

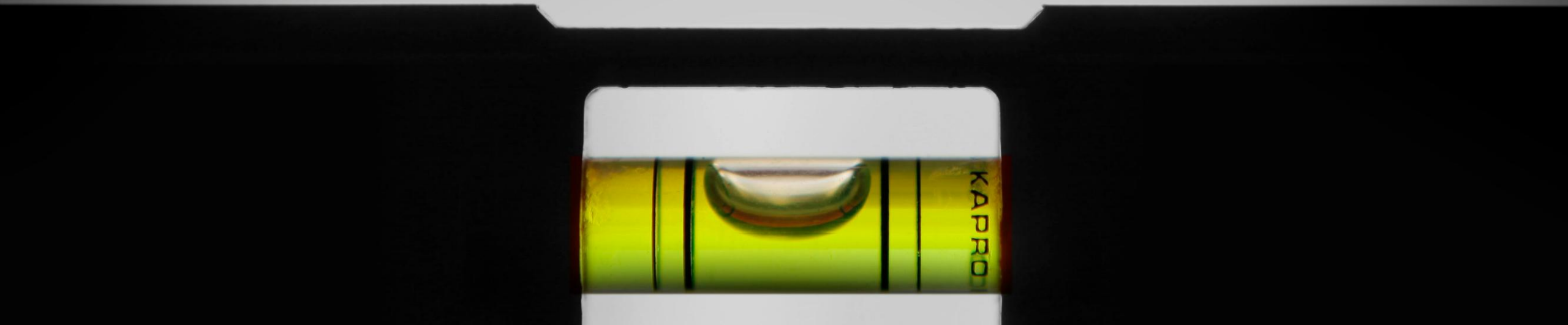
From Chaos to Collaboration

Introducing Git in Fabric Projects

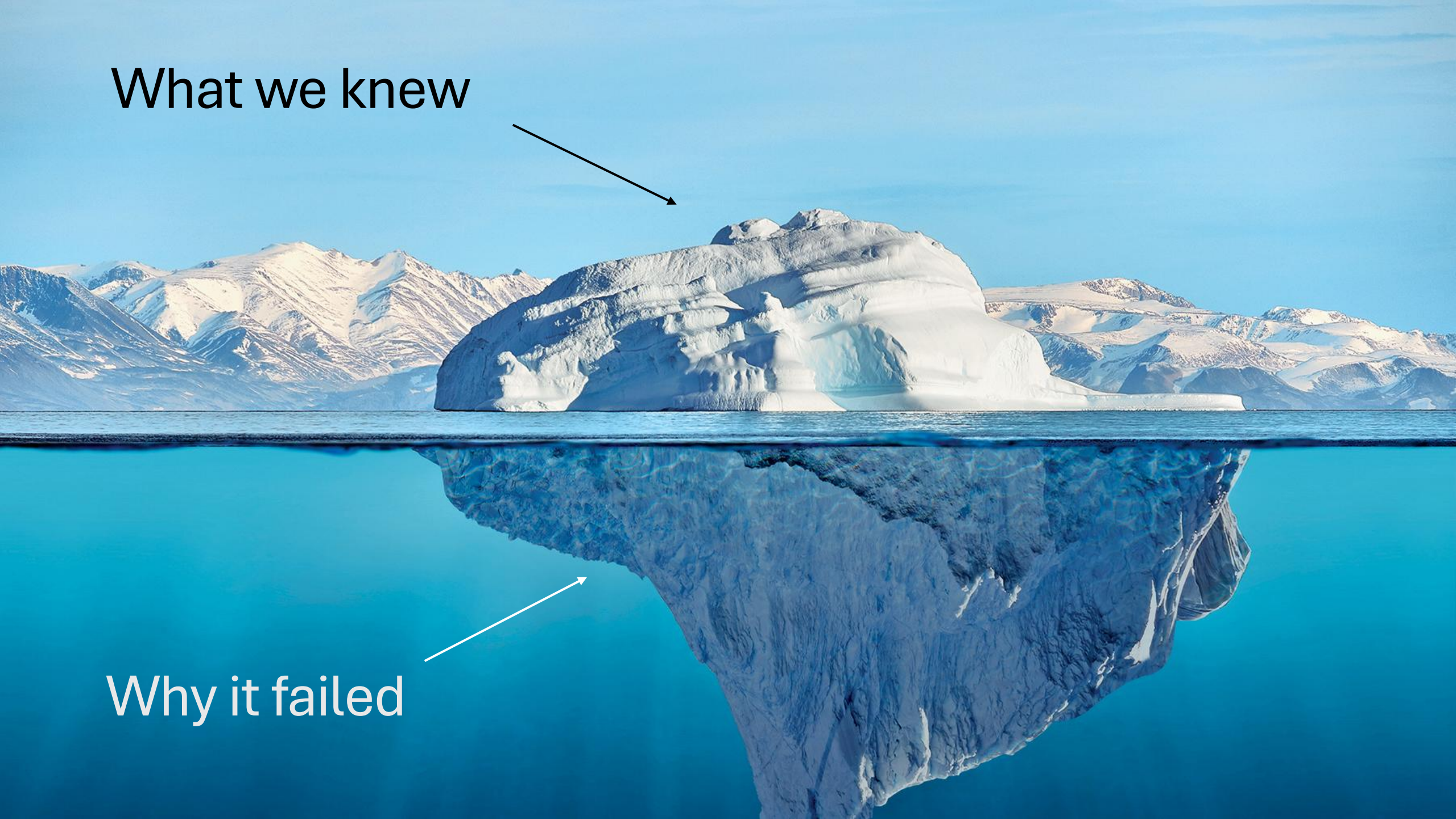
We were asked to help



We had the perfect Git course




And then...it failed.


What we knew
Why it failed


What we knew
Why it failed

Who are we?



Git enthusiast



Marisol Steinau

marisol.steinau@steinautech.com

- Autodidact Data Enthusiast
- > 10 years experience using Microsoft Data Platform
- Frequent guest at Legoland Resort Germany
- linkedin.com/in/marisol-steinau-bb1253253

Sebastian Steinau

sebastian.steinau@steinautech.com

- Techie (Azure, .NET, DevOps, IaC)
- Stepping into the data world with Fabric
- TV-Series geek and cook
- linkedin.com/in/sebastian-steinau-312888200

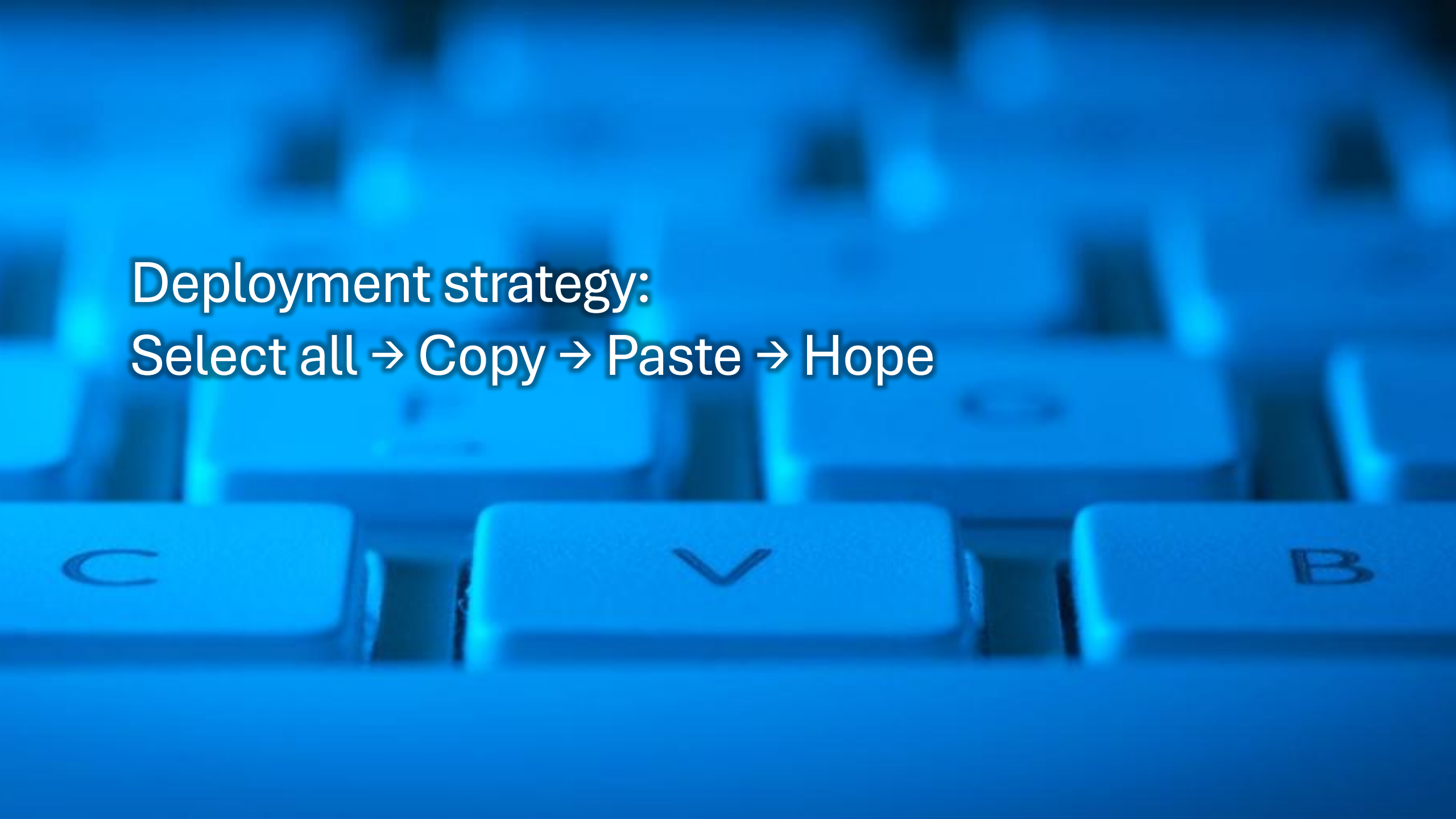
Our company: SteinauTech Data & Software Solutions GmbH, steinautech.com



The Chaos


It works on my machine.

