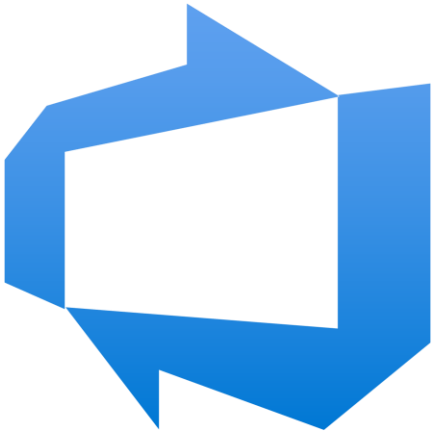
Microsoft Fabric and Azure DevOps – The story so far for 2025

Microsoft Fabric



Azure DevOps

Kevin Chant

Managing expectations

- Posts are available for majority demos shown
- All on Trial tenant (no NDA material)
- Includes details about Microsoft CI/CD workflows document
- SQL database in Fabric covered

Agenda

- Bio
- Intro to Microsoft Fabric
- Intro to Azure DevOps
- Configuring Azure Repos for Git integration
- Using Azure DevOps with suggested CI/CD workflow options
- CI/CD for Data Warehouses and SQL database in Fabric

Kevin Chant

- Lead Technology Advocate in the Netherlands
- Worked in IT since the days of Windows 95
- Experience in various sectors
- Various certifications, Data Platform MVP

- Twitter/Blue Sky: @kevchant
- LI: https://www.linkedin.com/in/kevin-chant/
- Blog: https://www.KevinRChant.com
- GitHub: https://github.com/kevchant

Microsoft Fabric

Comp analytics platform

Lake centric and open

Empower every Business user

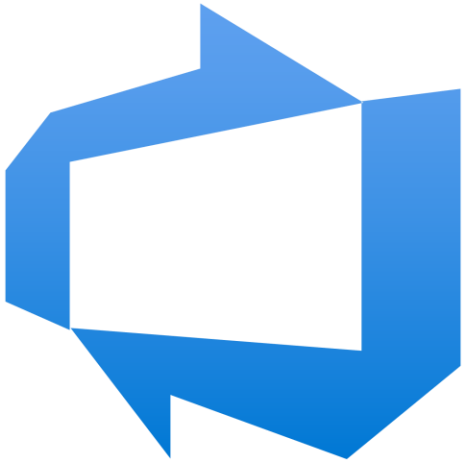
AI Powered

Fabric walkthrough

Azure DevOps

- Manages Application Lifecycle Management
- Collection of services
  - (Boards, Repos, Pipelines, etc)
- Two main versions
  - Services & Server

Azure DevOps Demo



About Microsoft Fabric Git Integration

- Allows **supported** items in a workspace to have metadata synchronized with a Git repository.
  - To be more precise a workspace synchronizes with a branch.
- Supports cloud-based versions of Azure DevOps and GitHub.
- Requires Fabric or Power BI Premium capacity.
- Items supported at various levels.

Supported items (allegedly)

- Data pipelines (preview)
- Dataflows gen2 (preview)
- Eventhouse and KQL database (preview)
- EventStream (preview)
- Lakehouse (preview)
- Mirrored database (preview)
- Notebooks
- Paginated reports (preview)
- Reflex (preview)
- Warehouses (preview)
- Reports (except reports connected to semantic models hosted in Azure Analysis Services, SQL Server Analysis Services, or reports exported by Power BI Desktop that depend on semantic models hosted in MyWorkspace) (preview)
- Semantic models (except push datasets, live connections to Analysis Services, model v1) (preview)
- Spark Job Definitions (preview)
- Spark environment (preview)
- SQL database (preview)

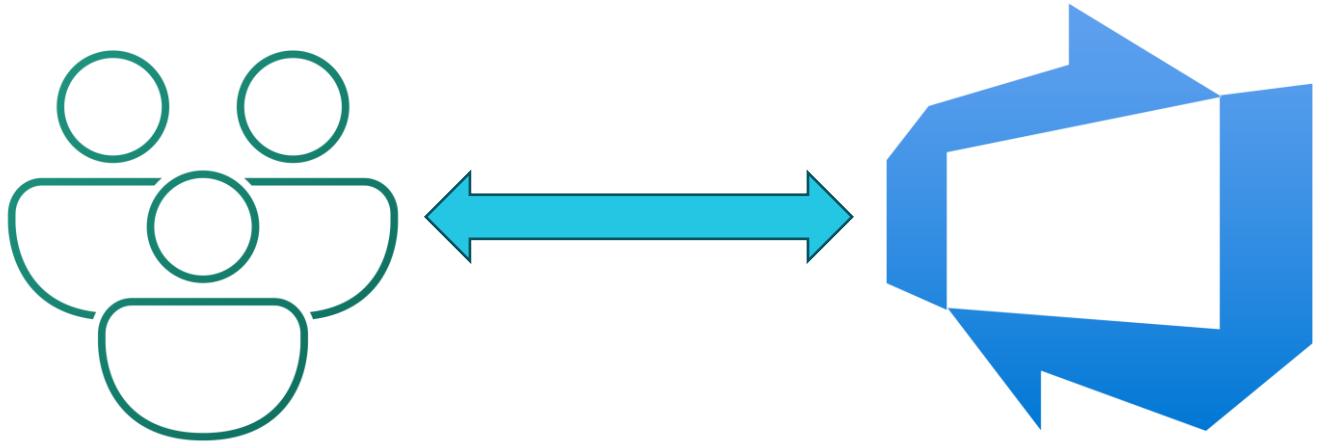
Configuring Microsoft Fabric Git Integration with Azure DevOps

- Utilizes Microsoft Entra ID authentication.
- User requires access to both workspace and repository.
- Entra user needs to be on same tenant, Azure DevOps org **does not**!
- Unsupported items can still be deployed in the workspace.

Security considerations

- Keep organization & repository in relevant region.
- Keep organization private.
- Avoid exposing sensitive details in Repos & Pipelines.
- Consider GitHub Advanced Security for Azure DevOps.

