děkuji 감사합니다

© Microsoft Corporation

Microsoft Azure

© Copyright Microsoft Corporation. All rights reserved.