DevOps

Hybrid Integration Enablement

October 2017

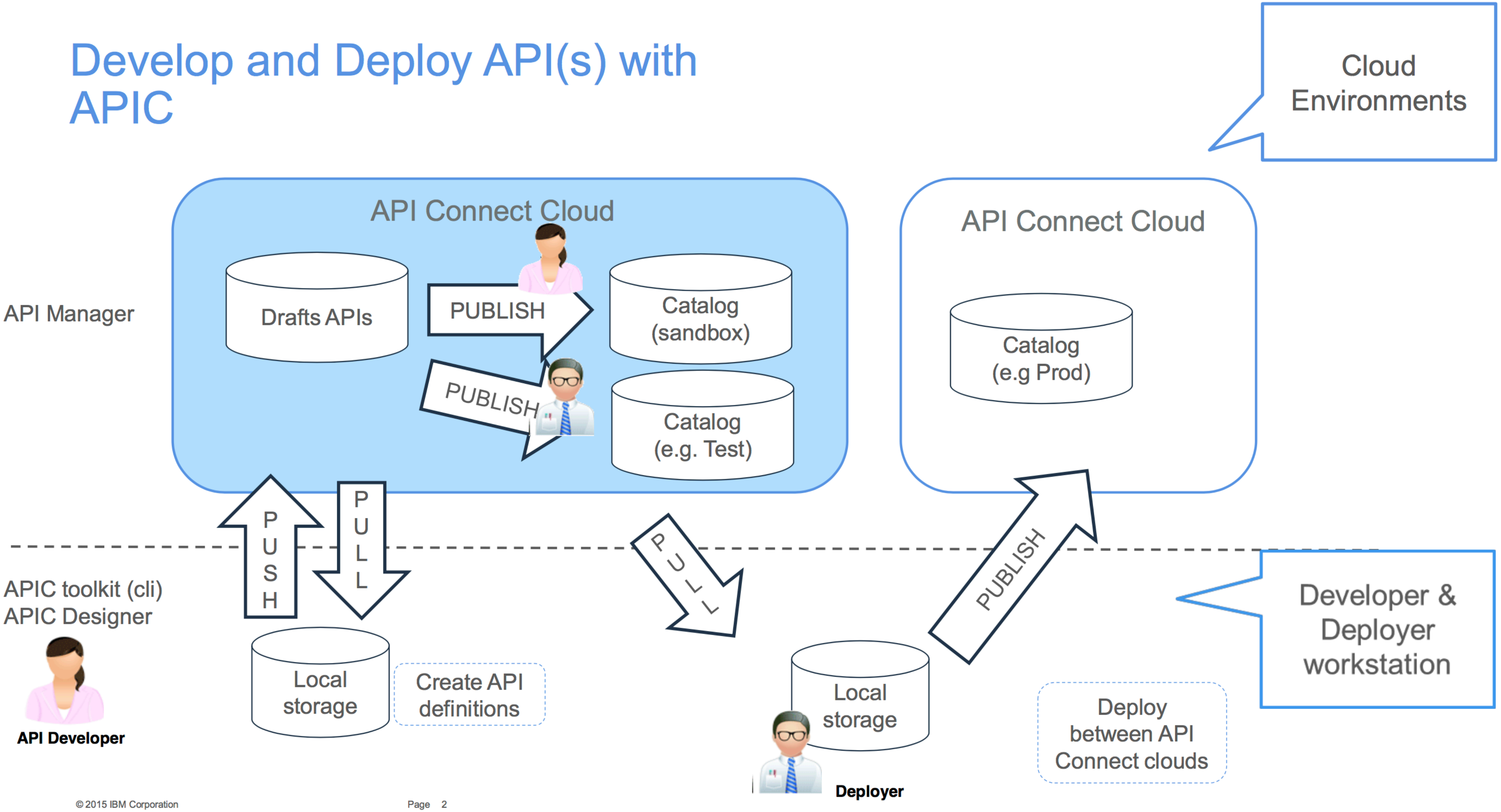
DEVOPS

Use Case: How do I integrate Continuous Integration with API Connect? Does it work with Jenkins?