Git integration Demo



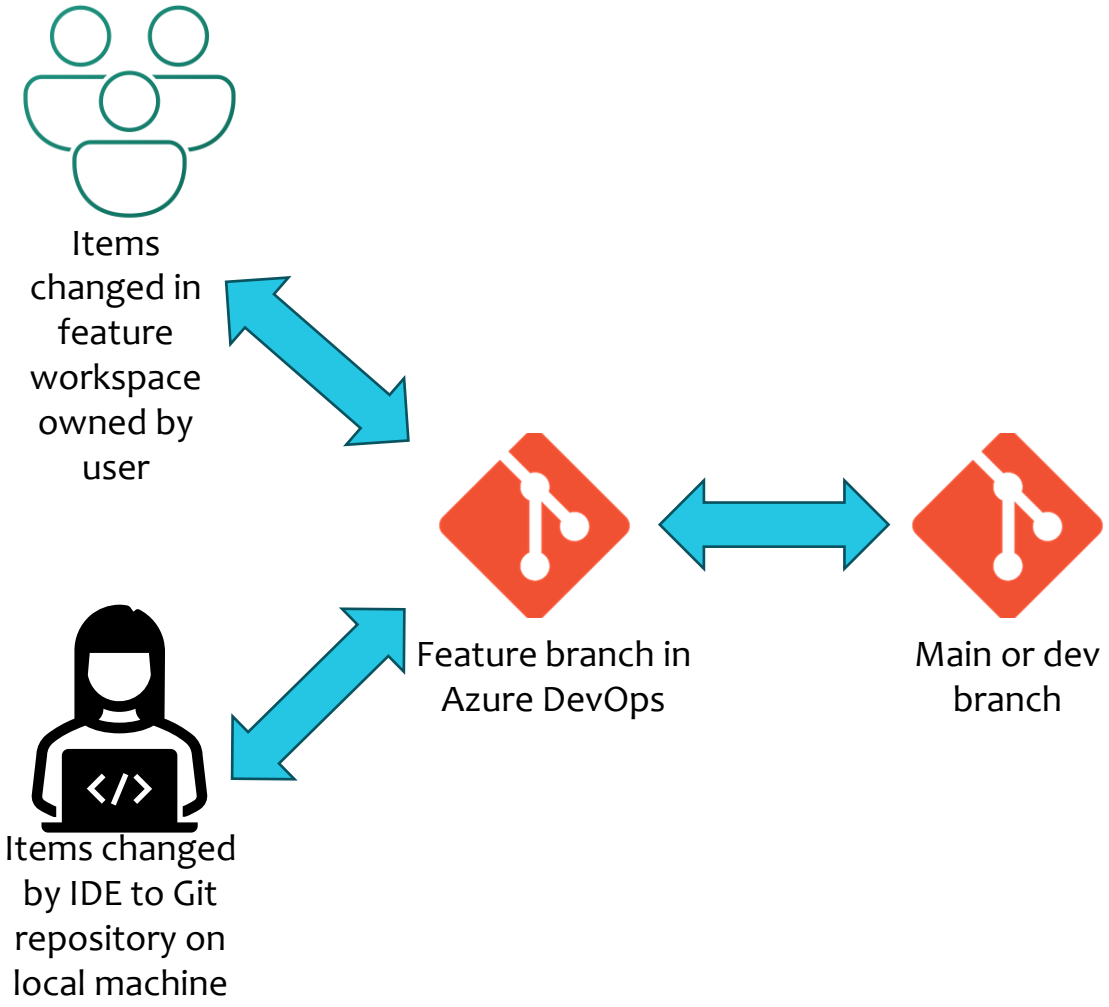
Microsoft Fabric workspace

Azure DevOps

Using Azure DevOps with suggested CI/CD workflow options

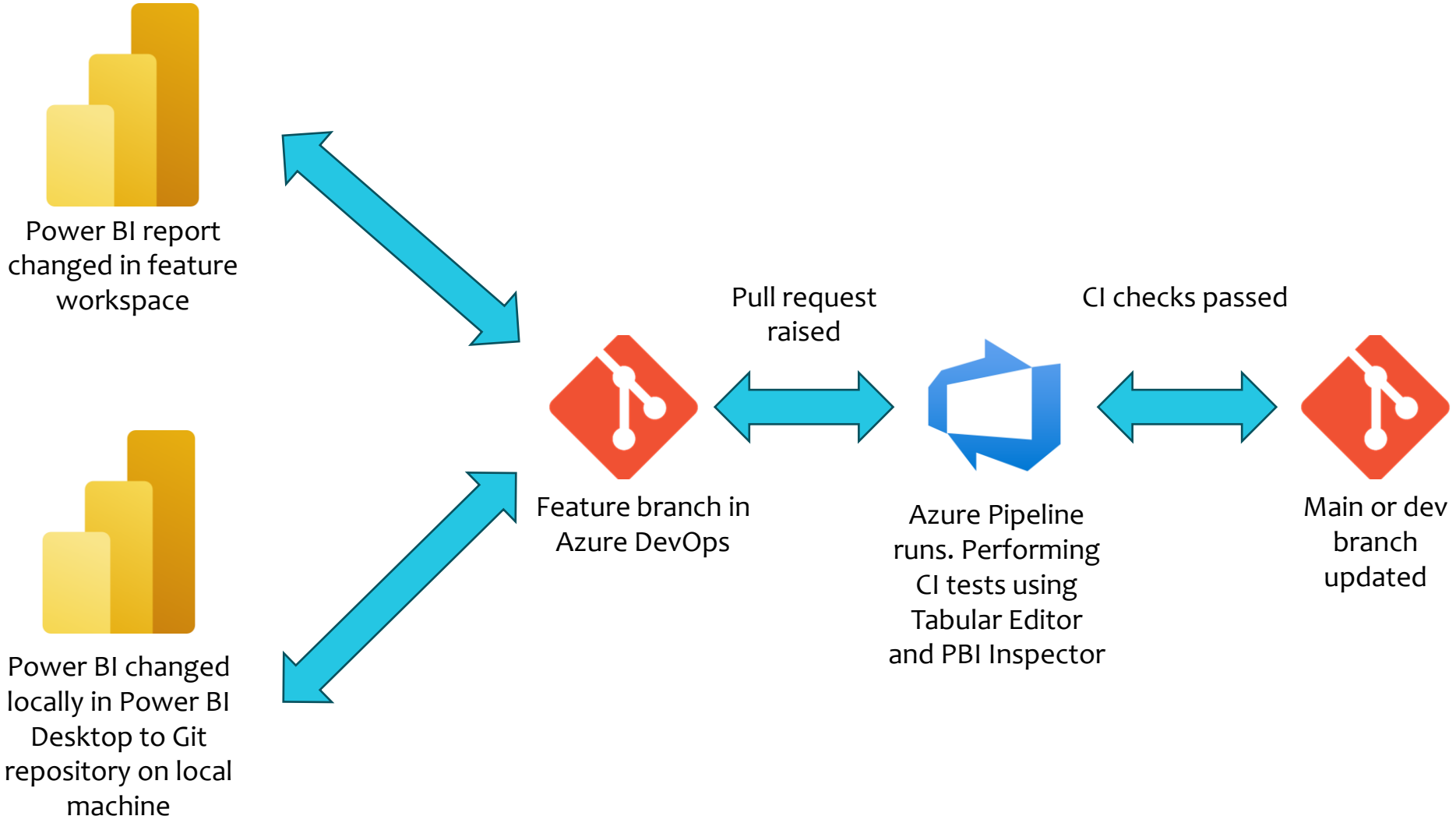
- Microsoft released article in September.
- I provide guidance how Azure DevOps fits into suggested CI/CD workflows.