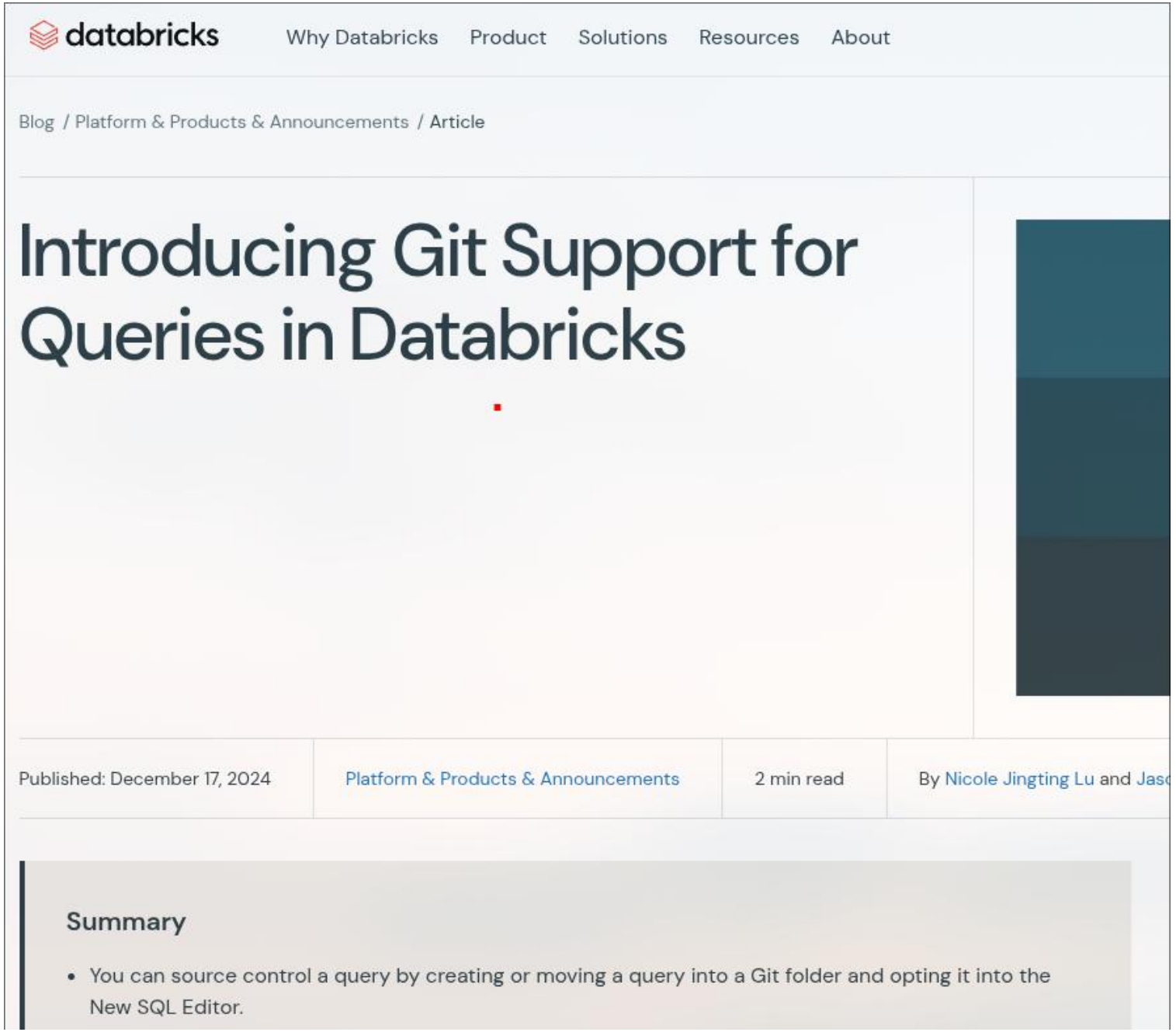

Working directly on PROD


Deployment strategy:
Select all → Copy → Paste → Hope


Aaaand it broke. Who knew.


Shared workspace

Databricks
2024-12-17



https://www.databricks.com/blog/introducing-git-support-queries-databricks

Databricks
2024-12-17
Snowflake
2024-04

Snowflake Builders Blog: ...

Best practices, tips & tricks from Snowflake experts and community

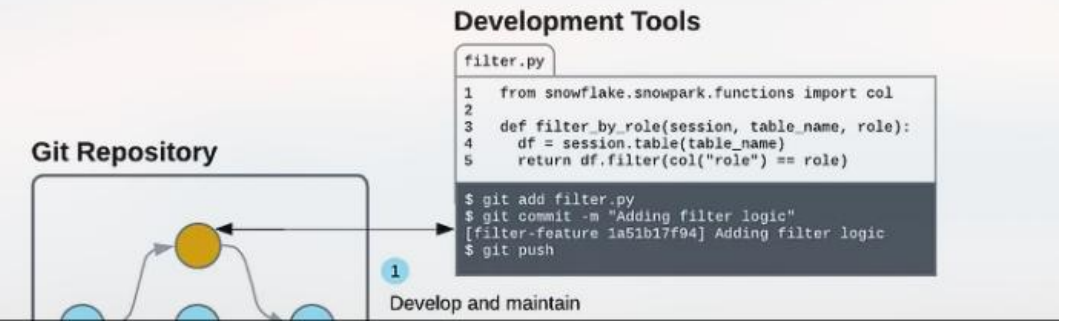
Follow publication

Connect Git to Snowflake, now GA

Jeff Hollan Follow 4 min read · Apr 3, 2024

274 9

We're excited to announce that the first wave of features for integrating git repos and git workflows into Snowflake is now in GA (as of Jul 2024) across all clouds! With Snowflake git integration, you can connect securely to a git repo hosted outside of Snowflake (GitHub, GitLab, Azure DevOps, or BitBucket currently supported) and have the contents of that repo synced to your Snowflake account. This allows you to have the "source of truth" for SQL scripts, Snowpark functions and procedures, Native Apps, and Streamlit Apps to pull from git.



https://select.dev/posts/snowflake-git-integration

Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Microsoft Fabric Updates Blog

Updates > All > Introducing git integration in Microsoft Fabric for seamless source con...

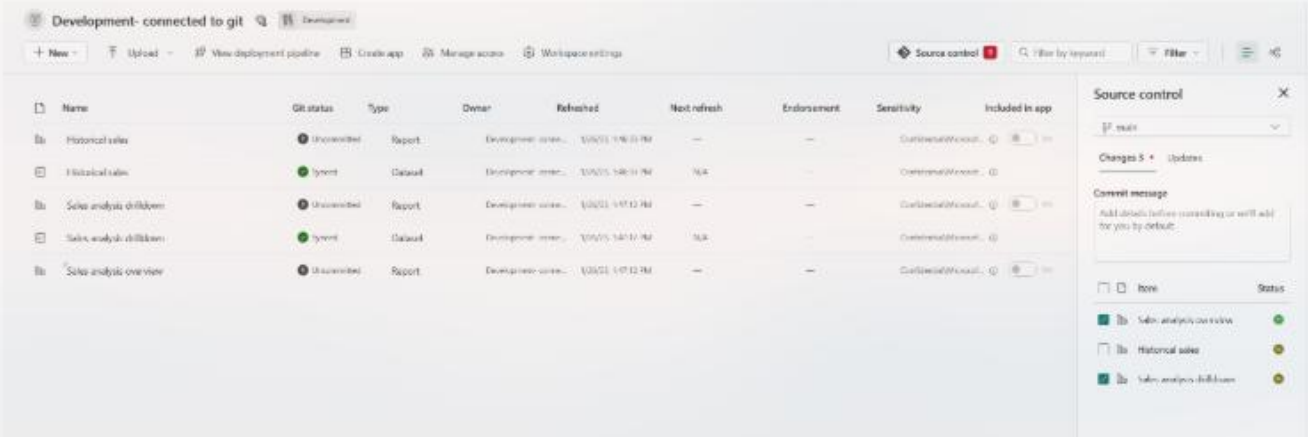
Introducing git integration in Microsoft Fabric for seamless source control management