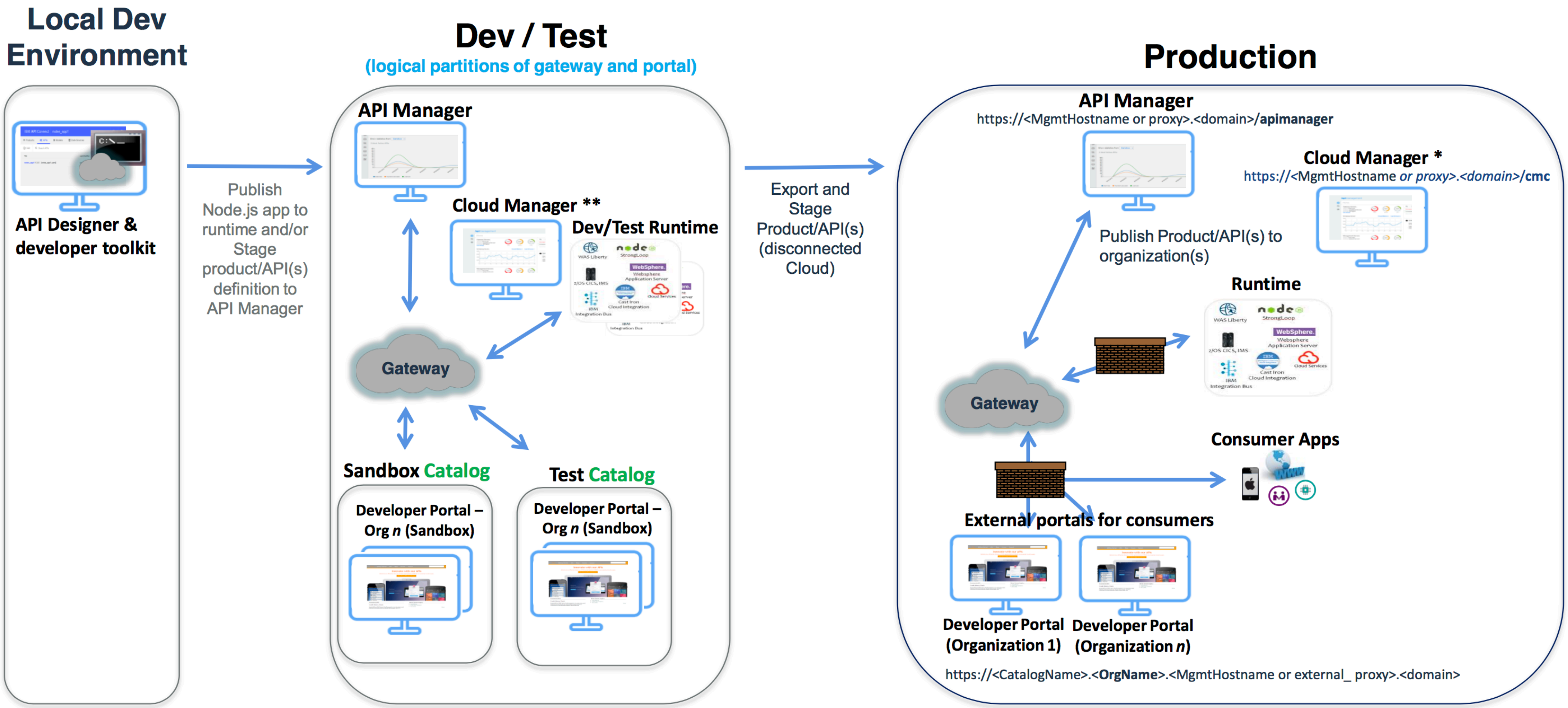
Use Case: What are strategies to maintain one source code folder for my environments?