Recommended development process



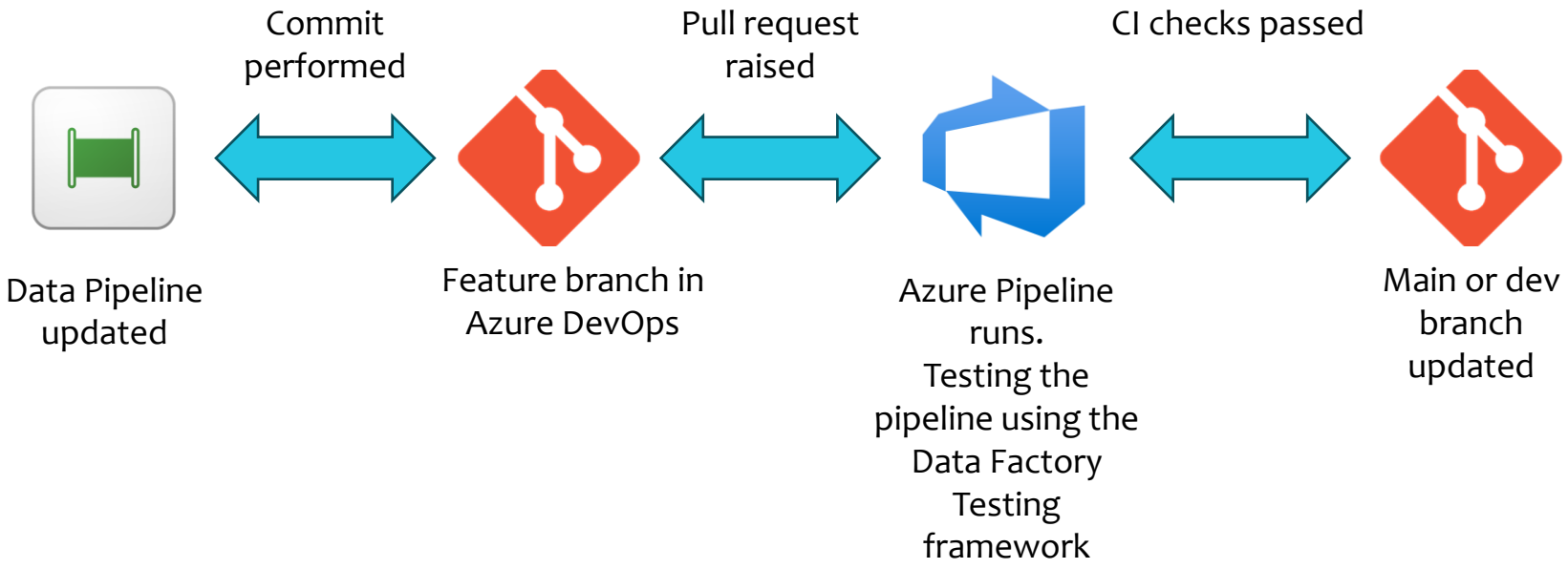
Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

