



CI\CD PIPELINE FOR POWER BI REPORTS

- Jayendran Arumugam
- Data Engineer @ EY
- Blog: jayendran.com
- Twitter: [@Jayendran_A](https://twitter.com/Jayendran_A)

AGENDA

- Need of CI/CD Power BI
- Defining Environments in Power BI
- Best Practices
- Power BI Authentication
- Power BI Action and PowerShell way of deployment

What is What ! ? 🤔

From Your View 🧐

- What is Power BI ?
- What is CICD Pipeline ?
- What is Azure DevOps ?



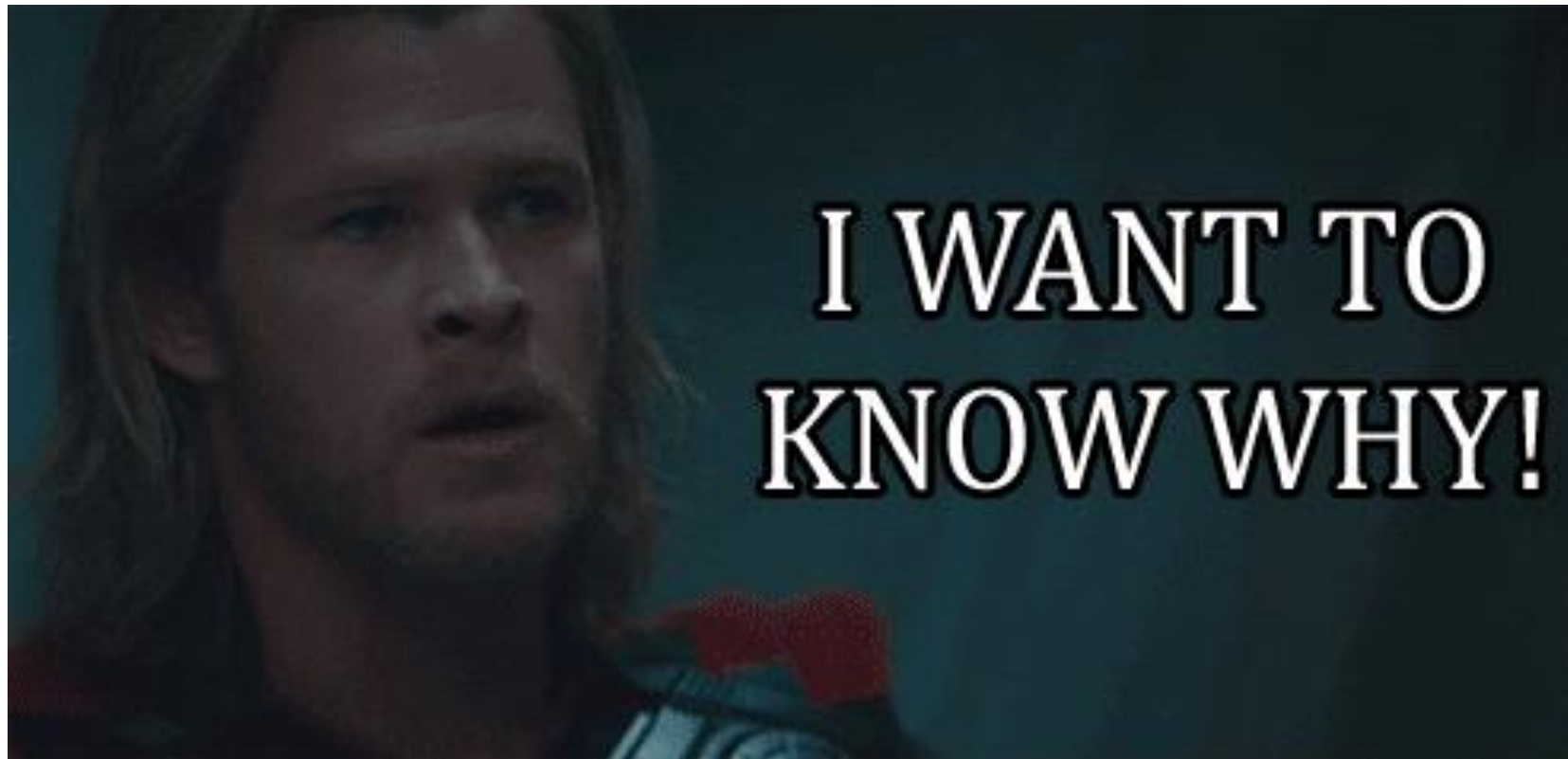


DEFINITIONS



- Power BI is a business analytics service by Microsoft. It aims to provide interactive visualizations and business intelligence capabilities with an interface simple enough for end users to create their own reports and dashboards
- Continuous Integration (CI) is a development practice that requires developers to integrate code into a shared repository. Each check-in is then verified by an automated build, allowing teams to detect problems early.
Continuous Delivery is the ability to get changes of all types—including new features, configuration changes, bug fixes and experiments—into production, or into the hands of users, safely and quickly in a sustainable way.
- Azure DevOps is a tool that allows teams to get up and running with managing their project and automating their workflows to increase productivity with a very short initial learning curve

Ok... Why We Need CI/CD For Power-BI Reports ?