Deployment Overview

Develop and Deploy API(s) with APIC



Logical Architecture



Local Dev Environment

API Designer & developer toolkit

Publish Node.js app to runtime and/or Stage product/API(s) definition to API Manager

Dev / Test

(logical partitions of gateway and portal)

API Manager

Cloud Manager **

Dev/Test Runtime

Gateway

Sandbox Catalog

Developer Portal – Org *n* (Sandbox)

Test Catalog

Developer Portal – Org *n* (Sandbox)

Export and Stage Product/API(s) (disconnected Cloud)

Production

API Manager

https://<MgmtHostname or proxy>.<domain>/**apimanager**

Cloud Manager *

https://<MgmtHostname *or proxy>.<domain>*/**cmc**

Publish Product/API(s) to organization(s)

Runtime

Gateway

Consumer Apps

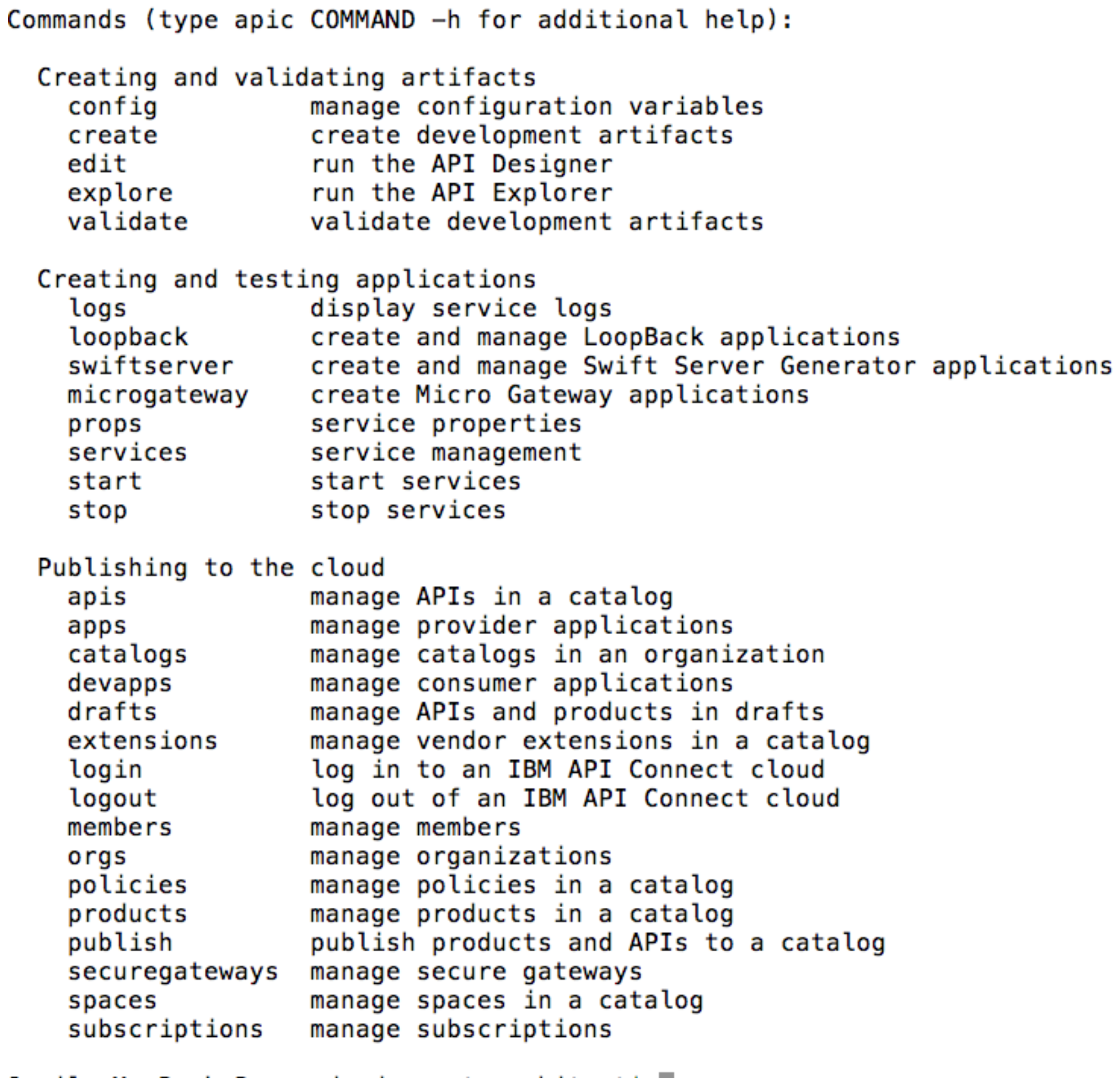
External portals for consumers

Developer Portal (Organization 1)