One step further by introducing unit tests



Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

Testing Data Pipelines

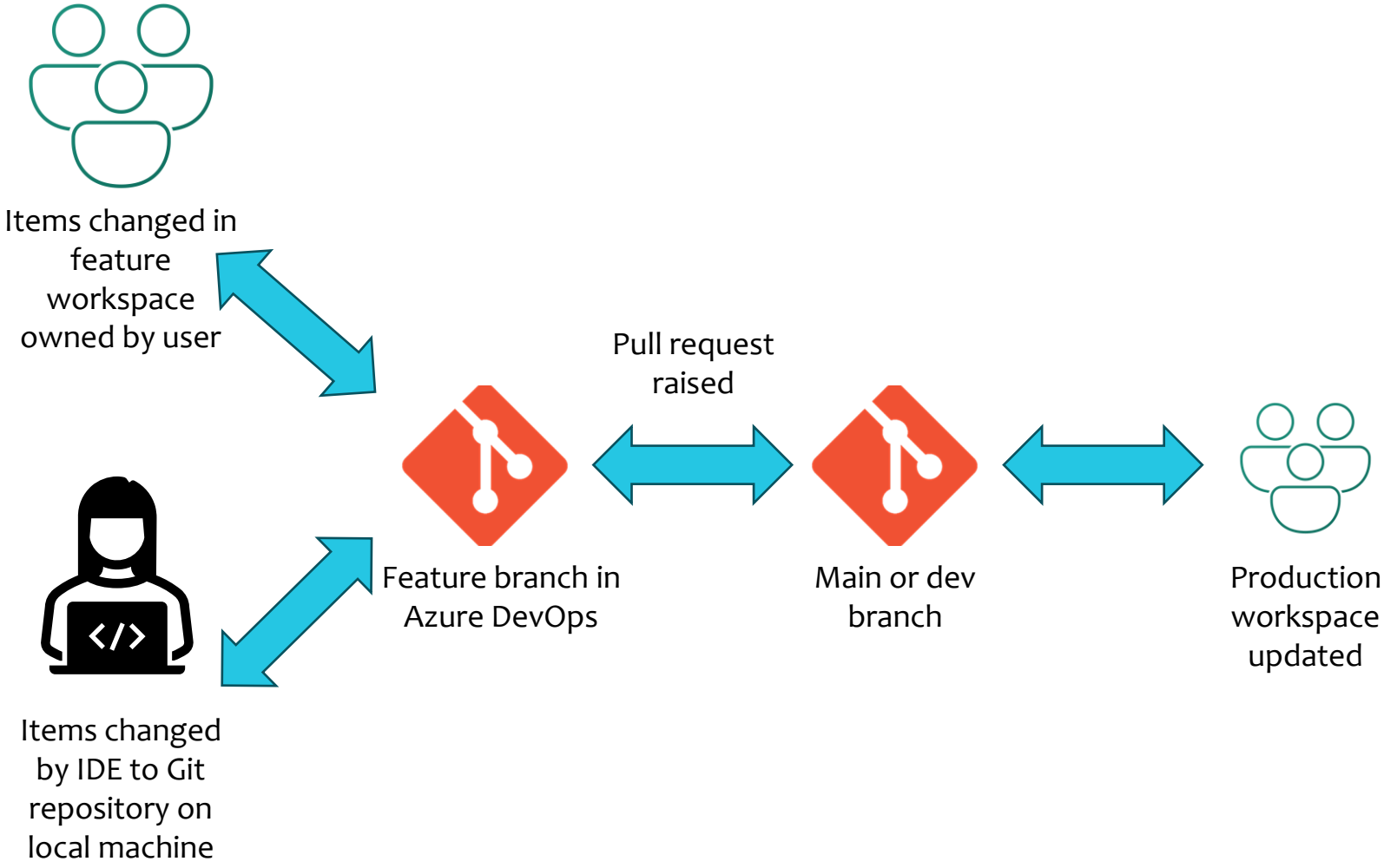


Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

Release Option 1 – Git-based deployments

- Deploy to multiple workspaces connected to the same Git repository.
- Achieved by having workspaces connected to different branches.

Release Option 1 - Diagram

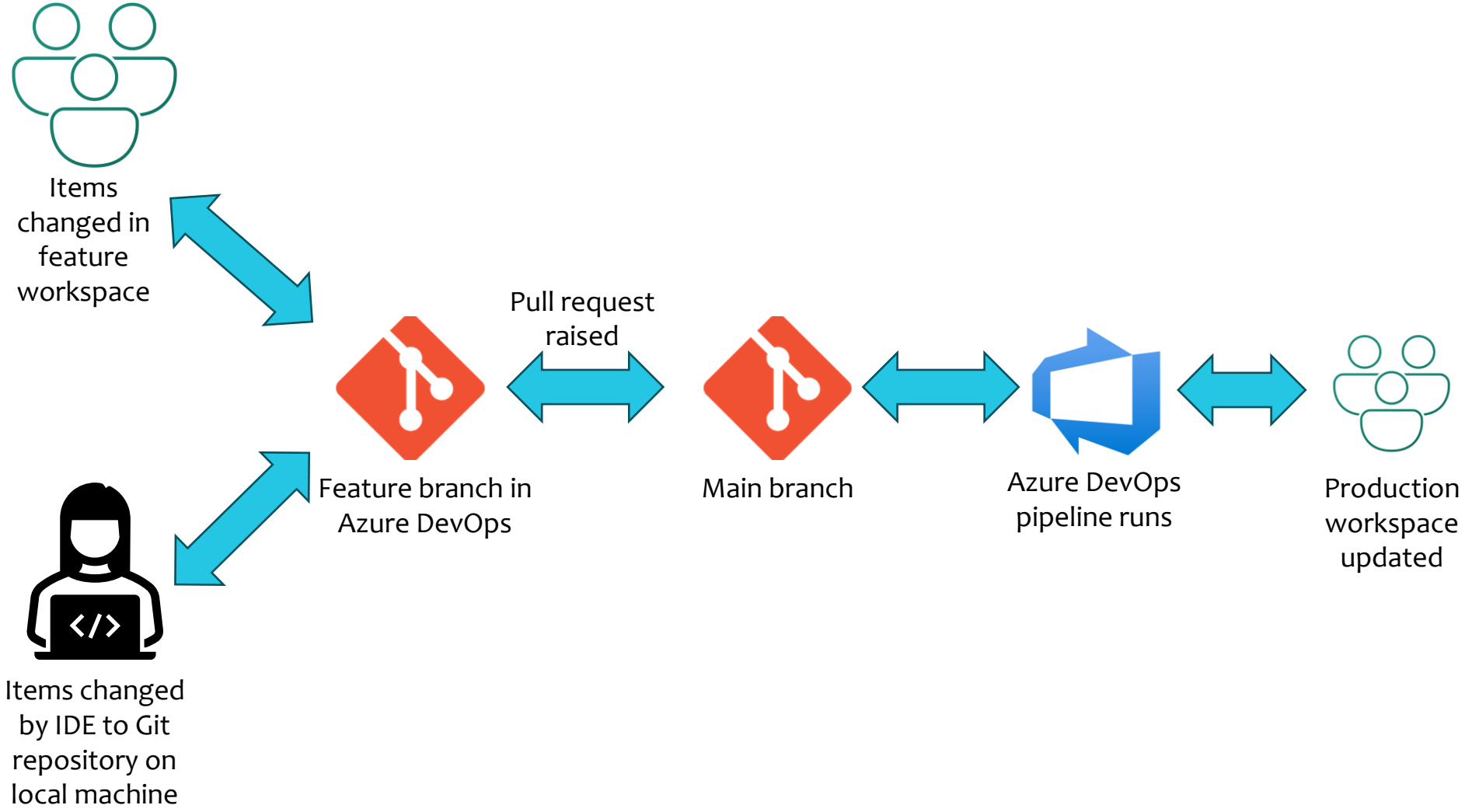


Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

Release Option 2 – Git-based deployments using build pipeline

- Deploy to different workspaces using Azure Pipelines.
- Recommendation is that each workflow contains a build and release process.
  - Build for unit tests.
  - Release to perform update.
- Note that for various APIs only Power BI items are supported by service principals at this moment in time.

