The Database DevOps (CI/CD) Showdown

PASS24
Data Community Summit

VS

Tonie Huizer

He/him

Olivier van Steenlandt

He/him

PASS 24
Data Community Summit

ROUND 3
Development Practices

redgate

State-based Deployment

Overview

Deploy to Development

- How to deploy?
- Build / Deploy
- Publishing Profiles
- Advanced Configuration

Lab: Deploy your changes

Manual Deployment

- SQL Server Data Tools (GUI)
- Command Line

Build / Deploy Phase

Build

- Validation of code (references, syntax,…)
- Copy build result based on configuration
- Result: .dacpac file

Deploy

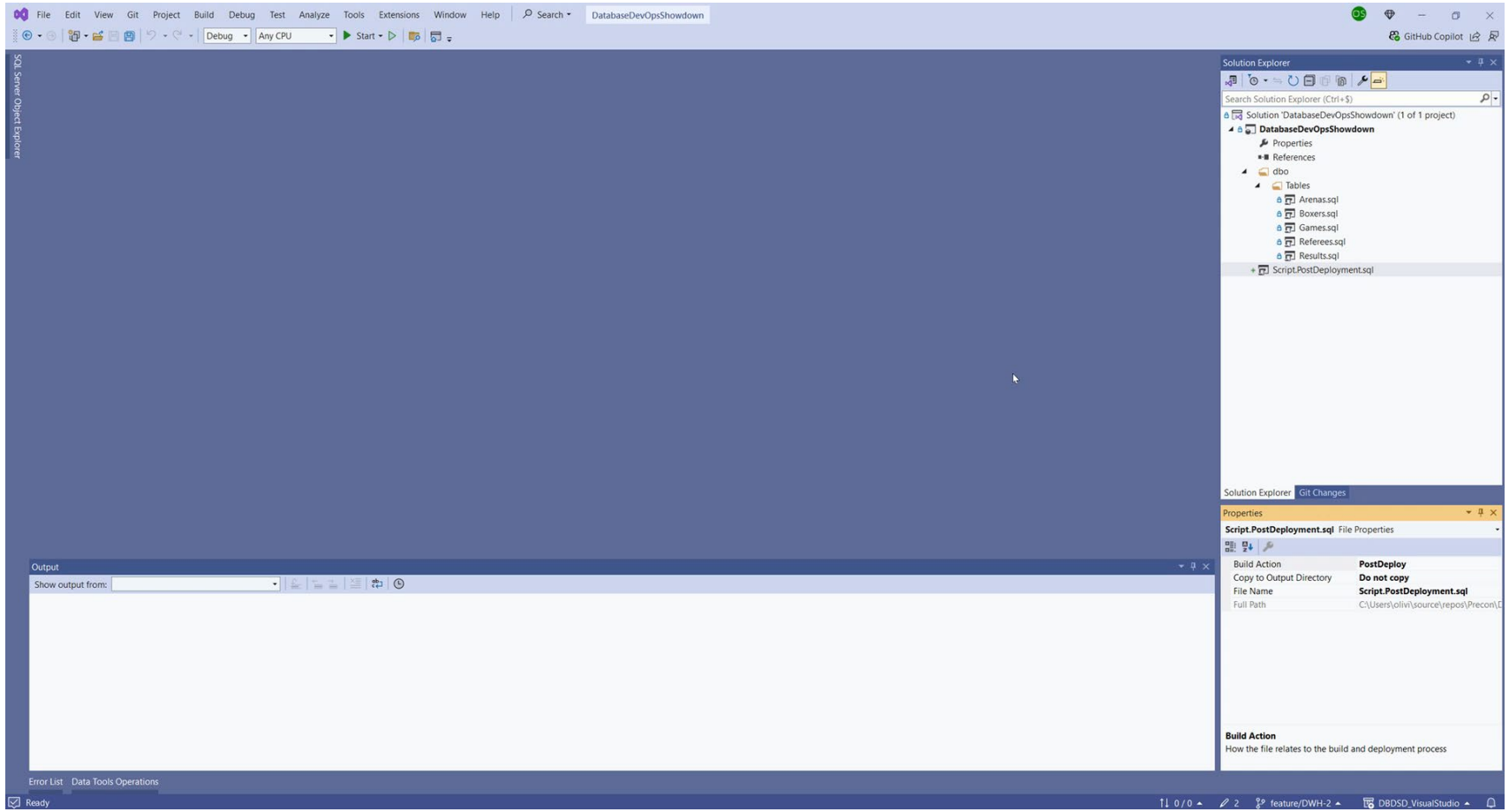
- Input: .dacpac file

MSBuild (Build Phase)

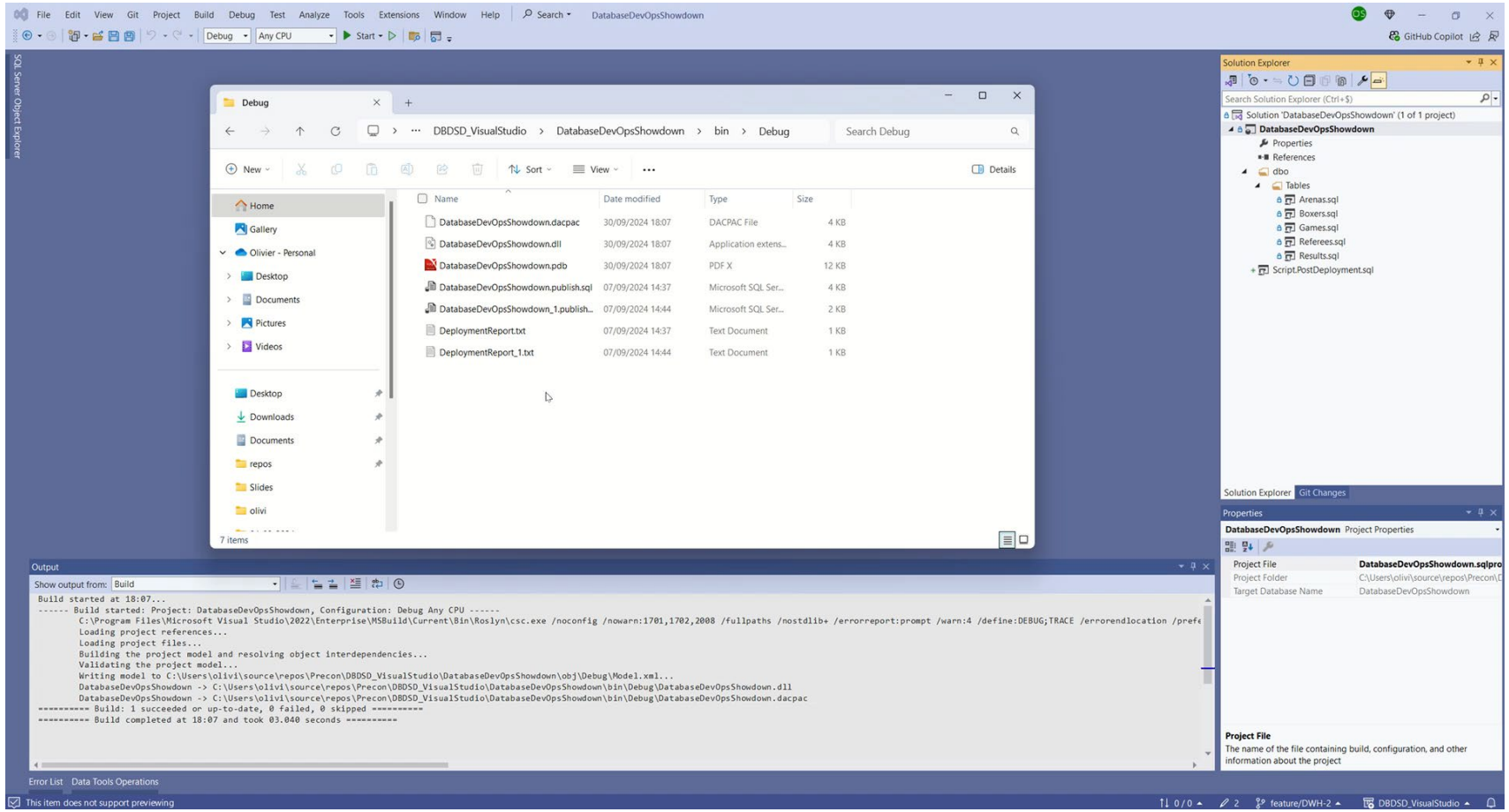
- MSBuild Engine
- Platform for building applications