Developer Portal (Organization *n*)

https://<CatalogName>.<**OrgName**>.<MgmtHostname or external_ proxy>.<domain>

Remember the CLI

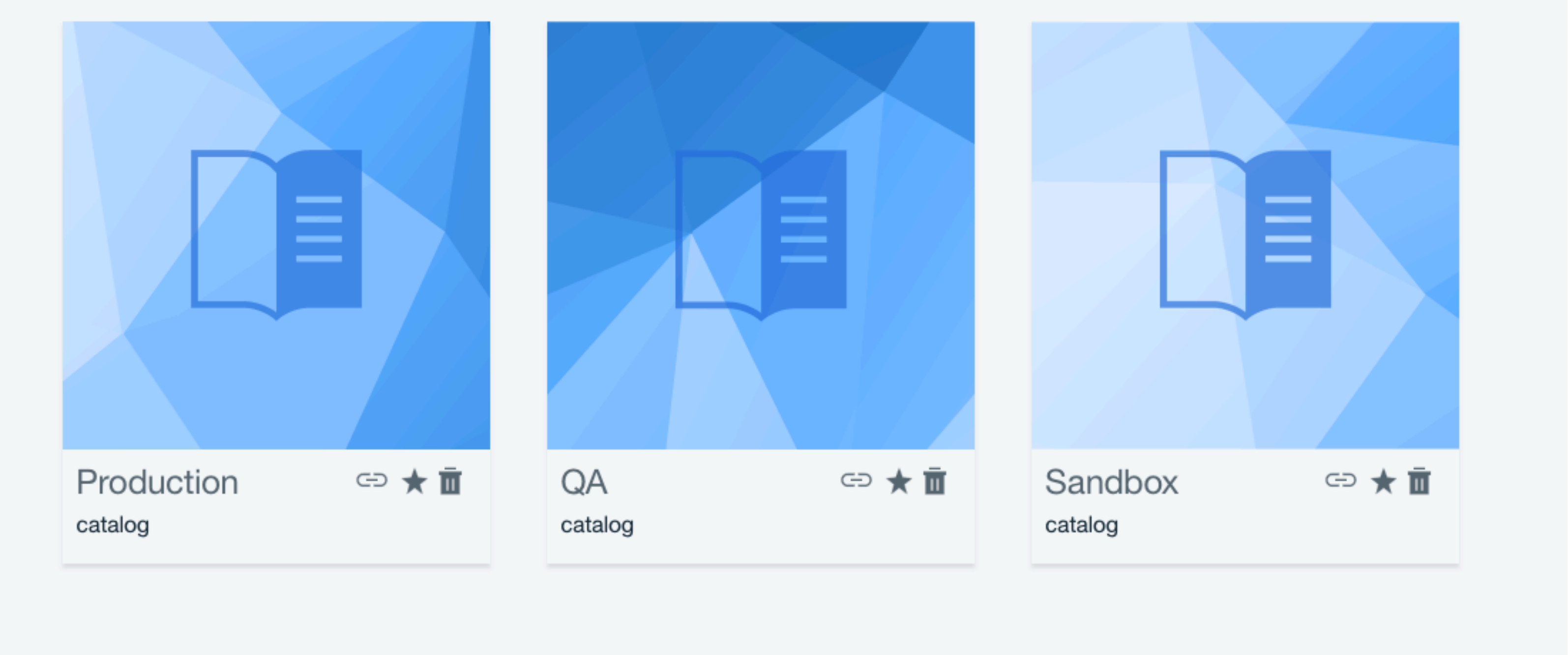


```
Commands (type apic COMMAND -h for additional help):

  Creating and validating artifacts
    config          manage configuration variables
    create          create development artifacts
    edit            run the API Designer
    explore         run the API Explorer
    validate        validate development artifacts

  Creating and testing applications
    logs            display service logs
    loopback        create and manage LoopBack applications
    swiftserver     create and manage Swift Server Generator applications
    microgateway    create Micro Gateway applications
    props           service properties
    services        service management
    start           start services
    stop            stop services

  Publishing to the cloud
    apis            manage APIs in a catalog
    apps            manage provider applications
    catalogs        manage catalogs in an organization
    devapps         manage consumer applications
    drafts          manage APIs and products in drafts
    extensions      manage vendor extensions in a catalog
    login           log in to an IBM API Connect cloud
    logout          log out of an IBM API Connect cloud
    members         manage members
    orgs            manage organizations
    policies        manage policies in a catalog
    products        manage products in a catalog
    publish         publish products and APIs to a catalog
    securegateways  manage secure gateways
    spaces          manage spaces in a catalog
    subscriptions   manage subscriptions
```