Release Option 2 - Diagram



Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

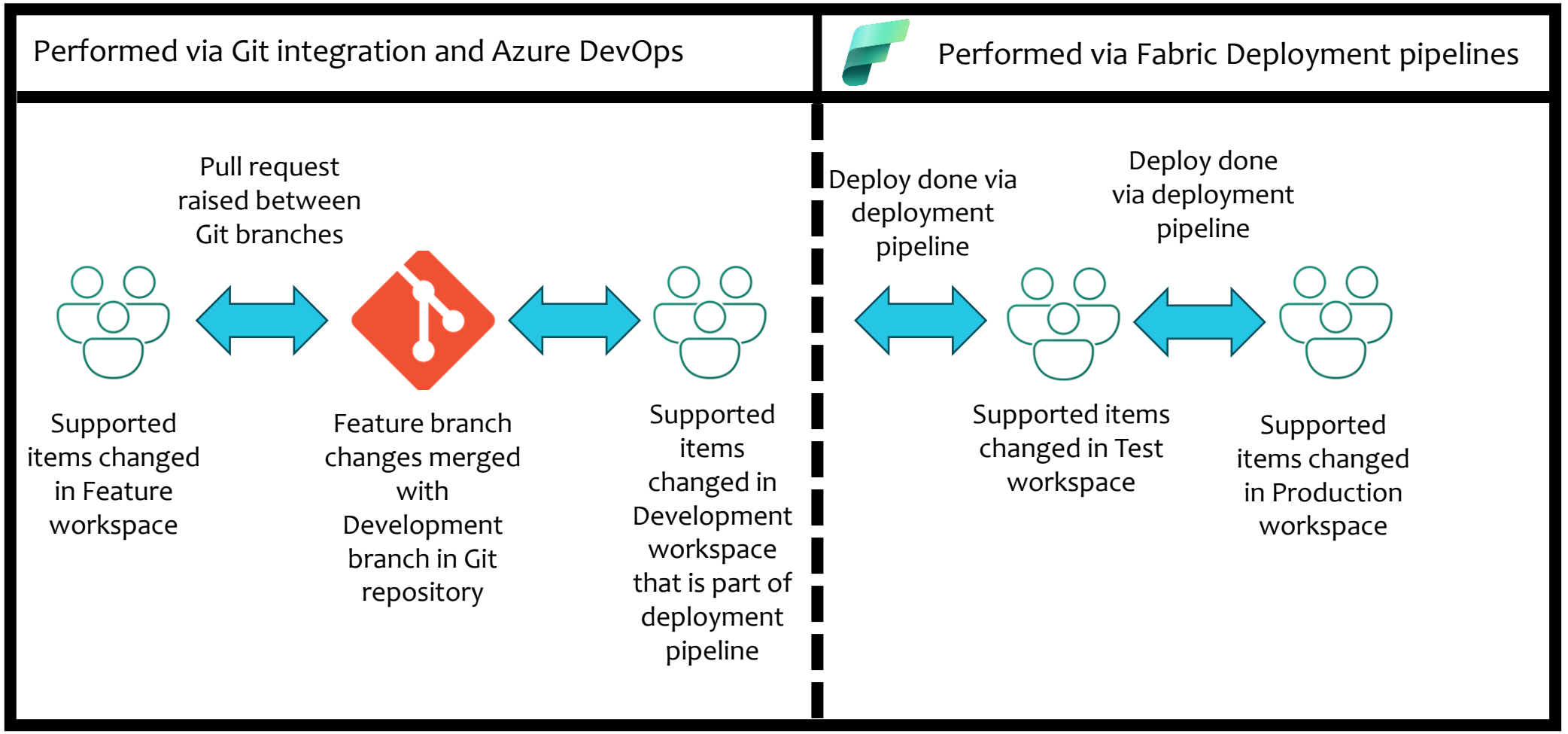
Azure Pipeline recommendations

- Consider YAML Pipelines.
  - You cannot do a PR on a classic pipeline!
- Self-hosted Azure Pipeline agents to keep workload secure.
- Avoid sensitive values in Pipelines. Store them in either:
  - Variable groups.
  - Azure Key Vault (preferred).
- Implement approvals process for production workloads.