- In Visual Studio (GUI)
- MSBuild.exe
- dotnet build

MSBuild In Visual Studio (Build Phase)



.dacpac



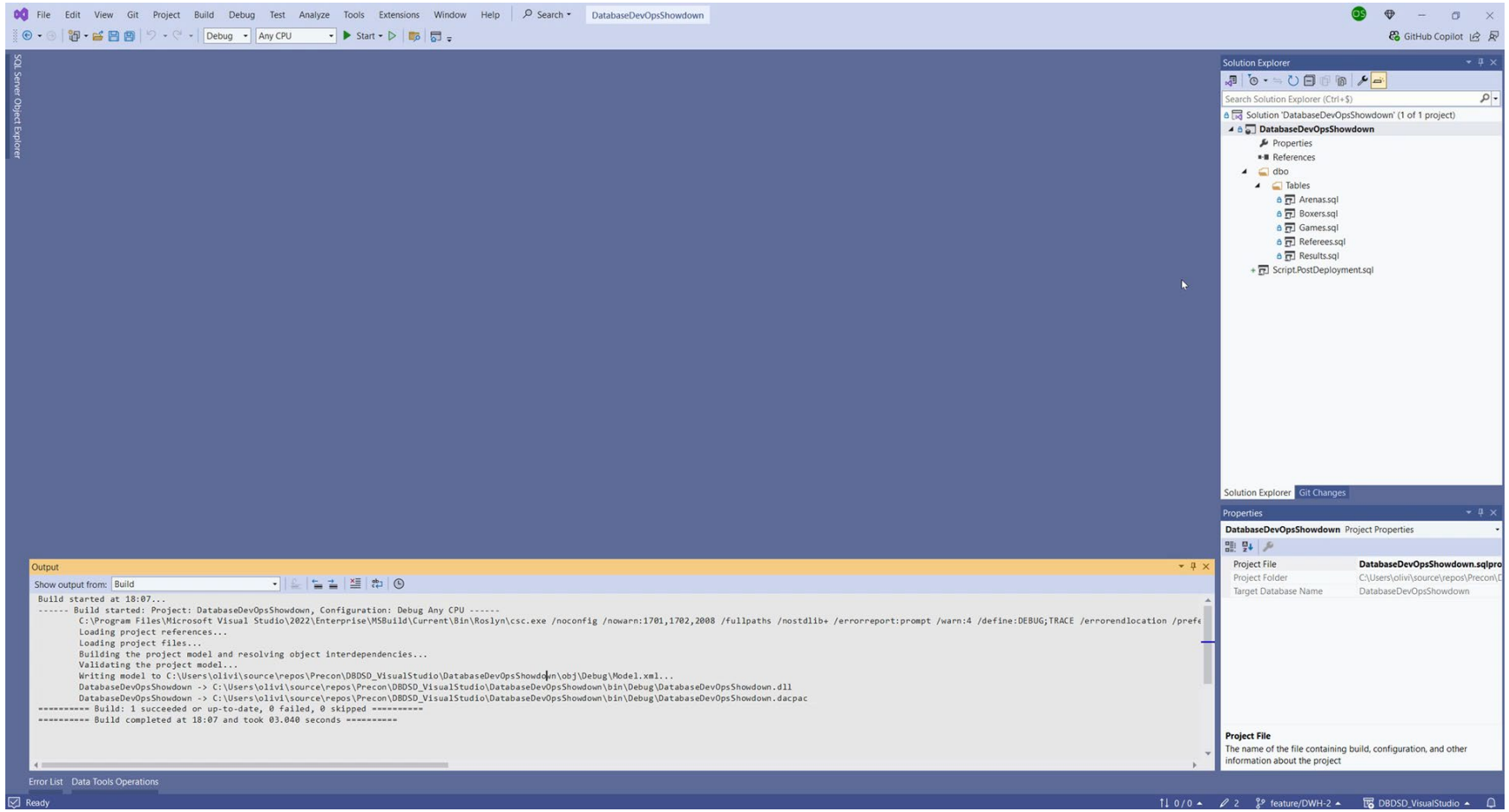
SqlPackage / DacFx (Deployment Phase)