Announcements Fabric platform

May 24, 2023 by Nimrod Shalit 129,046 Views

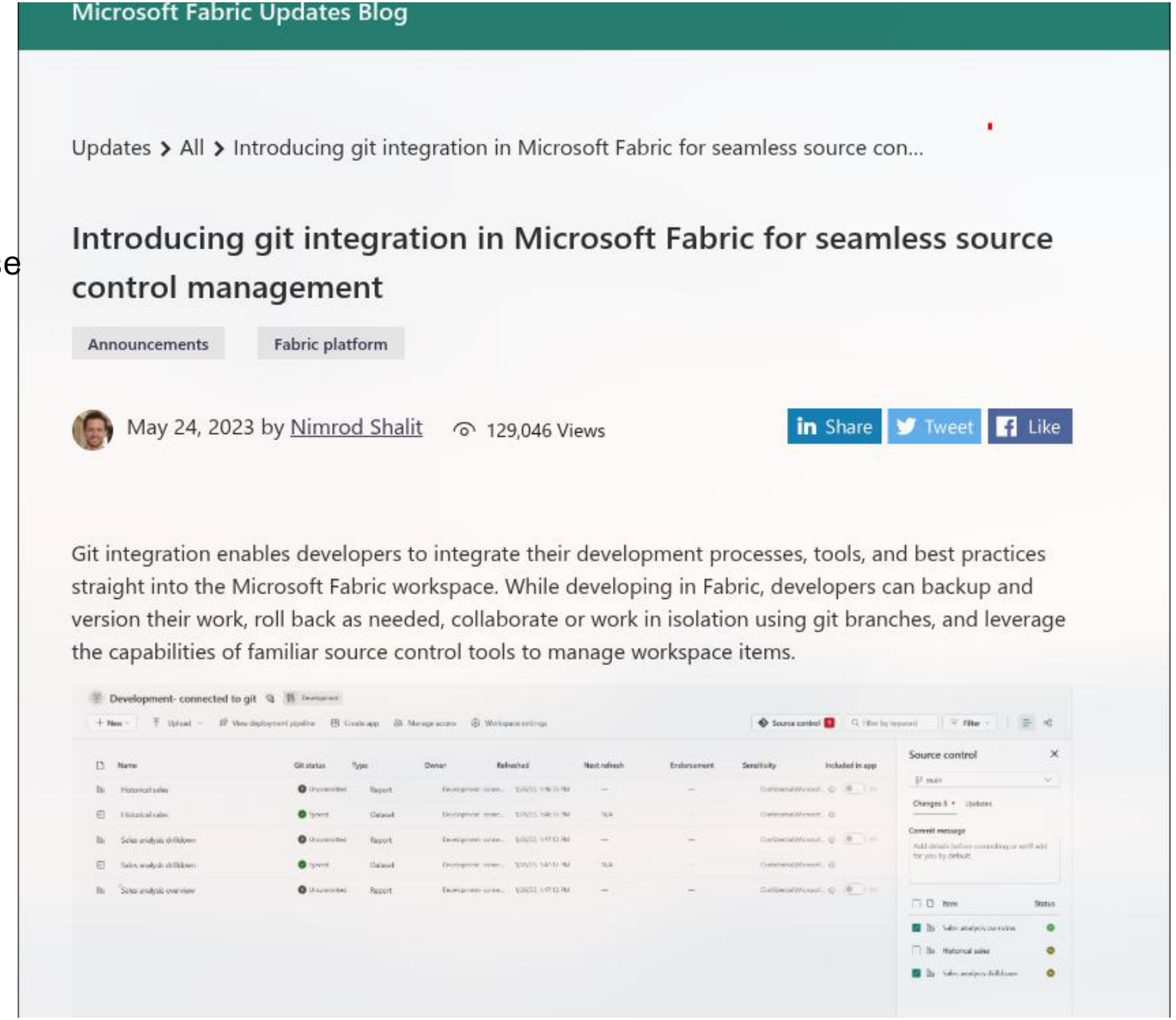
Share Tweet Like

Git integration enables developers to integrate their development processes, tools, and best practices straight into the Microsoft Fabric workspace. While developing in Fabric, developers can backup and version their work, roll back as needed, collaborate or work in isolation using git branches, and leverage the capabilities of familiar source control tools to manage workspace items.



https://blog.fabric.microsoft.com/en-GB/blog/introducing-git-integration-in-microsoft-fabric-for-seamless-source-control-management/

- Databricks 2024-12-17
- Snowflake 2024-04
- Fabric 2023-05-24
- Fabric Release 2023-05



https://blog.fabric.microsoft.com/en-GB/blog/introducing-git-integration-in-microsoft-fabric-for-seamless-source-control-management/

Power BI / TMDL
2025-09-24

Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Fabric Release
2023-05

Microsoft Power BI Updates Blog

ANNOUNCEMENTS DEVELOPERS POWER BI SEMANTIC MODEL

TMDL view (Generally Available)

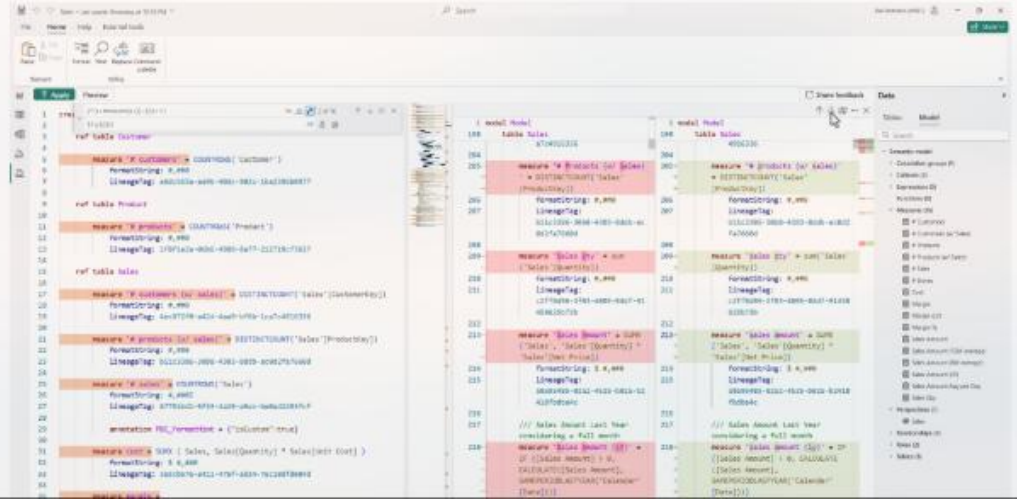


Rui Romano
Principal Program Manager

September 24, 2025

Share Tweet Like

With TMDL view you get full control and visibility over your semantic model with a modern code-first editing experience for Power BI semantic models using the human-friendly Tabular Model Definition Language (TMDL) modeling language.