Release Option 3 – Deploy using Microsoft Fabric deployment pipelines

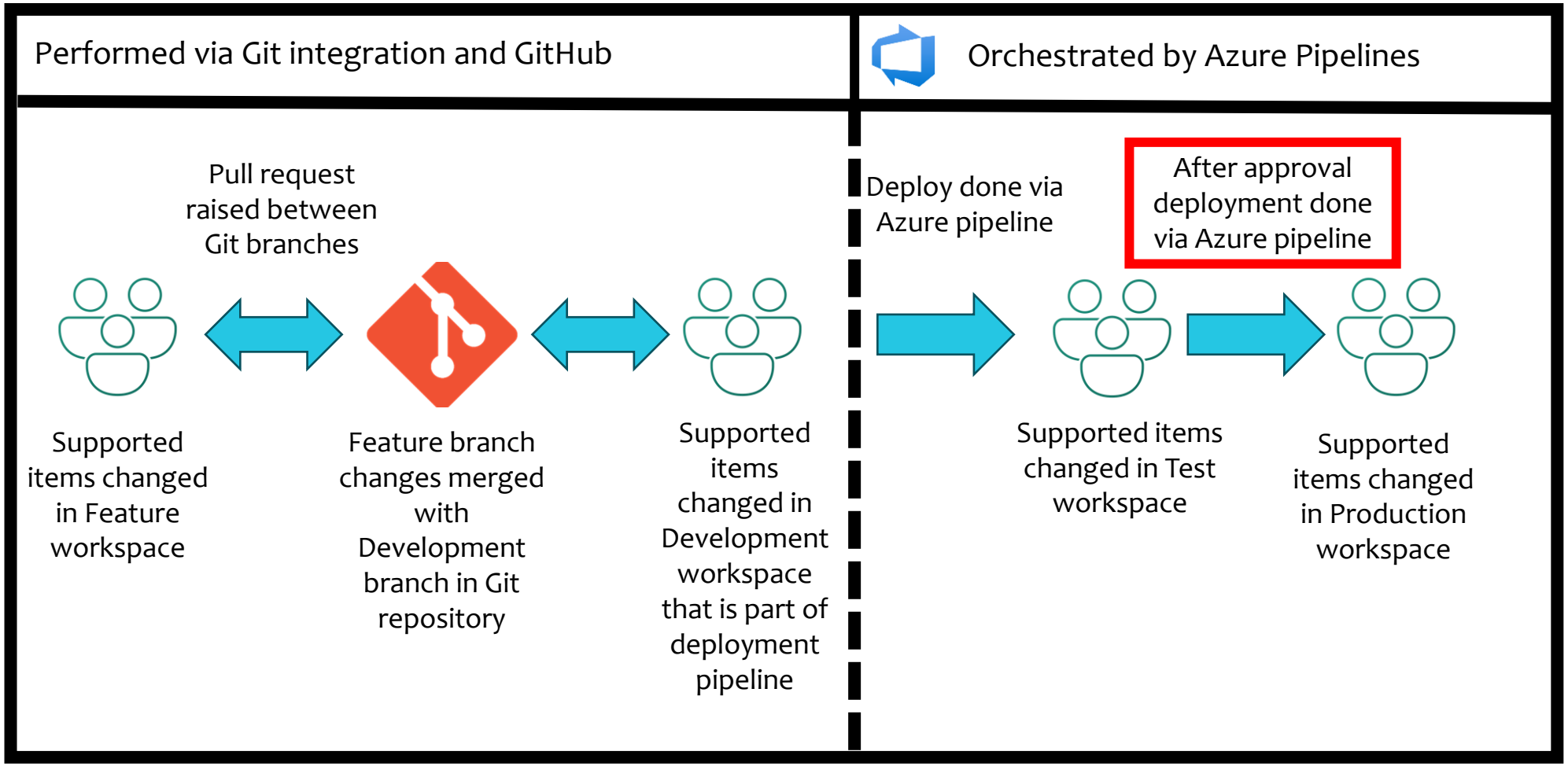
- Perform your pull request from feature branch to branch connected to a workspace that represents Dev stage of a deployment pipeline.
- From there orchestrate using Microsoft Fabric deployment pipelines.
- Alternatively, orchestrate to different Microsoft Fabric deployment pipeline stages using Azure DevOps.

Release Option 3 - Diagram



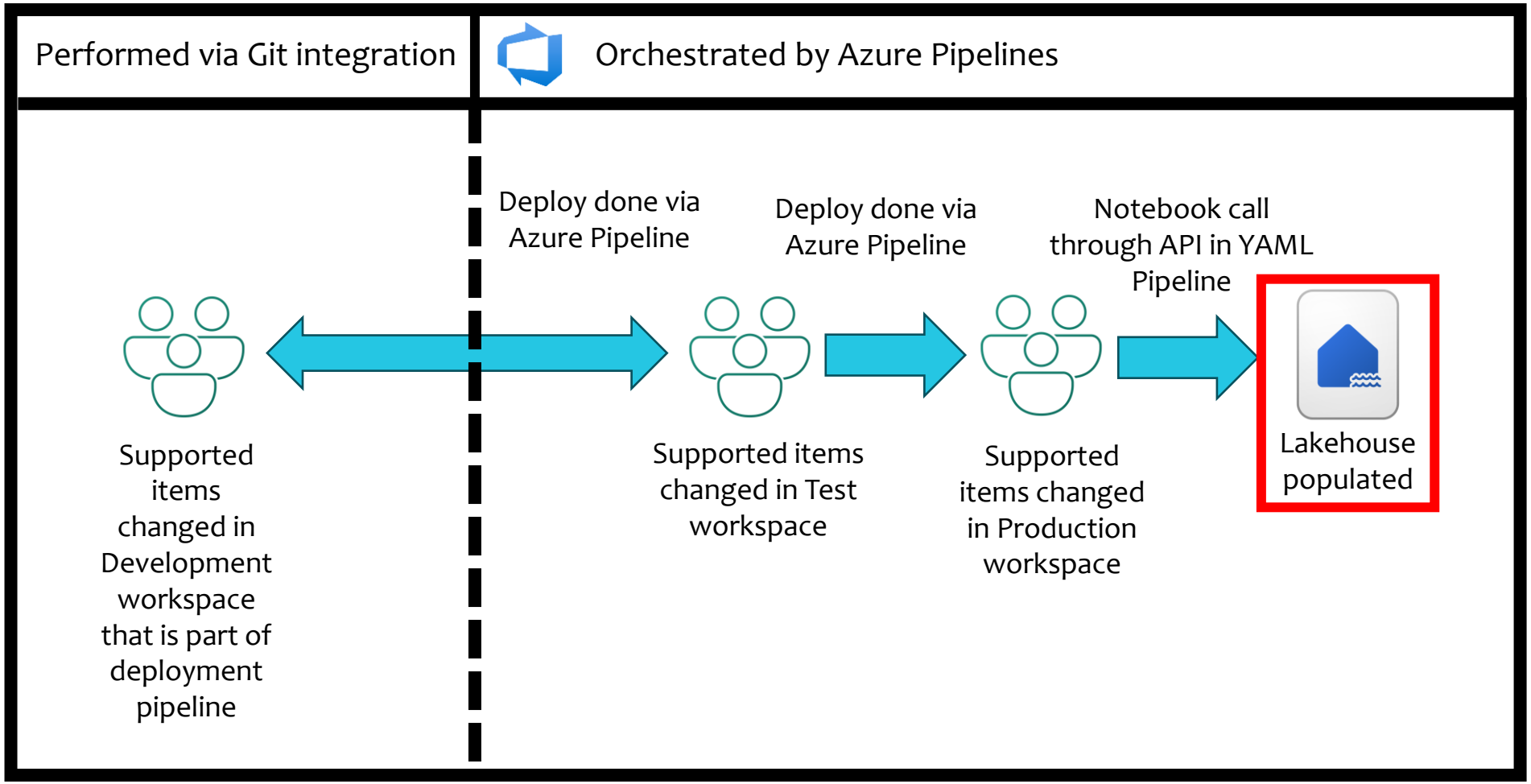
Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