- SqlPackage
  - CLI for DacFx Framework
  - Mechanism for database deployment
  - .dacpac / .bacpac
- In Visual Studio
- SqlPackage.exe

SqlPackage / DacFx (Deployment Phase)

- SqlPackage available for:
  - SQL Server
  - Azure SQL
  - Azure SQL MI
  - Azure Synapse Analytics
  - Microsoft Fabric*

SqlPackage / DacFx (Deployment Phase)



Publishing Profiles

- XML-file
- Deployment Properties

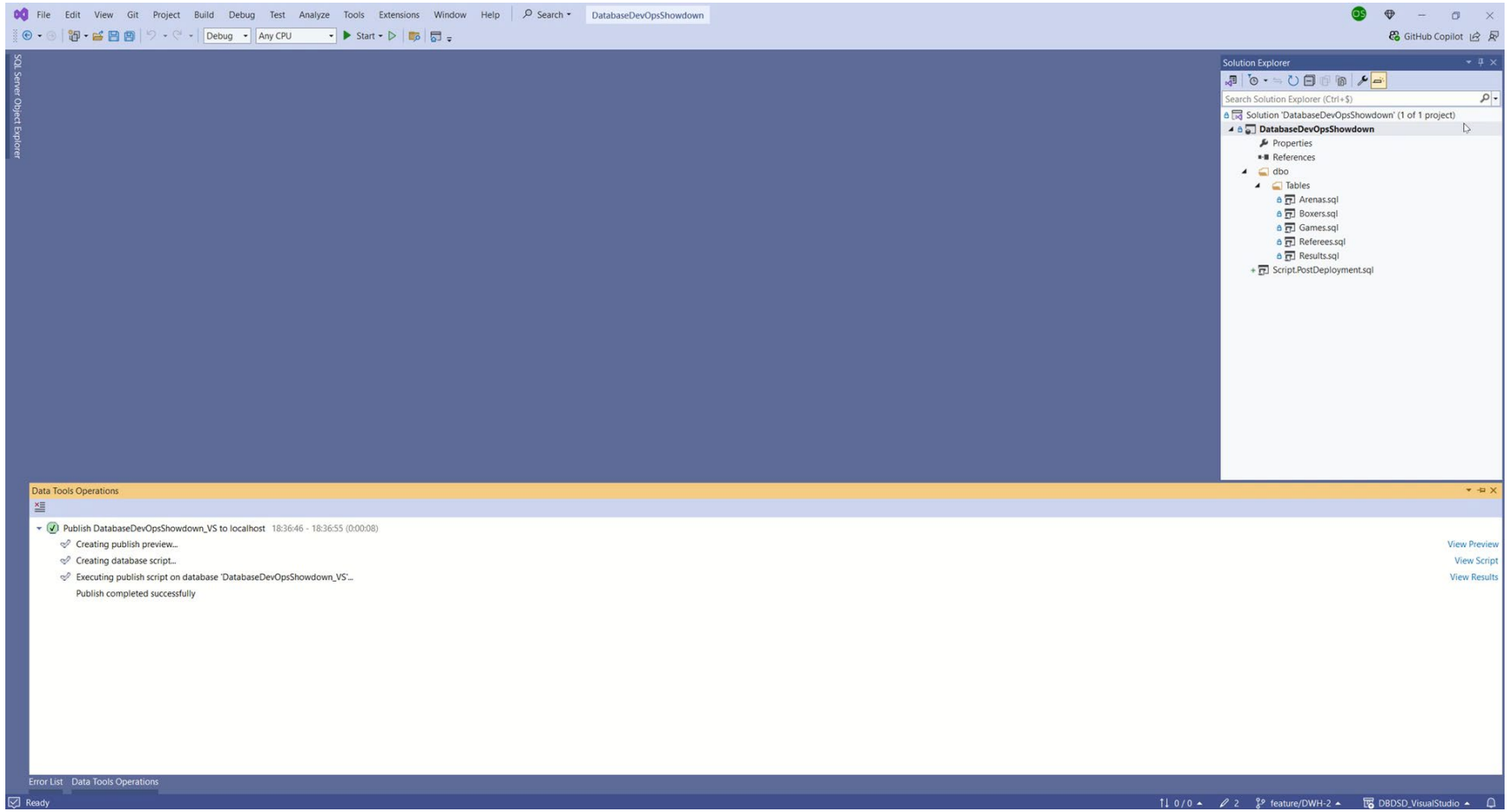
Advanced Deployment Configuration

- Variety of properties
- Most relevant:
  - Block if data loss…
  - Deploy database properties…
  - Drop indexes not in source…