WHAT
Power
analyz
busine
dashb
RE
TR
GET T
Ente
SU
SEARC
Admir

https://powerbi.microsoft.com/en-us/blog/tmdl-view-generally-available/

Power BI / TMDL
2025-09-24

Theobald
2025-06

Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Fabric Release
2023-05

Power BI / TMDL
2025-09-24

Theobald
2025-06

Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Fabric Release
2023-05

Airbyte
2020

Matillion
2019-08

Power BI / TMDL
2025-09-24

Theobald
2025-06

Insomnia
2024-12



Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Fabric Release
2023-05

Postman
2021



Airbyte
2020

Matillion
2019-08

Power BI / TMDL
2025-09-24

Theobald
2025-06

Insomnia
2024-12



Databricks
2024-12-17

Snowflake
2024-04

Fabric
2023-05-24

Fabric Release
2023-05

Postman
2021



Airbyte
2020

Matillion
2019-08



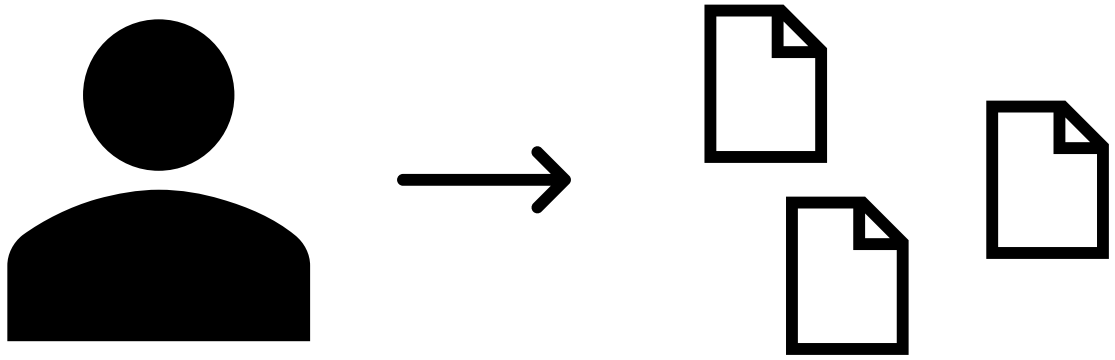
2008

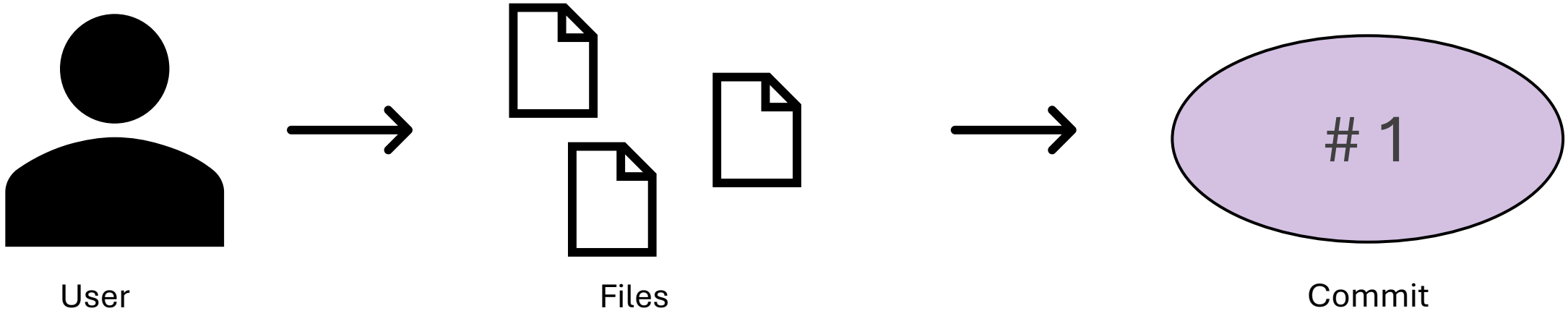


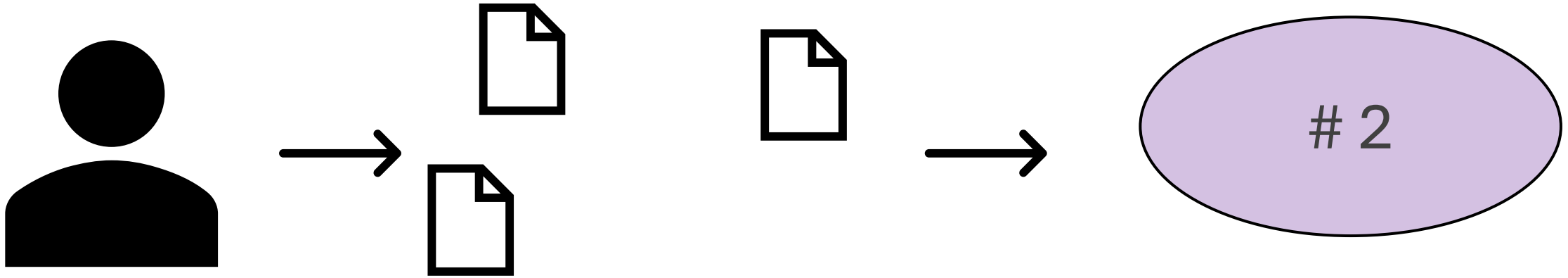
2005



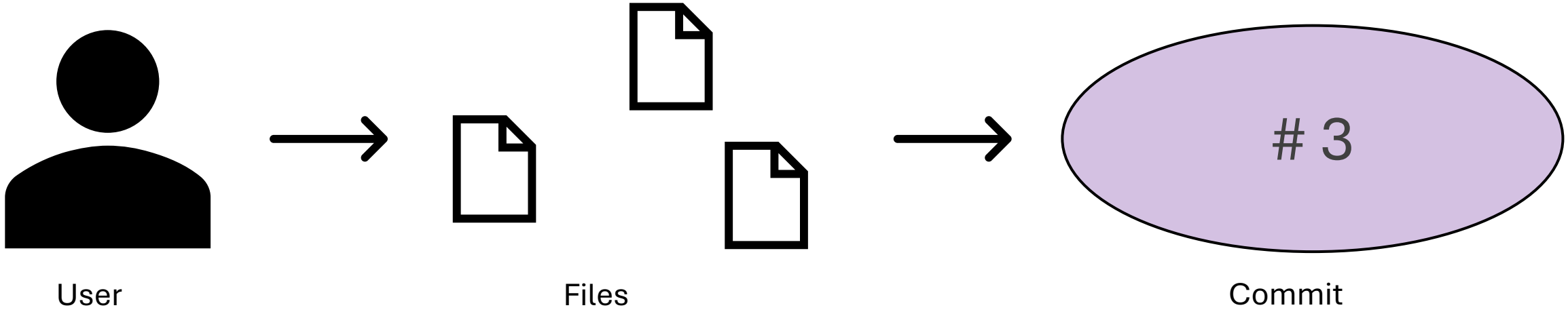

We do not develop software
We just build Power BI reports
We are only 3 people
It‘s too complicated
Fabric is cloud, anyway
I don‘t use the command line