Orchestrated by Azure Pipelines



Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

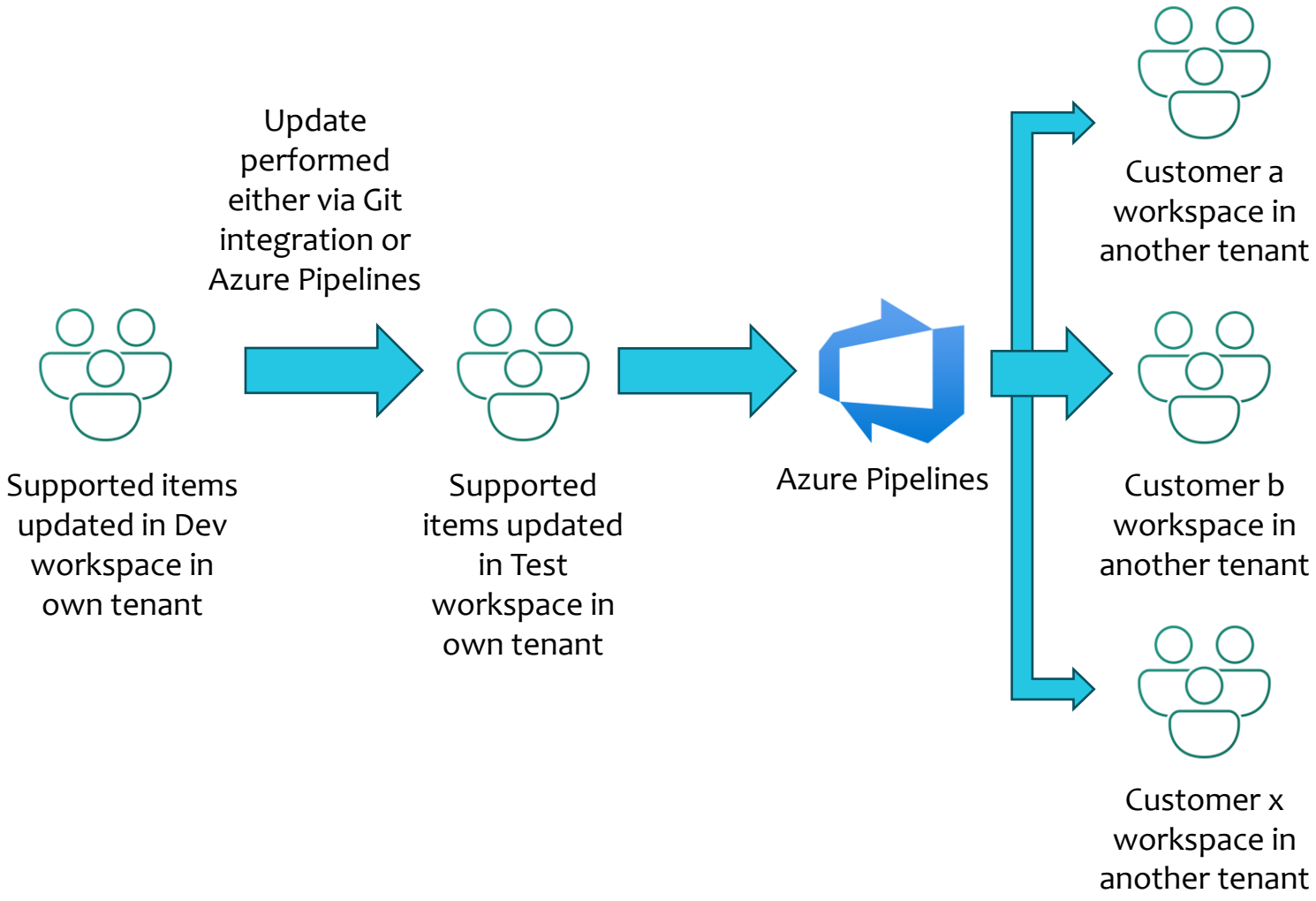
Another advantage of Azure Pipelines



Release Option 4 – For multiple customers/solutions/tenants

- Dev and test stages are managed in same Fabric tenant.
- Deployment to Prod stages to workspaces in other tenants using Azure Pipelines

Release Option 4 - Diagram



Demos

- Power BI Desktop Projects
- Deploying to multiple workspaces via Git integration
- Deploying via deployment pipelines

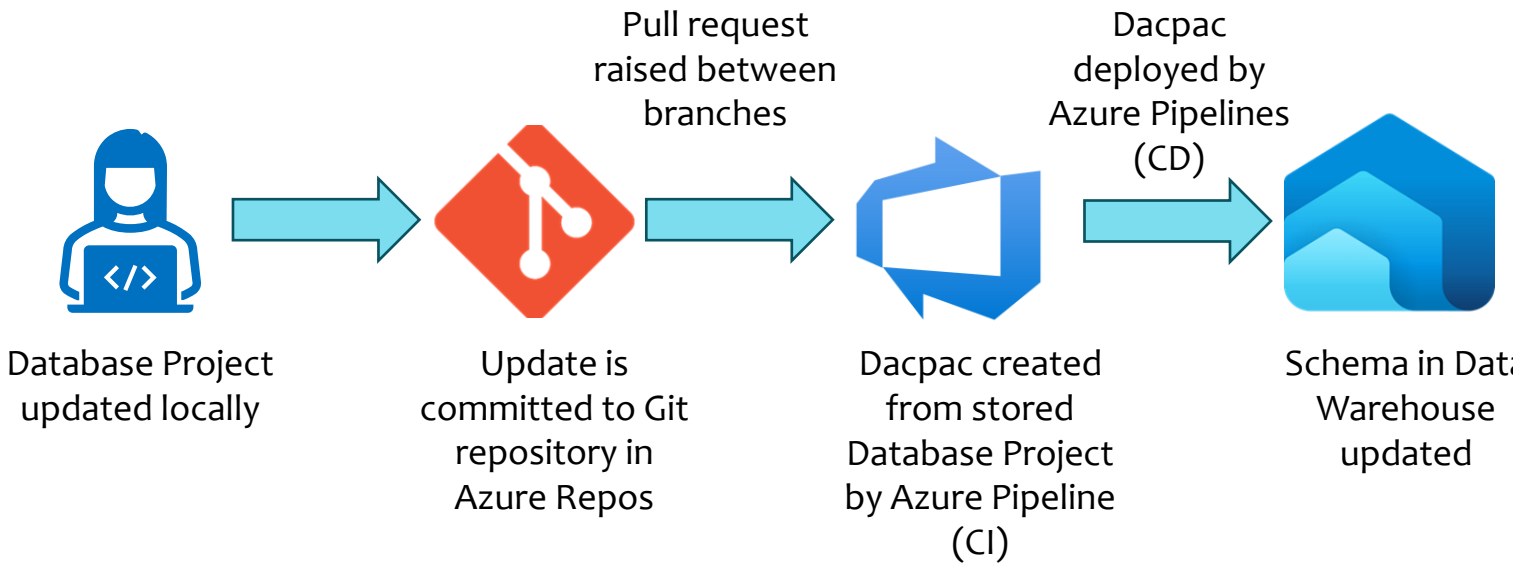
CI/CD methods for Data Warehouses and SQL databases in Fabric