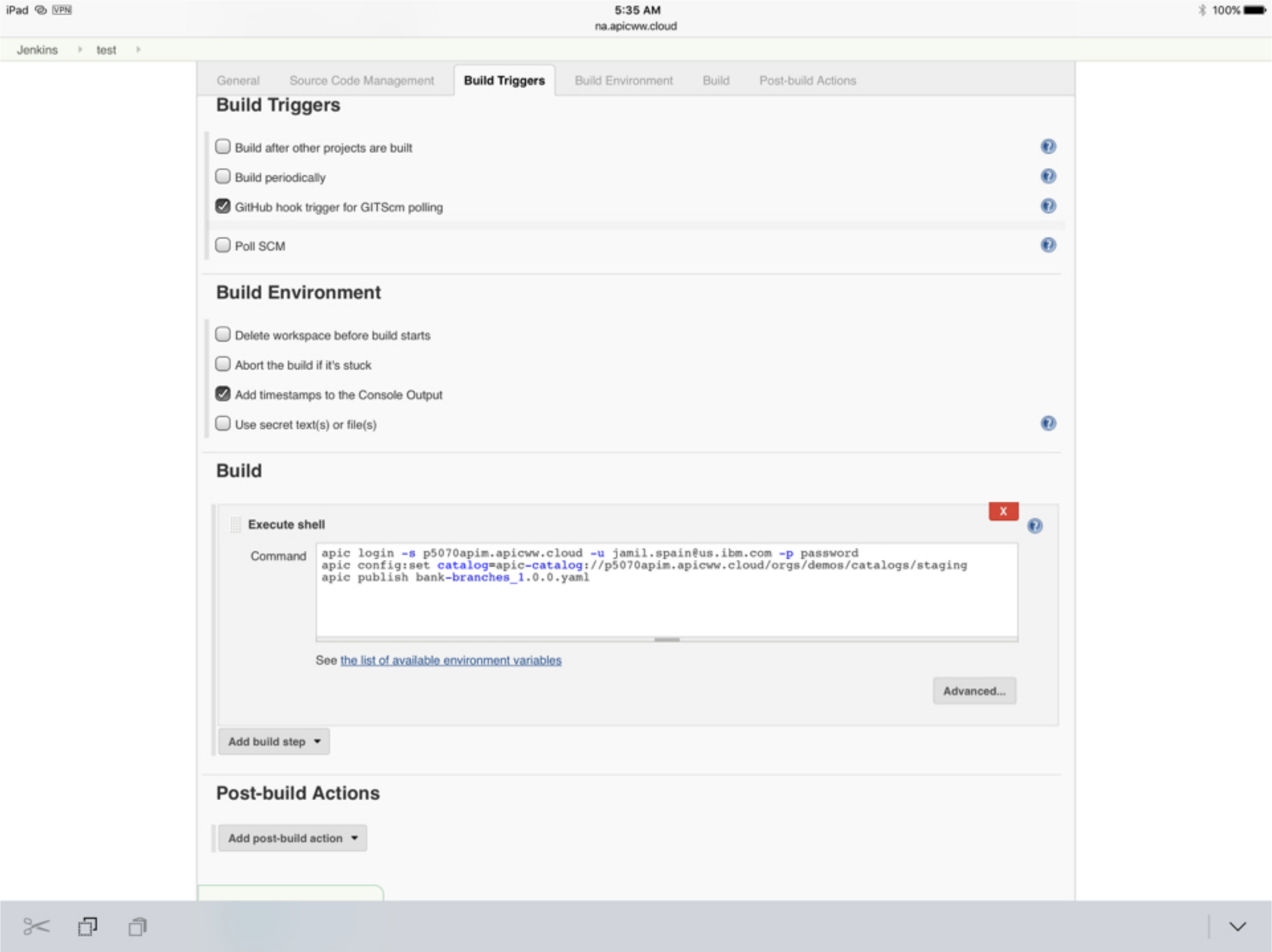
- Built – in Help files are available for the CLI
  - Be sure to learn more about what each command does
  - Valuable commands are here for automation
  - We are technology agnostic, but this component helps with integration (scripting)

