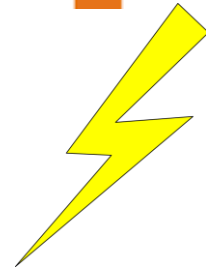


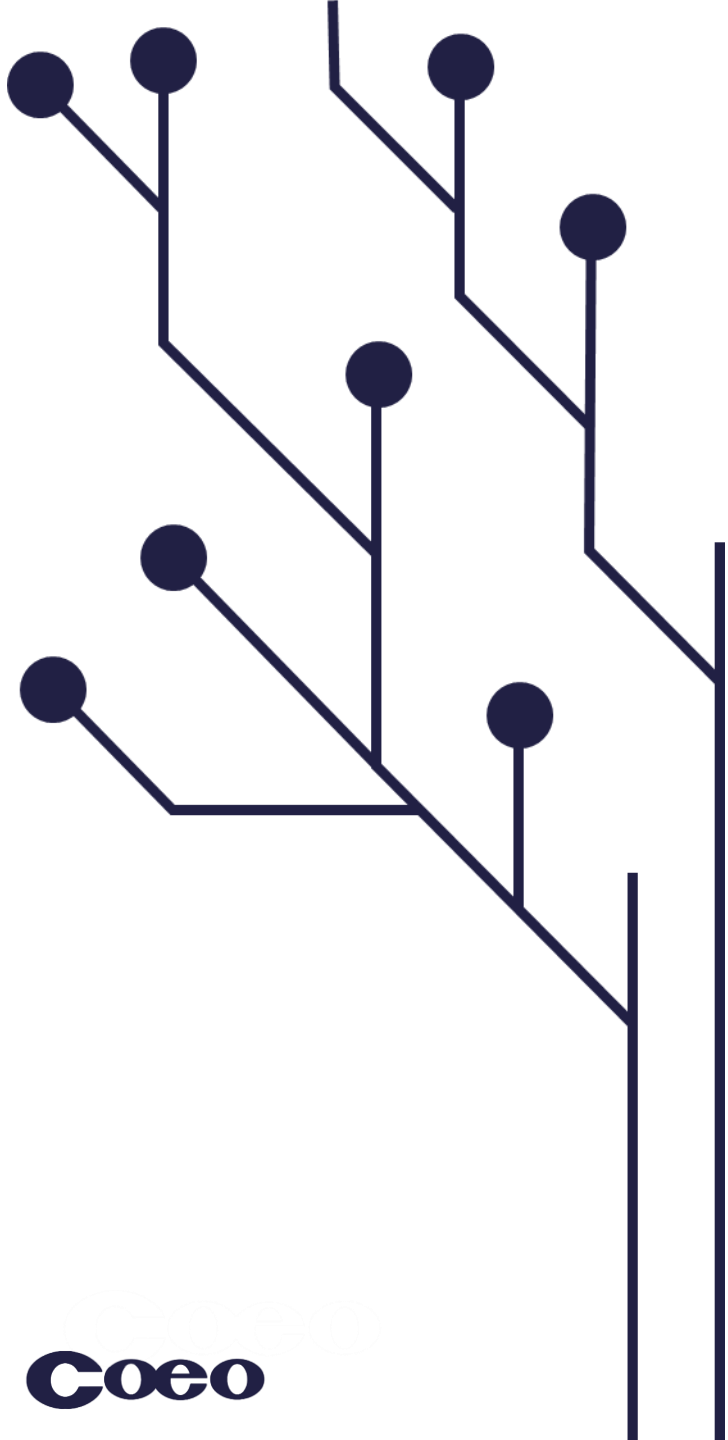
sqlbits



# Power BI Governance

sqlbits 2019

Andrew Mitchell



# Data Platform consultant



# Initialize Demo

# Power BI is fast becoming the Business Intelligence tool of choice for enterprises

## Challenges

- Single user development experience
- No version control for reports
- Manual workspace creation
- Manual migration of reports between workspaces



- Time Consuming and error prone
- Changes can be overwritten!
- No change tracking

## Ideal Solution

- Power BI Source Control Integration and change tracking
- Multi-User Editing
- Multi environment Publishing



- Requires changes to Power BI
- "Feature requests"

## Desired Outcomes

- Multiple users collaborating on the same project
- Change tracking for reports
- Report release cycles
- Gateway driven releases with approval workflow