BECAUSE OF.....

- Power BI Desktop (Client Tool) needs to be installed in all the Environment (Production, Dev, QA Servers)
- Without having a proper Source Version Control, the changes couldn't be tracked.
- The developer needs access to all the data sources across every environment. For E.g, if we have 2 environments say Dev and PROD and the data source is SQL server, then the developer needs access to both Dev SQL Server and PROD SQL Server.
- The developer needs to manually change the data source settings for every environment

Define Environments in Power-BI

- Different Power BI **Tenant** itself for each environment

- E.g,

- Dev Environment => Dev Azure Tenant

- UAT/QA Environment => UAT/QA Azure Tenant

- PROD Environment => PROD Azure Tenant



Power BI

©Microsoft Corporation 2019. All rights reserved.
Service version 13.0.11499.187
Client version 1911.3.022
Activity ID 6eba86b4-94a1-4ff6-b1f7-3fdb87453b09
Time Fri Nov 22 2019 15:53:44 GMT+0530 (India Standard Time)
Your data is stored in Central India (Pune)
Tenant URL s7ctid=4b1e49f8-1f29-4a77-b111-4f8ba2b99cd8

Close



Power BI

©Microsoft Corporation 2019. All rights reserved.
Service version 13.0.11499.187
Client version 1911.3.022
Activity ID 81653701-9356-4b50-b09c-1a772a4ce634
Time Fri Nov 22 2019 15:58:43 GMT+0530 (India Standard Time)
Your data is stored in Central India (Pune)
Tenant URL 'ctid=9f314ea2-0894-4854-ade4-2e7b4d0d846b

Close

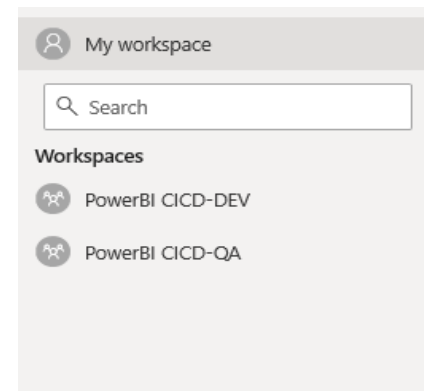
- Different Power BI **Workspaces** with-in the **Same/Common Tenant**

- E.g

- Dev Environment => Dev Workspace

- UAT/QA Environment => UAT/QA Workspace

- PROD Environment => PROD Workspae



Best Practices

- If you are using Cloud Data sources you don't need a gateway, but for on-premises data sources, we should use the gateway. Another Advantage of using the gateway for cloud data sources is your credentials were encrypted and stored in the gateway server and not with Power-BI
- Try to reduce hardcoding your data sources in your reports (or) In other words, try to use Parameters Jump for your data source as much as possible, Also keep in the mind updating parameter using PowerBI REST API don't support for the datasources like SQL, Oracle, Teradata & SapHana Direct Query Jump
- Organize your Source Control folder structure based on your requirements/client-specific /data sources.

**BEST
PRACTICE**

Authentication for Power BI

Service principal vs. master account

There are differences between using a service principal and a standard master account (Power BI Pro license) for authentication. The below table highlights some significant differences.

Function	Master User Account (Power BI Pro license)	Service Principal (app-only token)
Can sign in to the Power BI service	Yes	No
Enabled in the Power BI Admin portal	No	Yes
Works with app workspaces (v1)	Yes	No
Works with the new app workspaces (v2)	Yes	Yes
Needs to be a workspace admin if used with Power BI Embedded	Yes	Yes
Can use Power BI REST APIs	Yes	Yes
Needs a global admin to create	Yes	No
Can install and manage an On-premises data gateway	Yes	No

- **Master Account** :This basically username and password authentication along you need to reg an application in AAD to access Power BI API's
- **SPN (Service Principal)**: This is basically reg you application in AAD and add your app in powerbi portal. Read more [Jump](#)

Azure DevOps CI/CD For Power-BI Reports

- Using Power-BI Action Extension (Master Account Authentication)
 - Limitation:
 - SPN Authentication
 - Parameter updation
 - DataSet TakeOver
 - Update Credentials for the data source
- Using PowerShell Scripting (SPN Authentication)



Q&A



Before Going into the Demo..Any Questions So far





TIME FOR DEMO....



Enough of talking, Lets jump in to Action



References

- <https://social.technet.microsoft.com/wiki/contents/articles/53172.azuredevops-cicd-for-powerbi-reports.aspx>
- <https://gallery.technet.microsoft.com/AzureDevOps-CICD-for-fefd58b8>