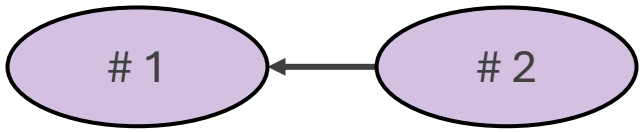

User
Files

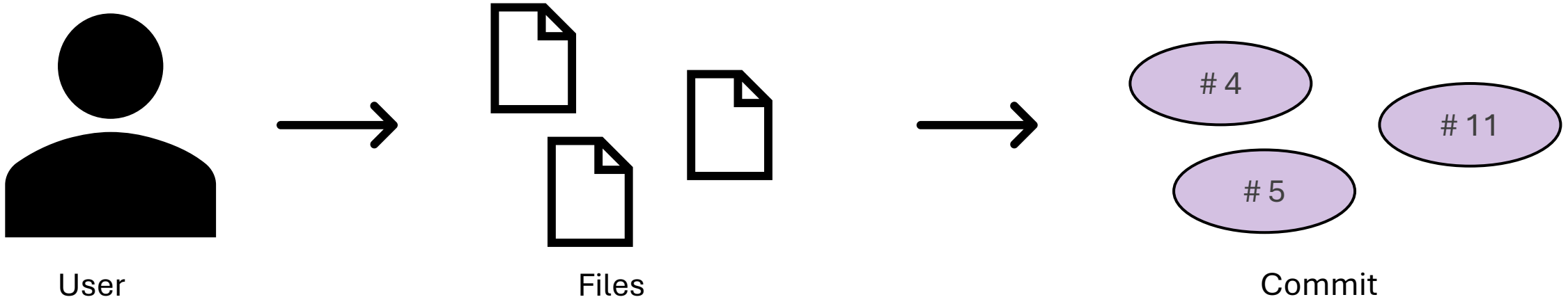

1
User
Files
Commit

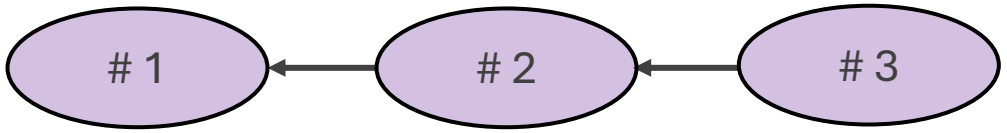

2
User
Files
Commit

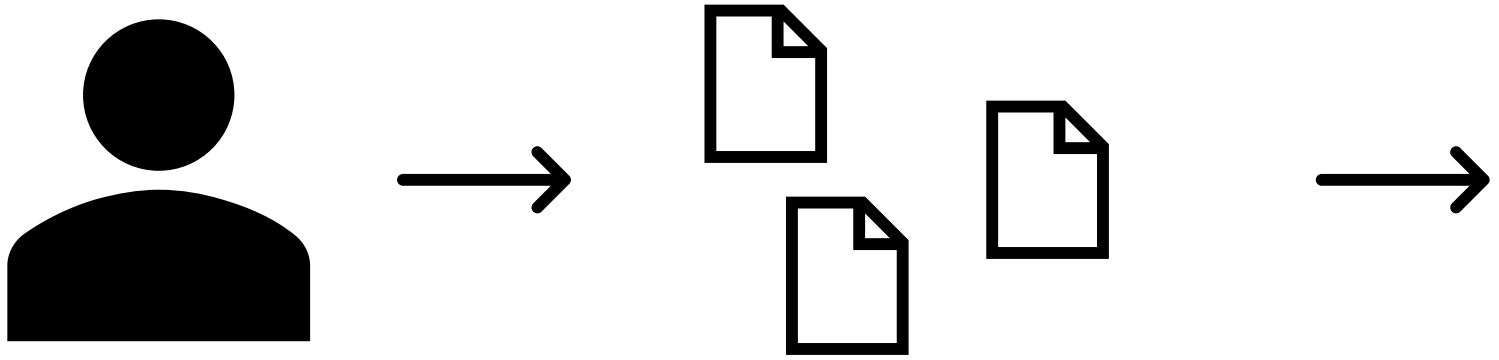

1

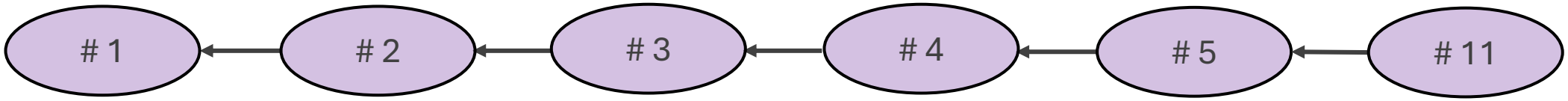

3
User
Files
Commit

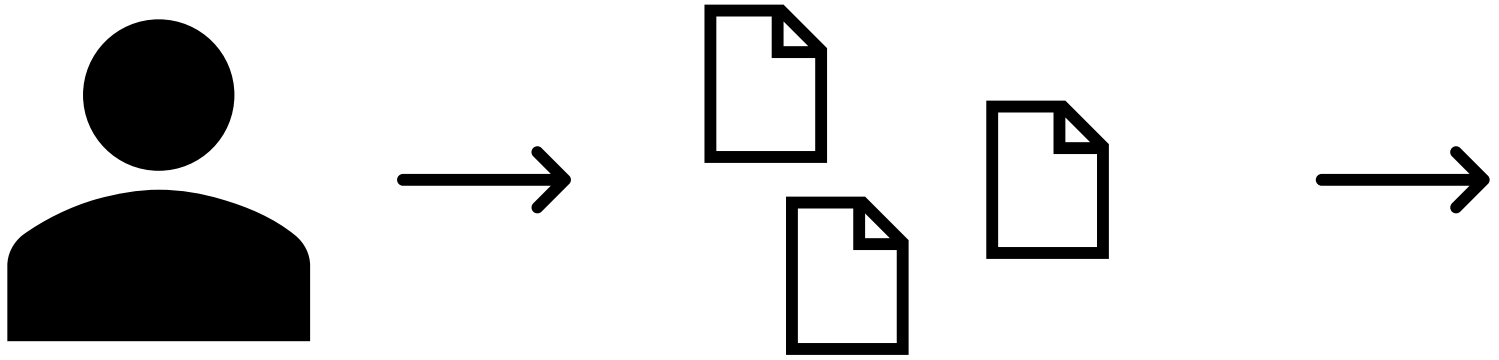

1
2

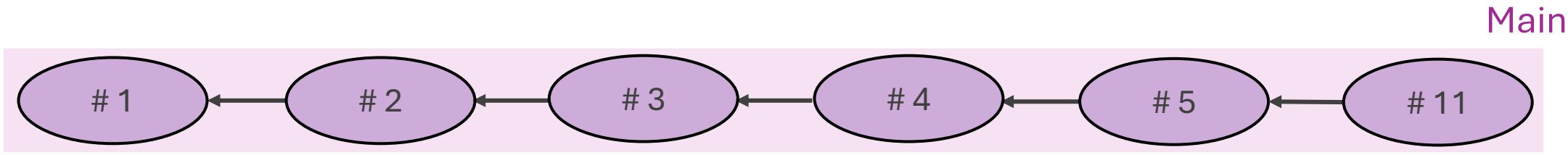

User
Files
4
11
5
Commit

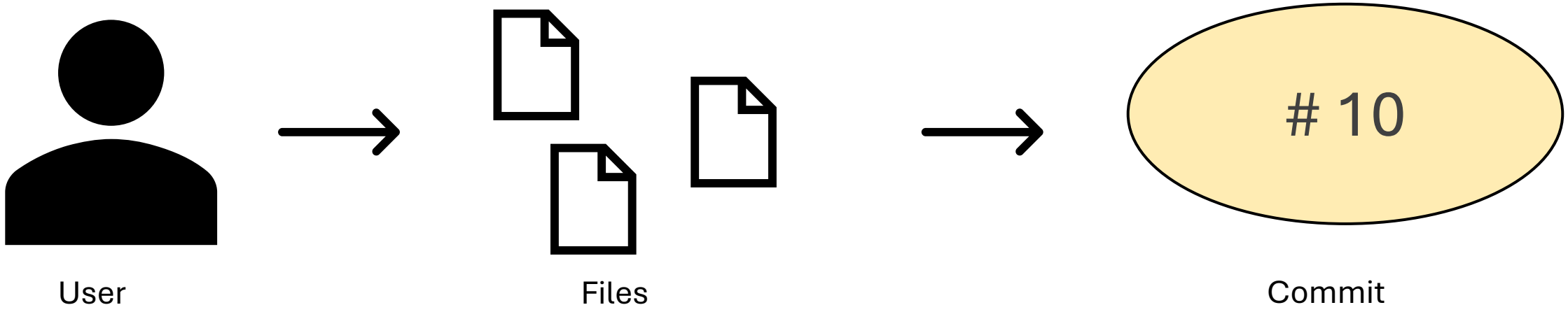

1
2
3

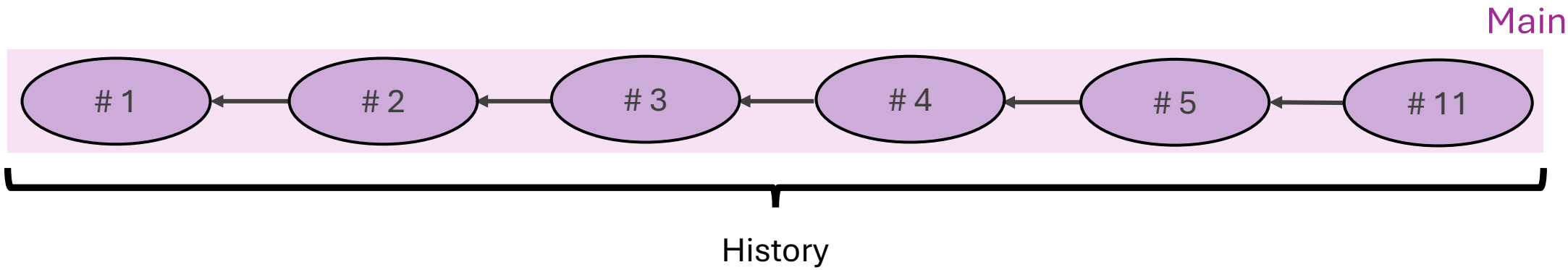

User
Files
Commit

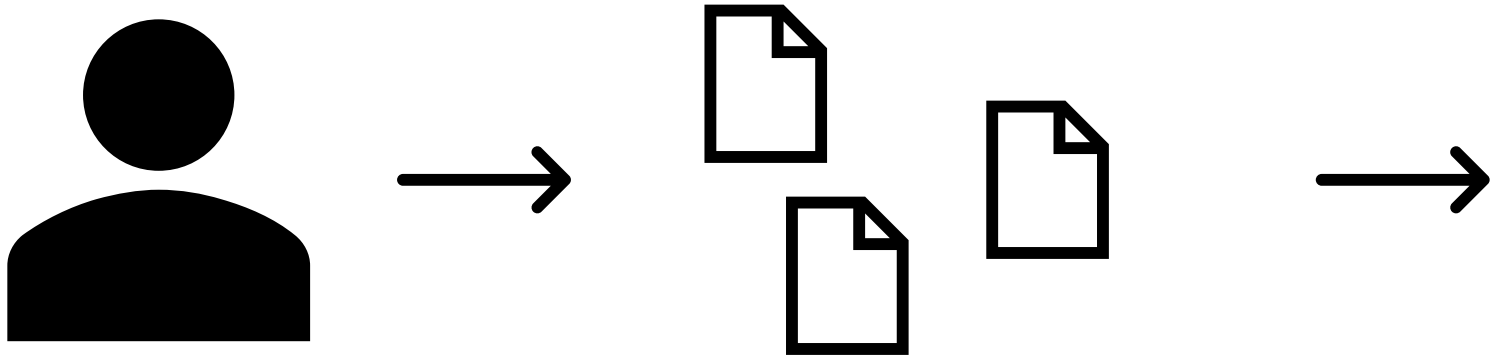

1
2
3
4
5
11

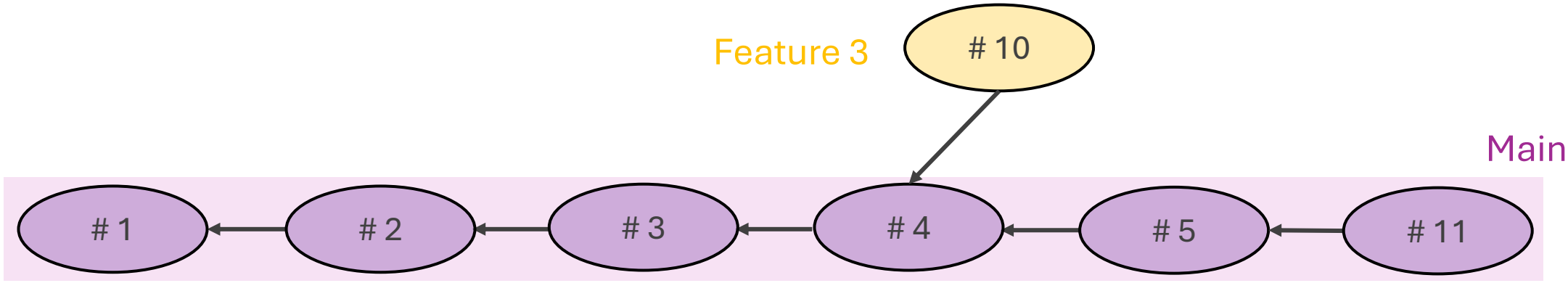

User
Files
Commit

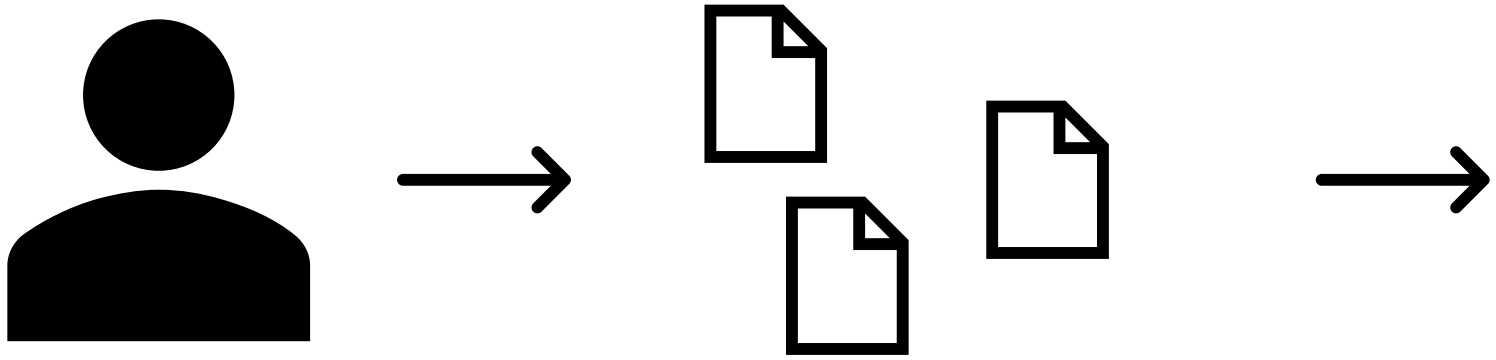