- Align with Dev-Ops practices

# Power BI Governance

Goal: To allow customers to gain the most value from Microsoft investments

## Source Control and change

Take advantage of Azure DevOps to deliver more Dev-Ops capabilities

- GIT GUI for Windows
- Multi User Change Tracking
- Cloud Based Version Control

## Report Publishing

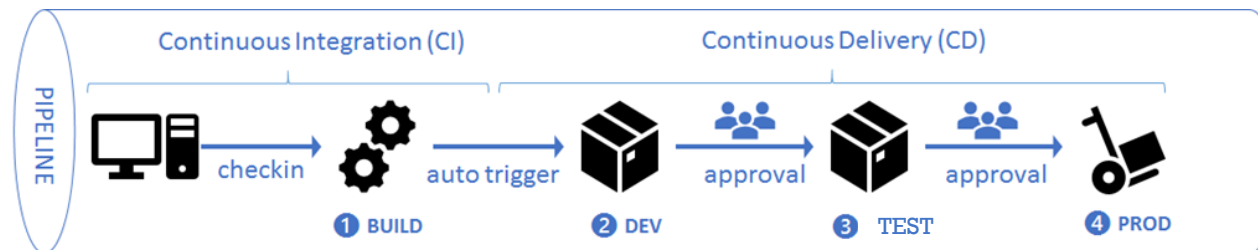
Azure DevOps Extension for Power BI

- Integration with the Power BI API
- Automatically publish reports using Continuous Integration (CI)
- Create Power BI Workspaces (Groups)

## Multi environment

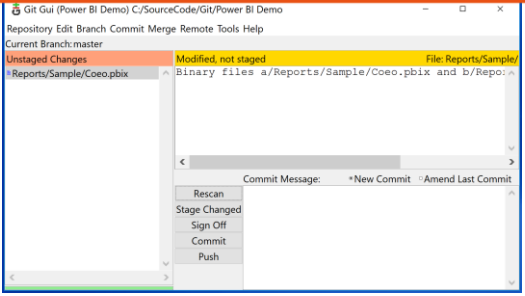
Azure DevOps release pipelines

- Pre-deployment approval
- Post-deployment approval
- Approval notifications
- Task and bug tracking



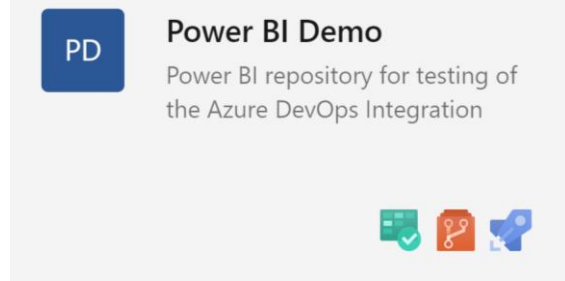
# What do we need?

## Power BI report handling




GIT GUI (Windows)

## Report Storage




GIT Repository (Azure DevOps)

## Report publishing



Custom Azure DevOps Extension

## Power BI report authoring



Power BI desktop

## Report rendering



Power BI.com

Report Updates

# Create a Project

The screenshot displays the Azure DevOps web interface. At the top left, the 'Azure DevOps' logo is visible. The breadcrumb navigation shows 'andymitchell / Power BI Demo / Overview / Summary'. A search bar and user profile icon are in the top right. The left sidebar contains navigation options: Power BI Demo, Overview, Summary (selected), Dashboards, Analytics views, Wiki, Boards, Repos, Pipelines, and Project settings. The main content area features a project header for 'Power BI Demo' with a 'Private' status and an 'Invite' button. Below the header, the 'About this project' section includes a description: 'Power BI repository for testing of the Azure DevOps Integration', a link to 'Power BI Demo / README.md', and sections for 'Introduction' and 'Getting Started'. The 'Introduction' section contains a TODO: 'Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.' The 'Getting Started' section contains a TODO: 'Guide users through getting your code up and running on their own system. In this section you can talk about:' followed by a list: 1. Installation process, 2. Software dependencies, 3. Latest releases, 4. API. To the right, the 'Project stats' section shows metrics for the last 7 days: 0 work items created and 0 completed, 0 pull requests opened, and 0 commits by 0 authors. At the bottom of the stats, two circular progress indicators show 0% for 'Builds succeeded' and 'Deployments succeeded'.

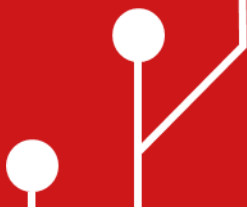
# Create a GIT Repo

Azure DevOps interface showing a repository named "Power BI Demo". The breadcrumb path is "andymitchell / Power BI Demo / Repos / Files / Power BI Demo". The current branch is "master". The repository contains a folder "Reports/Sample" and files "Data.xlsx" and "README.md". The commit history table is as follows:

Name ↑	Last change	Commits	
Reports/Sample	2/19/2019	ec54f5e0	Signed-off-by: Andrew Mitchell <andy.m
Data.xlsx	2/18/2019	d2e14677	Signed-off-by: Andrew Mitchell <andy.m
README.md	11/7/2018	1cba2a47	Added README.md file Andy Mitchell