Solving Our Use Case

- Scripts can publish Product / API between Catalogs
- Remember the API definitions to declare properties by catalog
- 3 – 4 steps to publish:
  - Logging in
  - Set Catalog
  - Publish API (validate optional)



Jenkins Excerpt



Build Triggers

- □ Build after other projects are built
- □ Build periodically
- ☑ GitHub hook trigger for GITScm polling
- □ Poll SCM

Build Environment

- □ Delete workspace before build starts
- □ Abort the build if it's stuck
- ☑ Add timestamps to the Console Output
- □ Use secret text(s) or file(s)

Build

Execute shell

Command

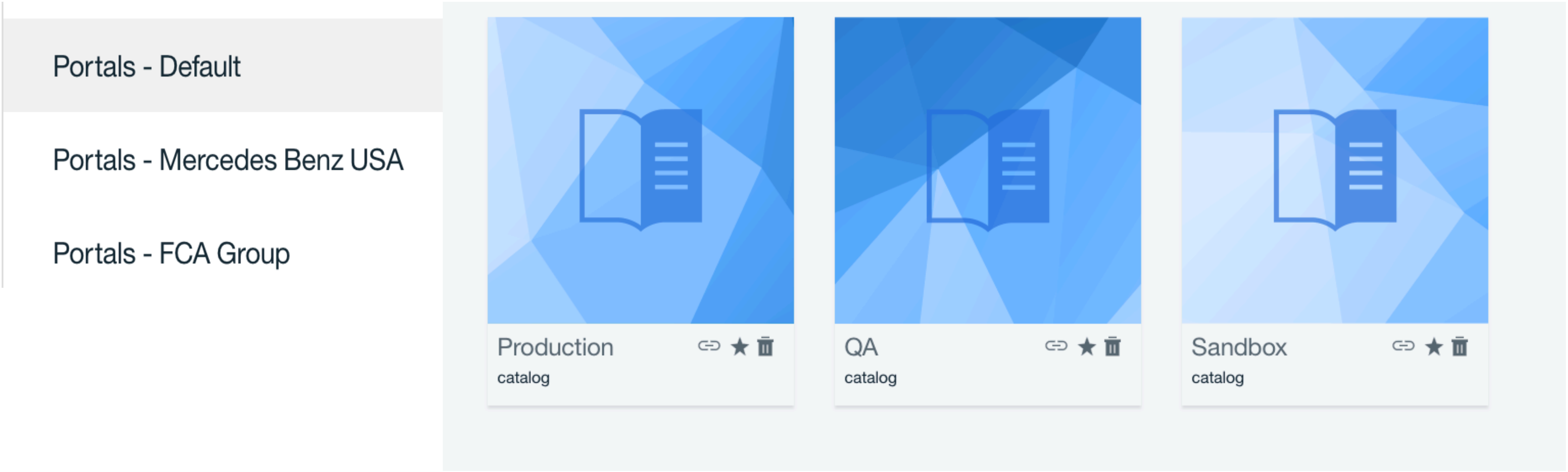
```
apic login -s p5070apim.apicww.cloud -u jamil.spain@us.ibm.com -p password
apic config:set catalog=apic-catalog://p5070apim.apicww.cloud/orgs/demos/catalogs/staging
apic publish bank-branches_1.0.0.yaml
```

See the list of available environment variables

Post-build Actions

API Developer Workflow

- API Designer uses the CLI to make this process easier.
- To move from one Catalog to another, you can create connections to each catalog and publish
- Spaces show up just like singular catalogs too, process is the same.