Main
1
2
3
4
5
11

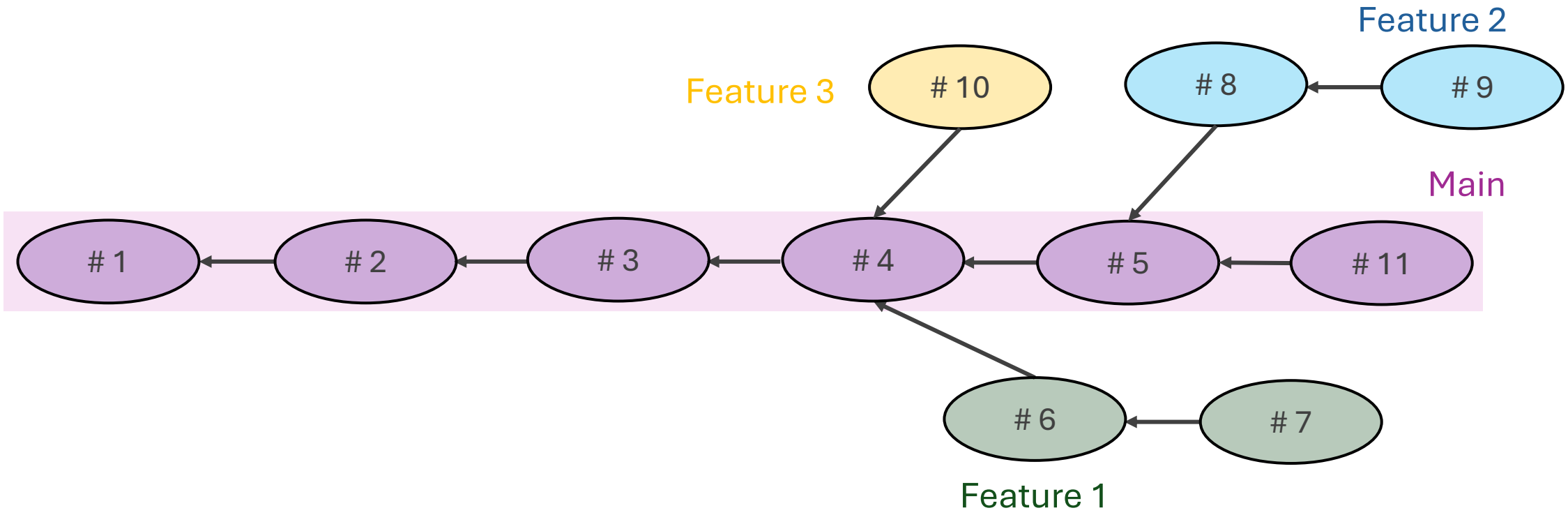

10
User
Files
Commit


Main
1
2
3
4
5
11
History


User
Files
Commit


Feature 3
10
Main
1
2
3
4
5
11


User
Files
Commit


Feature 2
Feature 3
10
8
9
Main
1
2
3
4
5
11
6
7
Feature 1

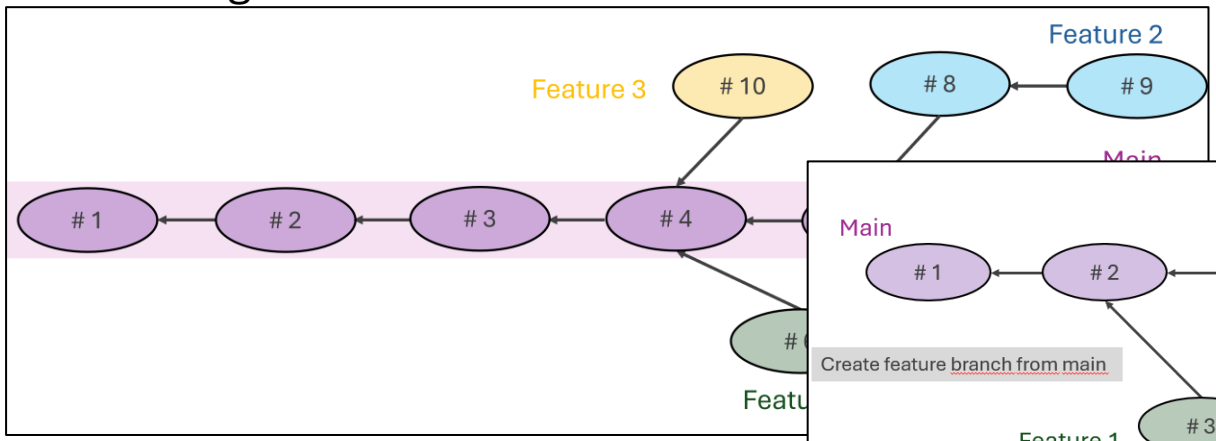

git
Versioning
Enables collaboration
Source of Truth
Consistency
Immutable history

Starting out

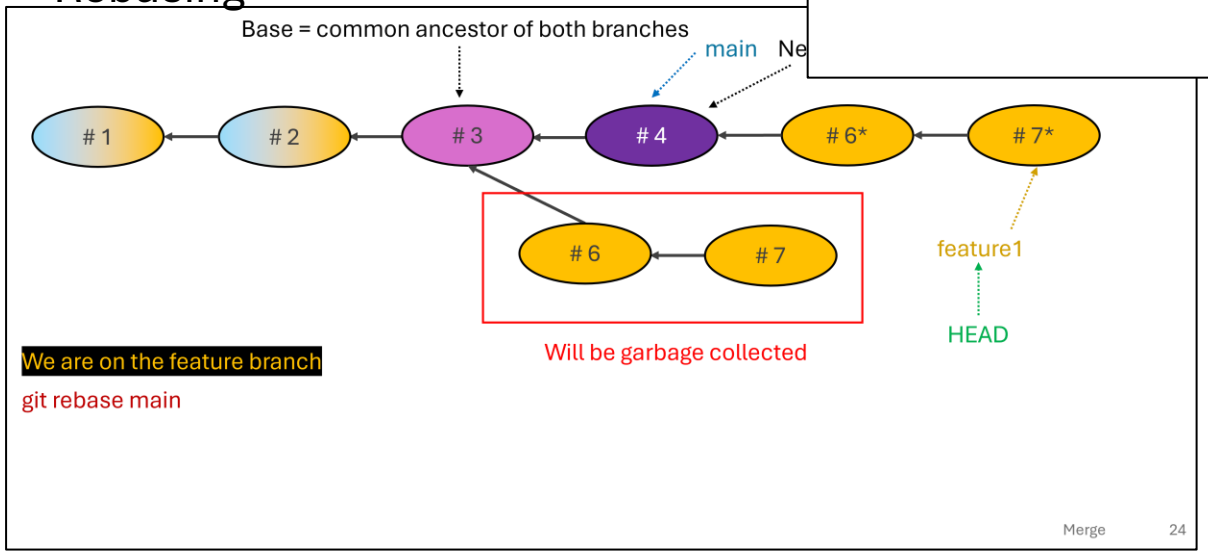
Great git course

- 01_Introduction
- 02_Repositories
- 03_BasicGitWorkflow
- 04_Undo
- 05_Branching
- 06_Merging
- 07_gitlog
- 08_Handling_commits
- 09_rebasing
- 10_Fabric_Workflow
- 11_PowerBI_Workflow