- Can connect to Data Warehouse and SQL database in Fabric via connection string
- Allows state-based deployments via database projects.
- Supports Database Projects created in number of places.
- CI/CD can be performed with Azure Pipelines.

Why alternative CI/CD?

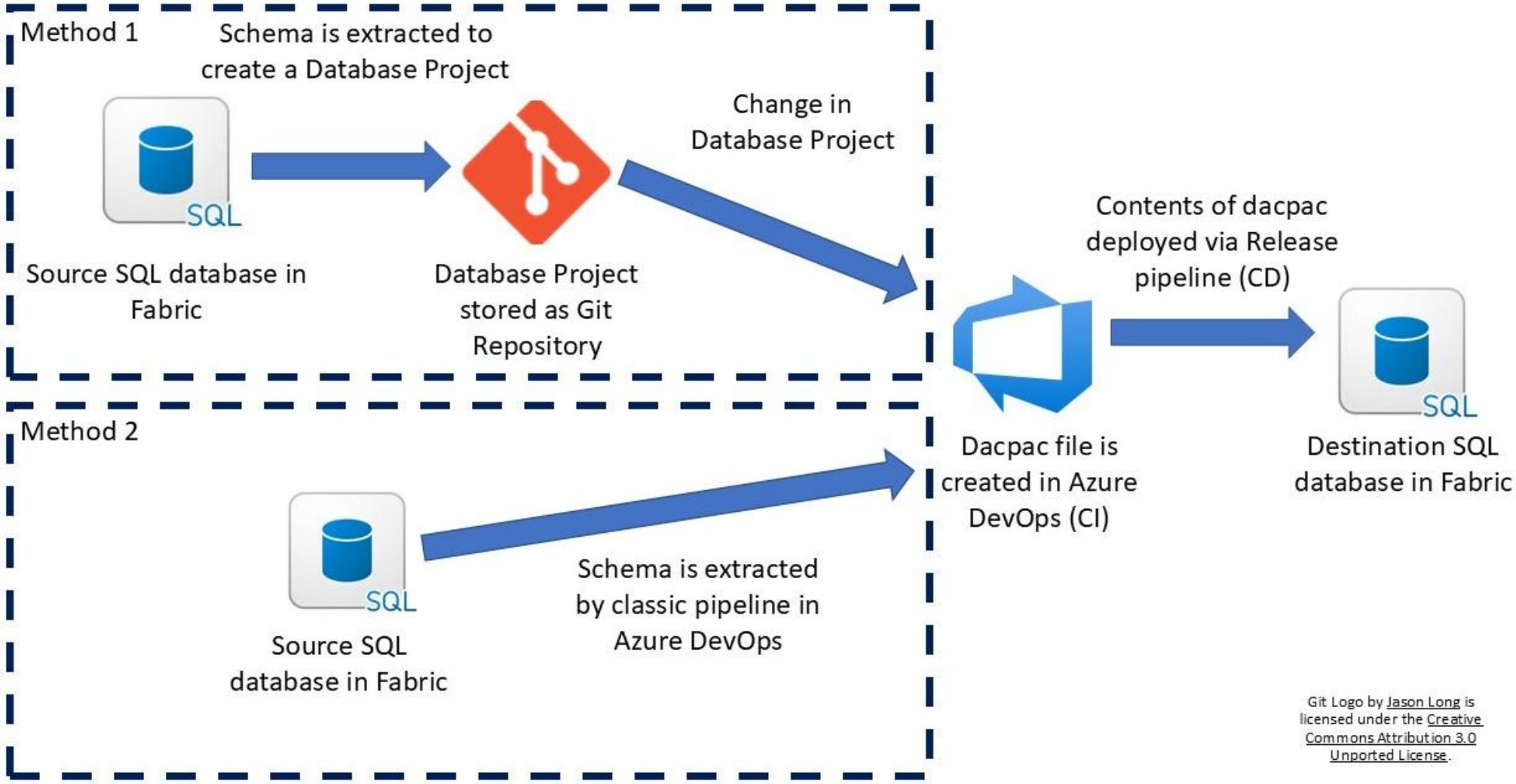
1. Multiple items in same workspace.
2. Git integration/Fabric deployment pipelines not implemented.
3. Deployments from another tenant.

CI/CD for Data Warehouses



Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

CI/CD for SQL databases in Fabric



Git Logo by Jason Long is licensed under the Creative Commons Attribution 3.0 Unported License.

Demos

- CI/CD for Data Warehouses.
- CI/CD for SQL database in Fabric

Questions

Thank you

- Twitter/Bluesky: @kevchant
- LI: https://www.linkedin.com/in/kevin-chant/
- Blog: https://www.KevinRChant.com
- GitHub: https://github.com/kevchant