Portals - Default

Portals - Mercedes Benz USA

Portals - FCA Group

Production
catalog

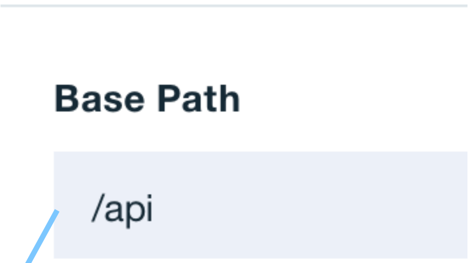
QA
catalog

Sandbox
catalog

Understanding Versioning

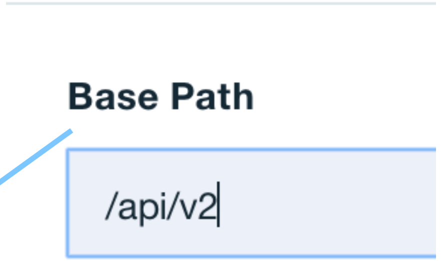
- When we version Products, we upgrade the version of Product & API.
- This works great if it’s same API version going in new Product, but for may other use cases we need a solution.

- API Defined Base Path can provide some context without forcing paths to take the on the obligation. i.e.
  - /v1/branches
  - /v2/branches



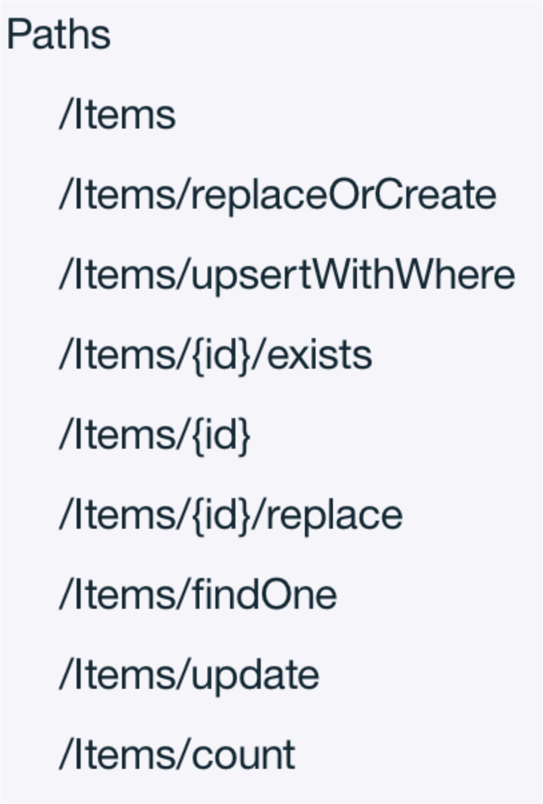
Base Path

/api



Base Path

/api/v2



Paths

/Items

/Items/replaceOrCreate

/Items/upsertWithWhere

/Items/{id}/exists

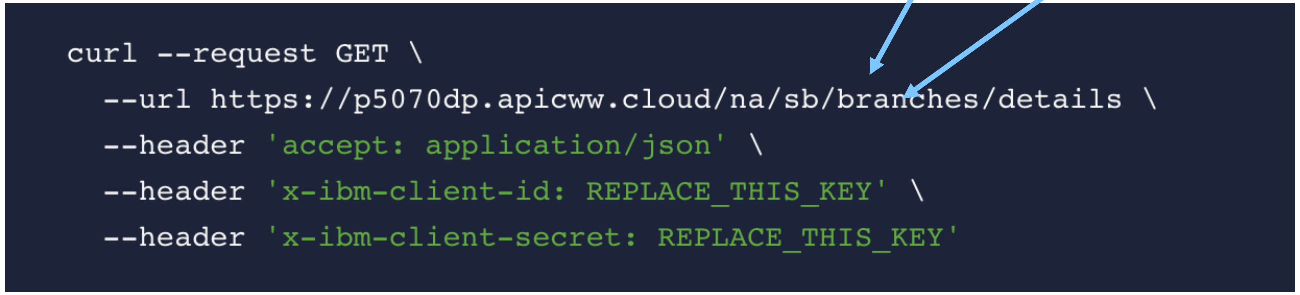
/Items/{id}

/Items/{id}/replace

/Items/findOne

/Items/update

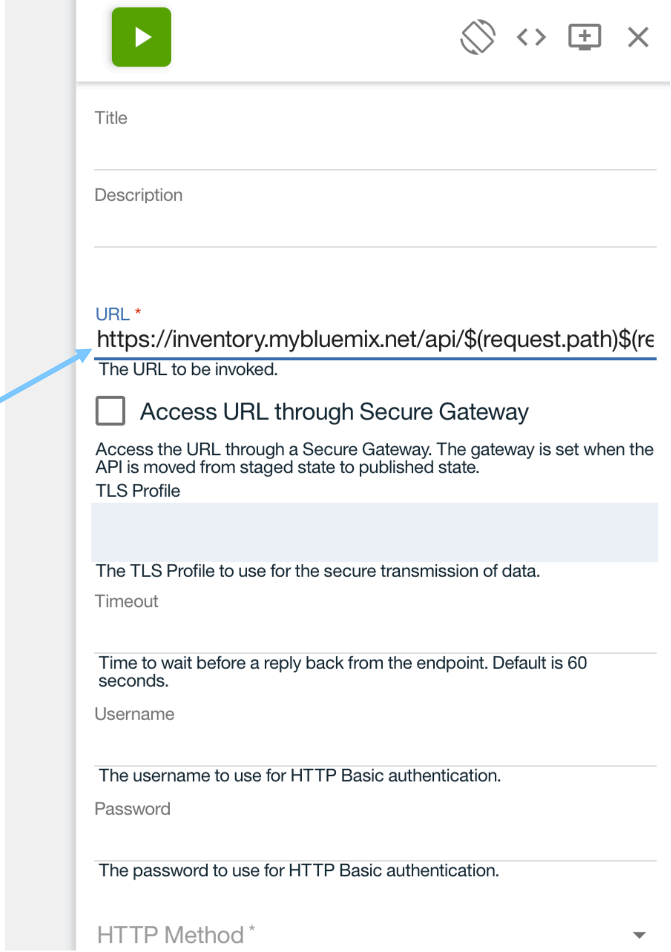
/Items/count



```
curl --request GET \
  --url https://p5070dp.apicww.cloud/na/sb/branches/details \
  --header 'accept: application/json' \
  --header 'x-ibm-client-id: REPLACE_THIS_KEY' \
  --header 'x-ibm-client-secret: REPLACE_THIS_KEY'
```

Accommodate the Invoke Panel

- Be Sure to account for API Base Path when setting up Invoke Panel Target
- Especially needed for Service APIs that mirror paths of the target service
- No matter what the API base path migrates to i.e. api/1 api/v2 /api/v3; the invoke will still proxy to the loopback service at /api



Title

Description

URL *
https://inventory.mybluemix.net/api/$(request.path)$(re
The URL to be invoked.

□ Access URL through Secure Gateway

Access the URL through a Secure Gateway. The gateway is set when the API is moved from staged state to published state.

TLS Profile

The TLS Profile to use for the secure transmission of data.

Timeout

Time to wait before a reply back from the endpoint. Default is 60 seconds.

Username

The username to use for HTTP Basic authentication.

Password

The password to use for HTTP Basic authentication.

HTTP Method *

DevOps Questions

- Open Conversation / Questions?

Up Next:

Demo