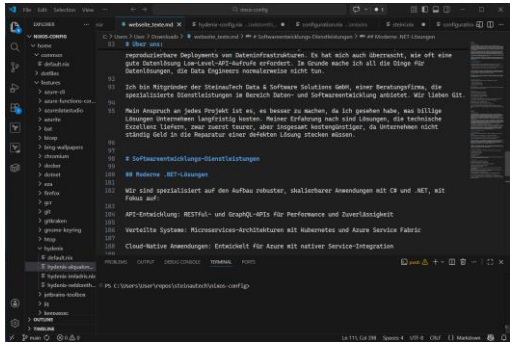
Branching



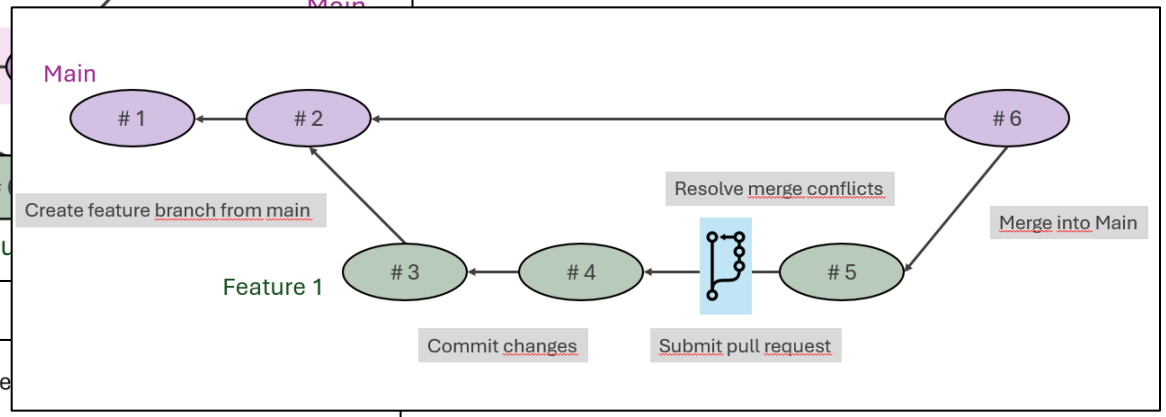
Rebasing



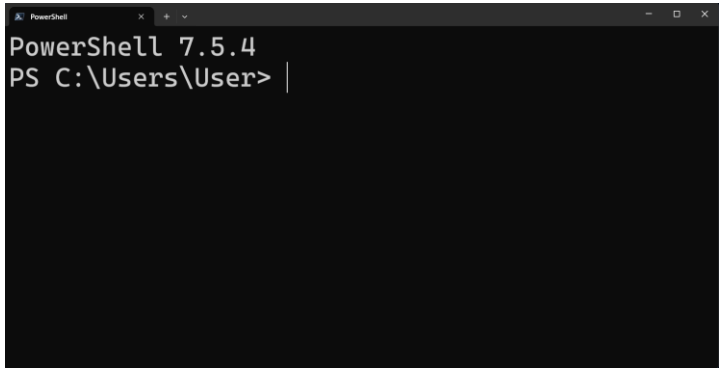
VS Code



Pull Requests



Command Line



We Thought We Were Ready

Why Version Control

Git Workflow

Reset

Merging

Git Areas

Command Line

Revert

Pull Request

Repositories

Visual Studio Code

Git Log

TMDL

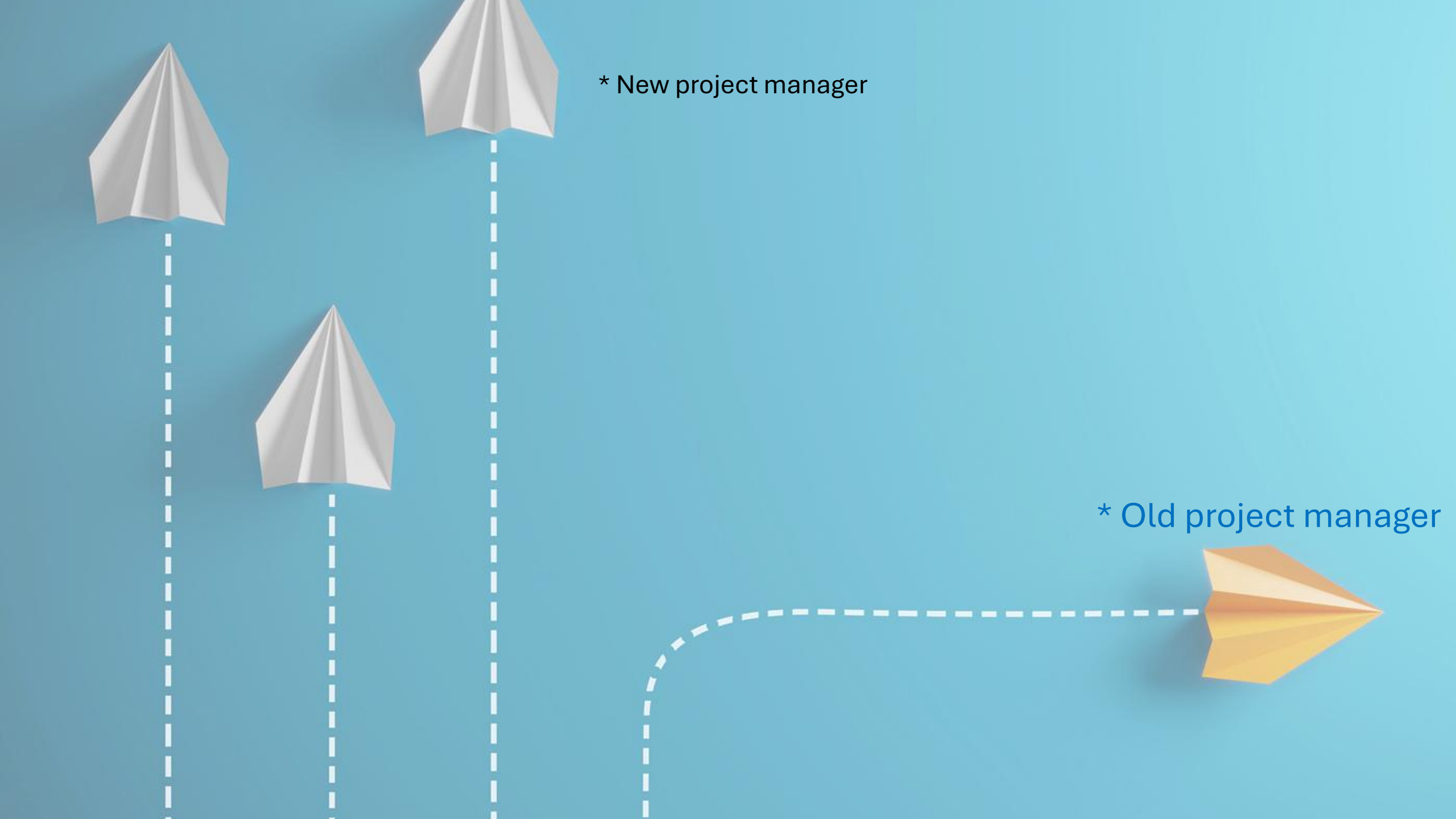
Branching

Rebase

Fabric Git Integration


* Not the actual project manager


Rebase


* New project manager
* Old project manager


KEEP
THINGS
SIMPLE


PowerShell 7.5.4
PS C:\Users\User>



And Then It Failed.


Technical Tools Are Not Enough

“We’ve Always Done It This Way”