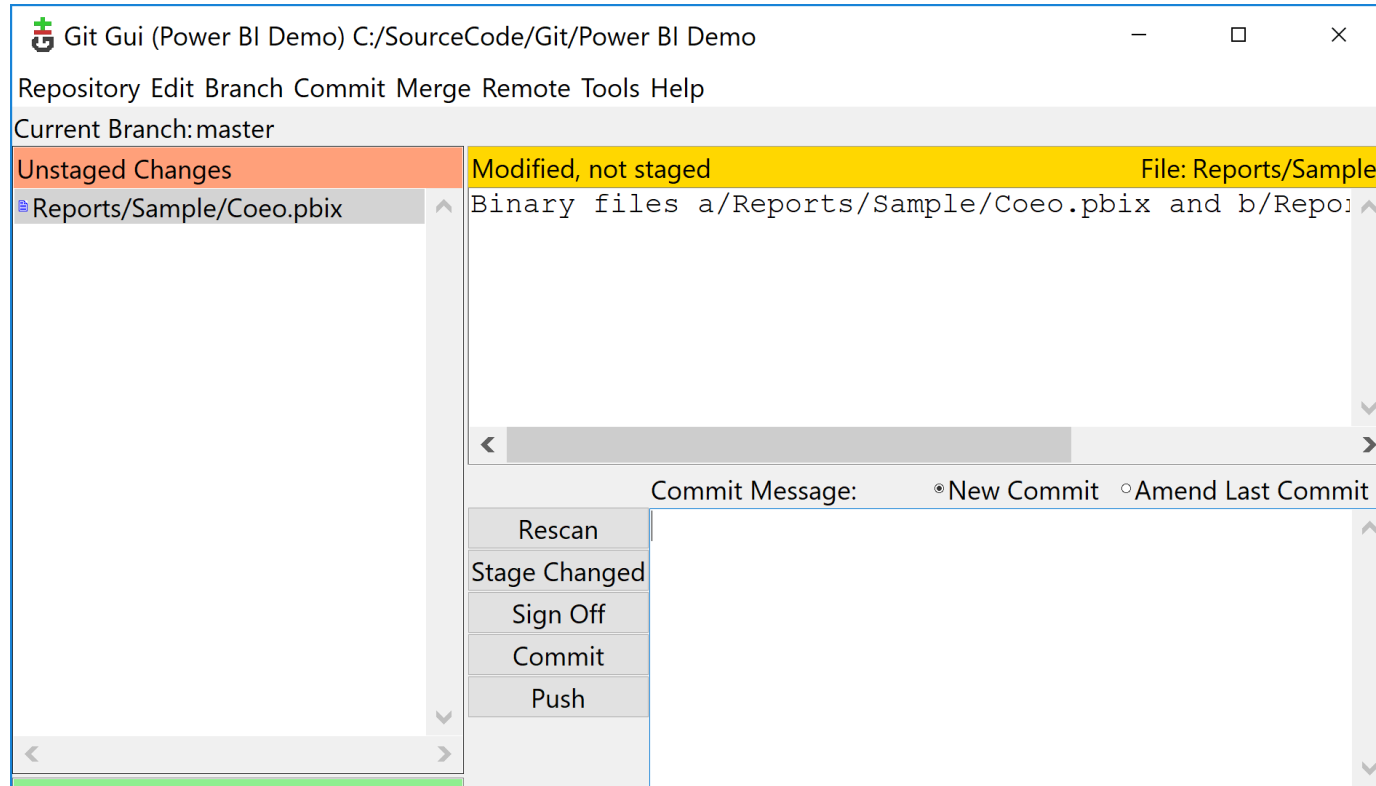
# Why GIT Gui?



- Free
- Simple to use
- Does not require additional knowledge (unlike Visual Studio / Code)

# Connected

<https://gitforwindows.org/>





# Azure DevOps Extension

# Add Workspace (Group)



Add Power BI Workspace (Group)  
Add Power BI Workspace



Display name \*

Add Power BI Workspace (Group)

Power Bi Service Endpoint \* ⓘ | [Manage](#)

Power BI API



+ New

Group Name \* ⓘ

NewWorkspace


ClientId \* ⓘ

1fe165cd-a395-4fde-ab05-81b073ec773b

Control Options ▾

Output Variables ▾

# Upload report to Group

 **Upload Report to Group**  
Upload pbix from VSTS

Display name \*

Power BI Service Endpoint \* [Manage](#)  
 [+ New](#)

Group Name \*

Reports to upload \*

ClientId \*

Overwrite if exists

Name Conflict Action \*  
 Ignore the operation  Abort the operation  Overwrite existing file

Prefer Client Routing

Ignore the operation

Creates Duplicate report if the report exists

Abort the operation

Aborts the import if the report exists

Overwrite existing file

Overwrites if the report exists

Default: Fails if report does not exist



# Published Demo

<https://github.com/Microsoft/PowerBIVSTSExtension>



[www.linkedin.com/in/andrewmitchelluk](http://www.linkedin.com/in/andrewmitchelluk)



@Andoius