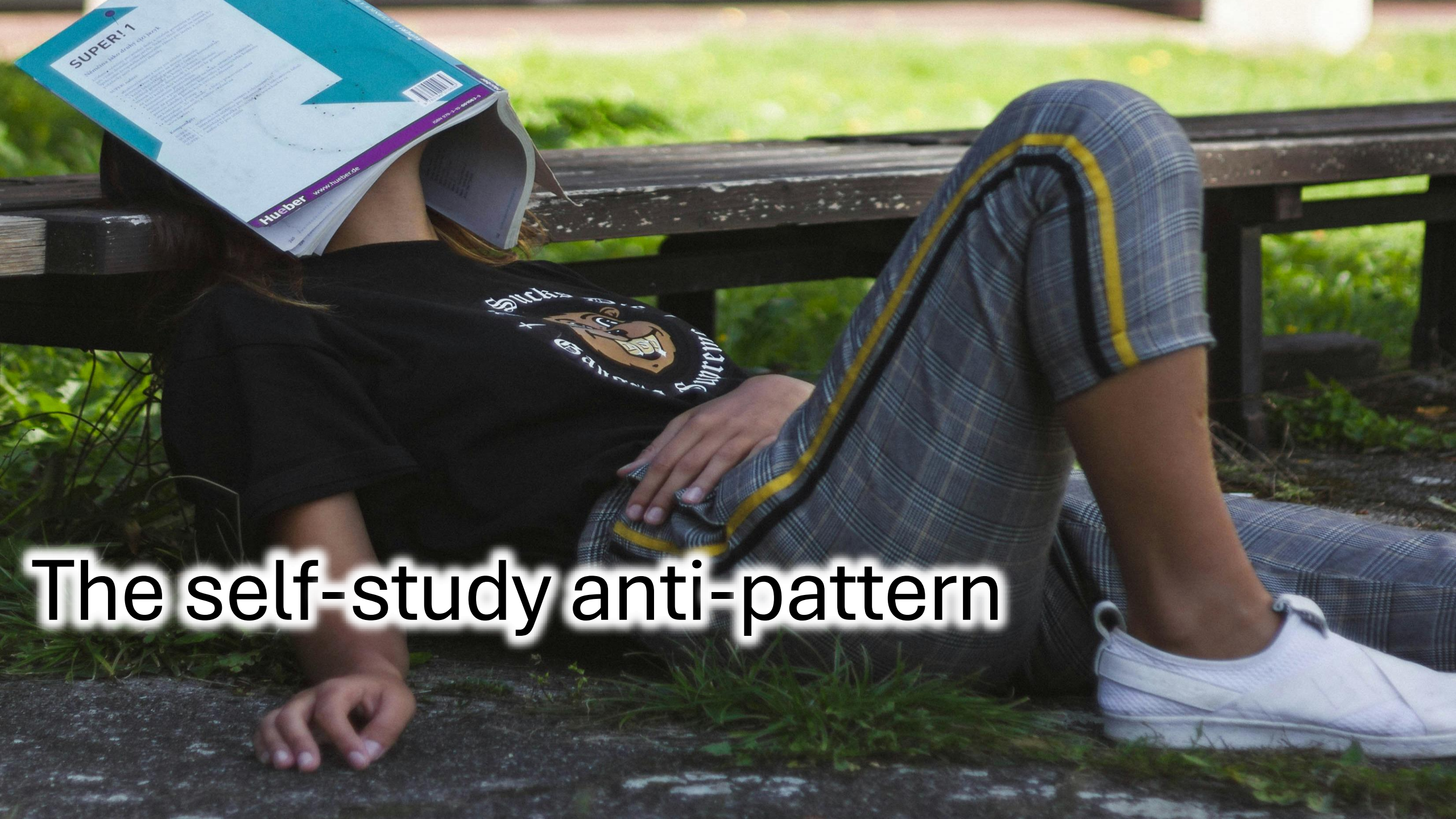


Adoption needs critical mass

Experimental evidence for tipping points in social convention | Science



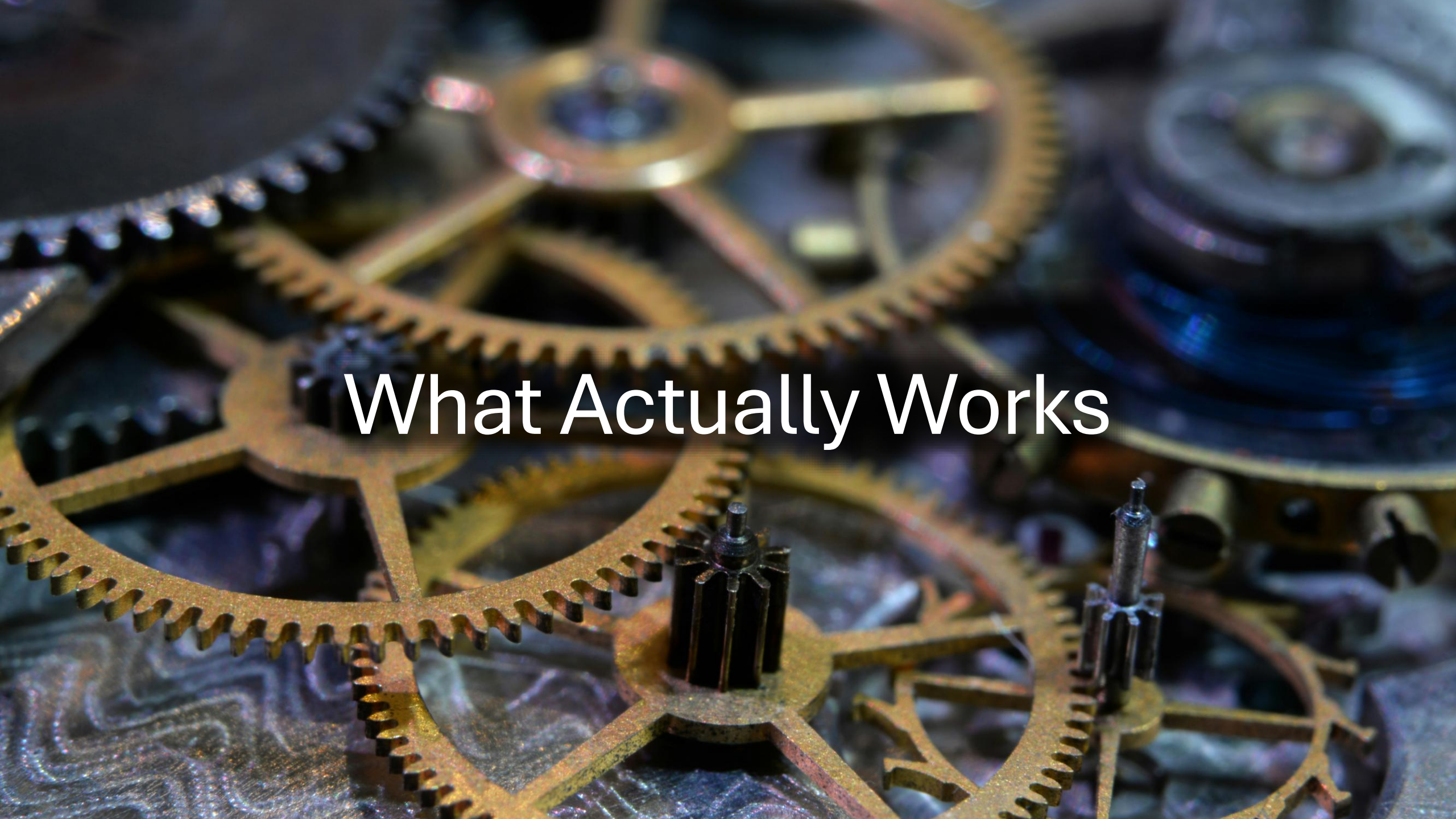
Account for time



The self-study anti-pattern

UI tools vs. Command line

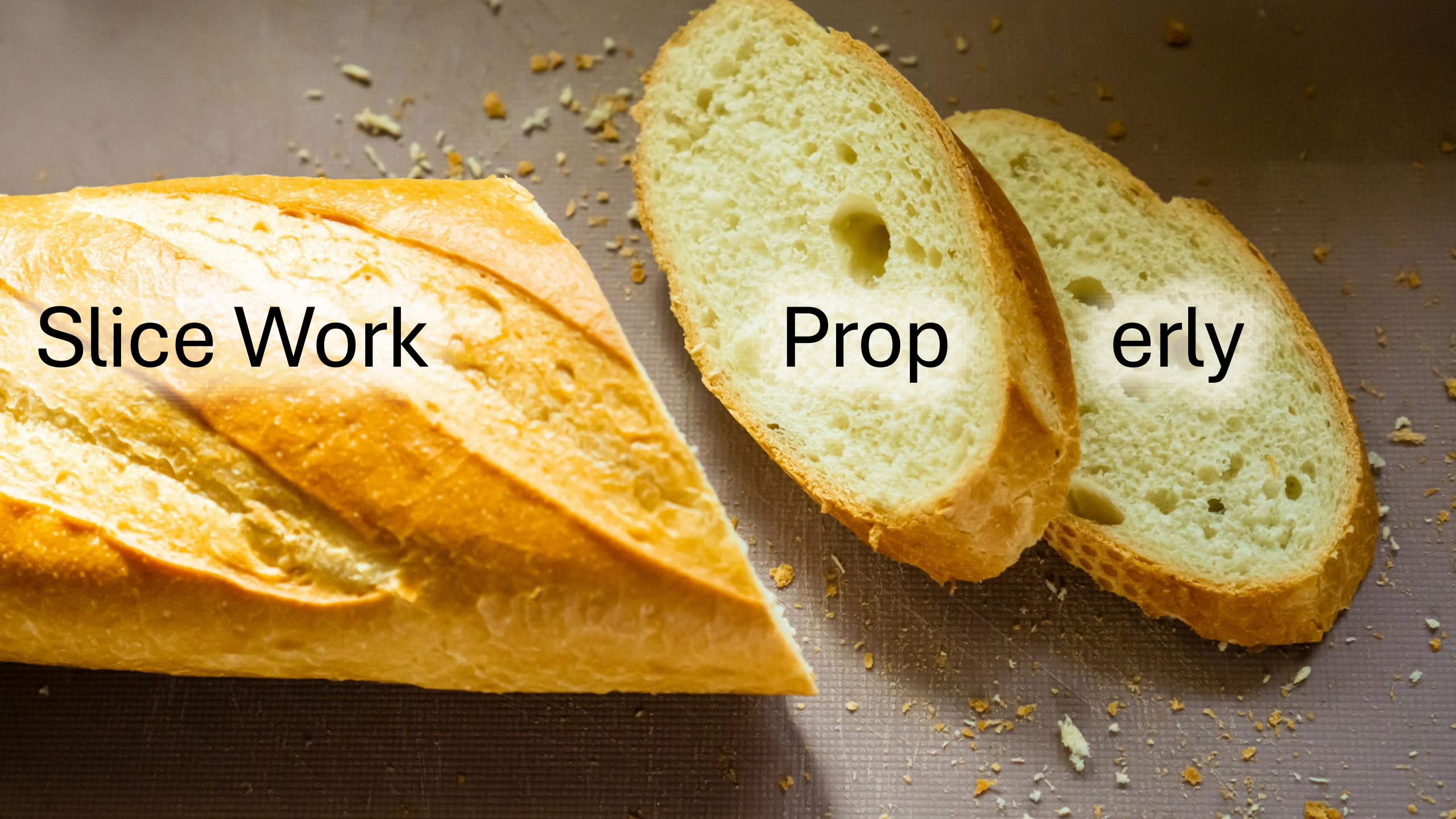




What Actually Works

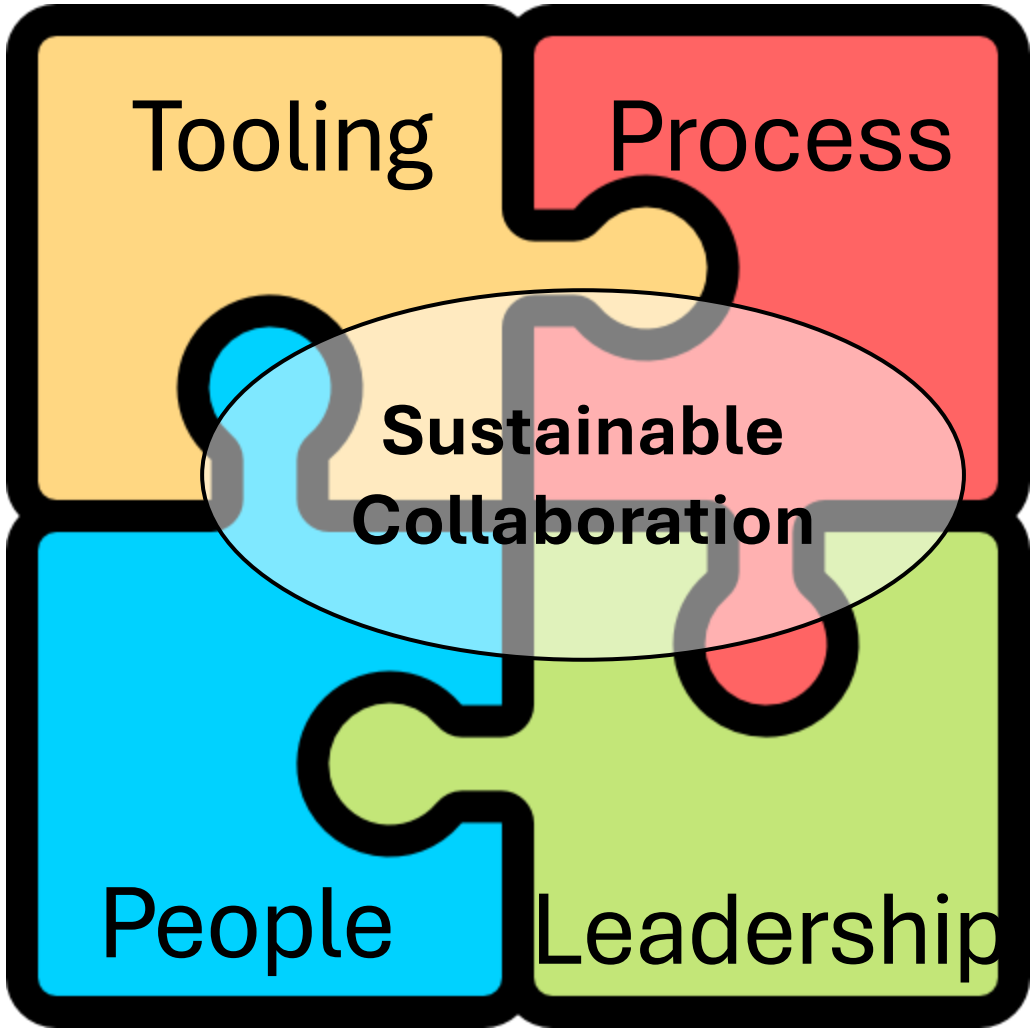

Reduce
Bottlenecks


Empower the
Right People


Slice Work
Prop
erly

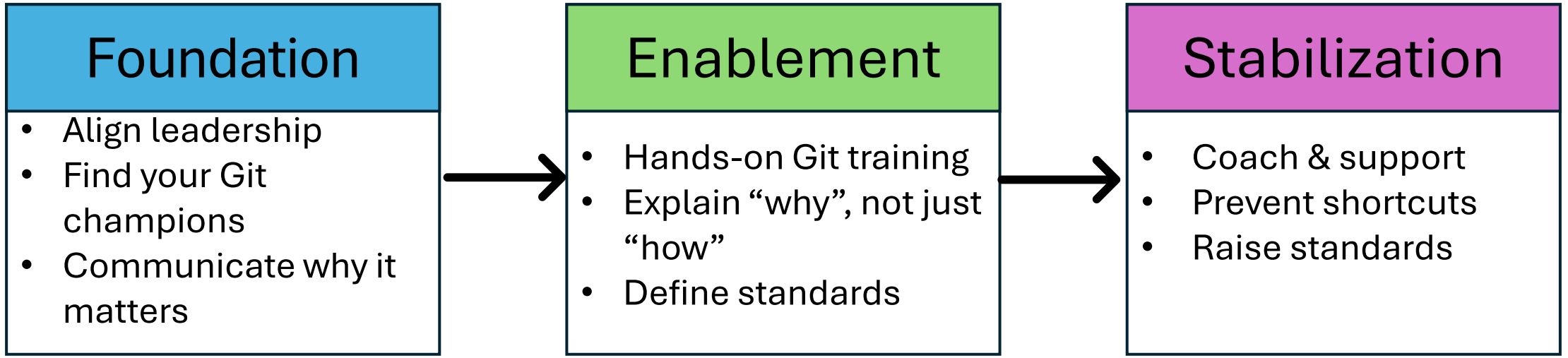

Collaboration is safe

Git Adoption



Git works when culture supports it. Otherwise, it becomes just another unused feature.

Roadmap for successful Git adoption



The biggest Git mistake is believing the problem is technical.
The real challenge is alignment, ownership and mindset.

Feedback is kindly appreciated