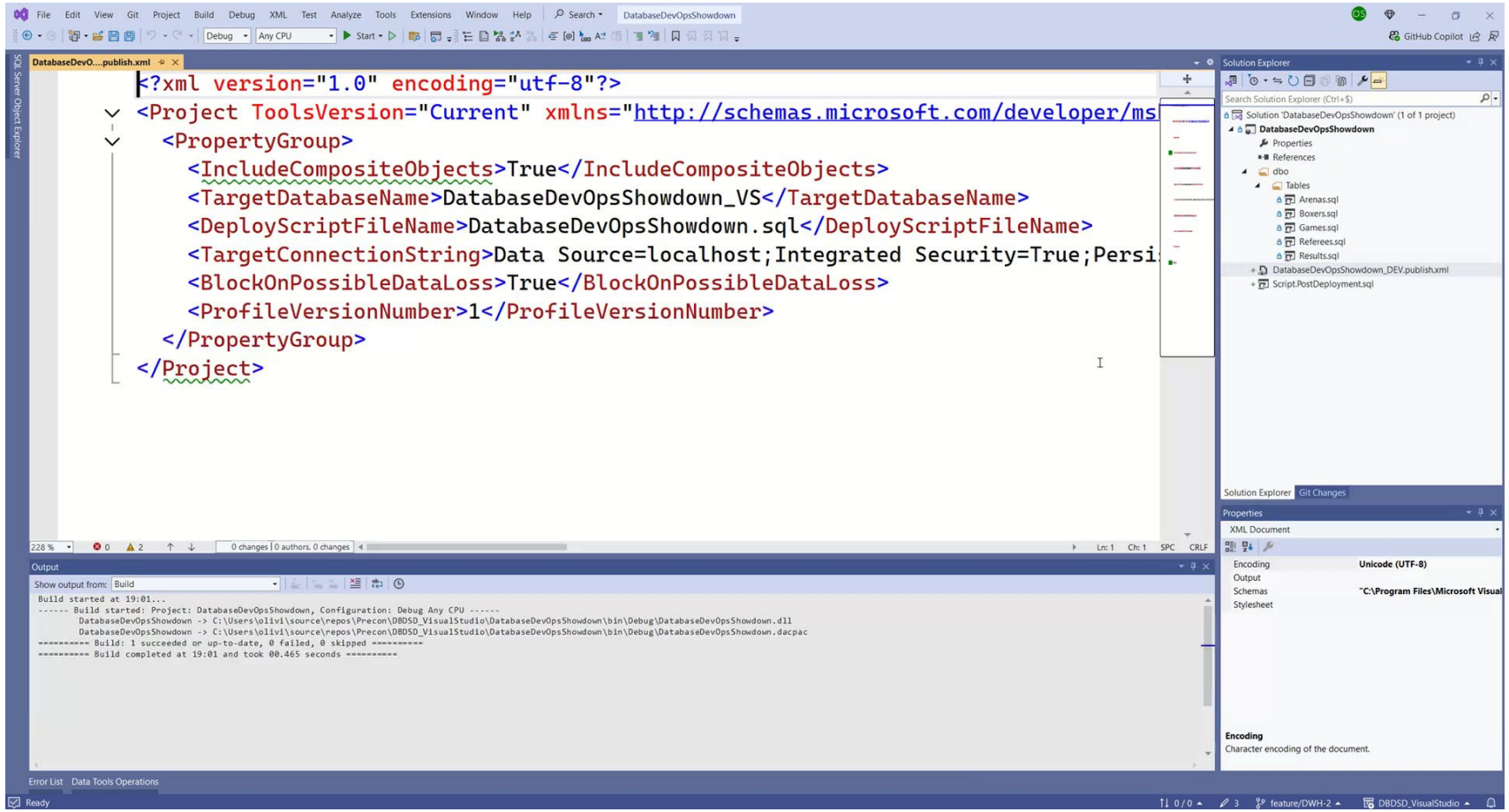
Tales from the trenches

- Deploy database properties…
- Drop indexes not in source…

Publishing Profile



Publishing Profile Usage



Lab

Deploy your changes

Person wrapping bandage photo – Free United kingdom Image on Unsplash

Migration-based Deployment

Overview

Deploy to Development

- How to deploy?
- Build vs Deploy
- Deploy
- Advanced Configuration

Lab 3: Deploy your changes

How to manual Deploy

- Flyway Desktop
- Flyway Command Line
  - Grate
  - EF Core

Manual Build vs Deploy Phase

Build

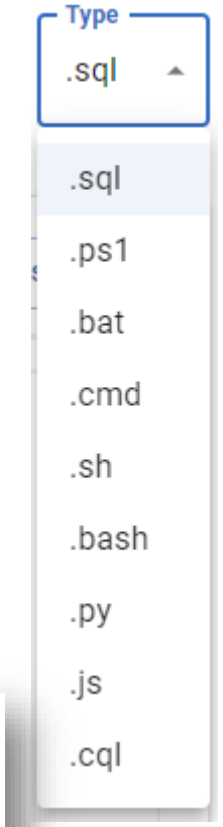
- Flyway Desktop doesn't build
- **Immutable** migrations already created
  - in dev phase (round 2)

Deploy

- Input: migrations files (immutable)

The “pre” build step for Flyway Enterprise

- Create a migration
  - With Flyway Desktop
  - Hand crafted
    - SQL and beyond
  - Migration types



Migration Type	Category	Example
SQL	Versioned	V2__Add_new_table.sql
SQL	Versioned (Undo)	U2__Add_new_table.sql
SQL	Repeatable	R__Add_new_table.sql
Java	Versioned	V2__Add_new_table.java
Java	Repeatable	R__Add_new_table.java
Script	Versioned	V1__execute_batch_tool.ps1
Script	Repeatable	R__execute_batch_tool.ps1

The “build” step for Flyway Community / Grate

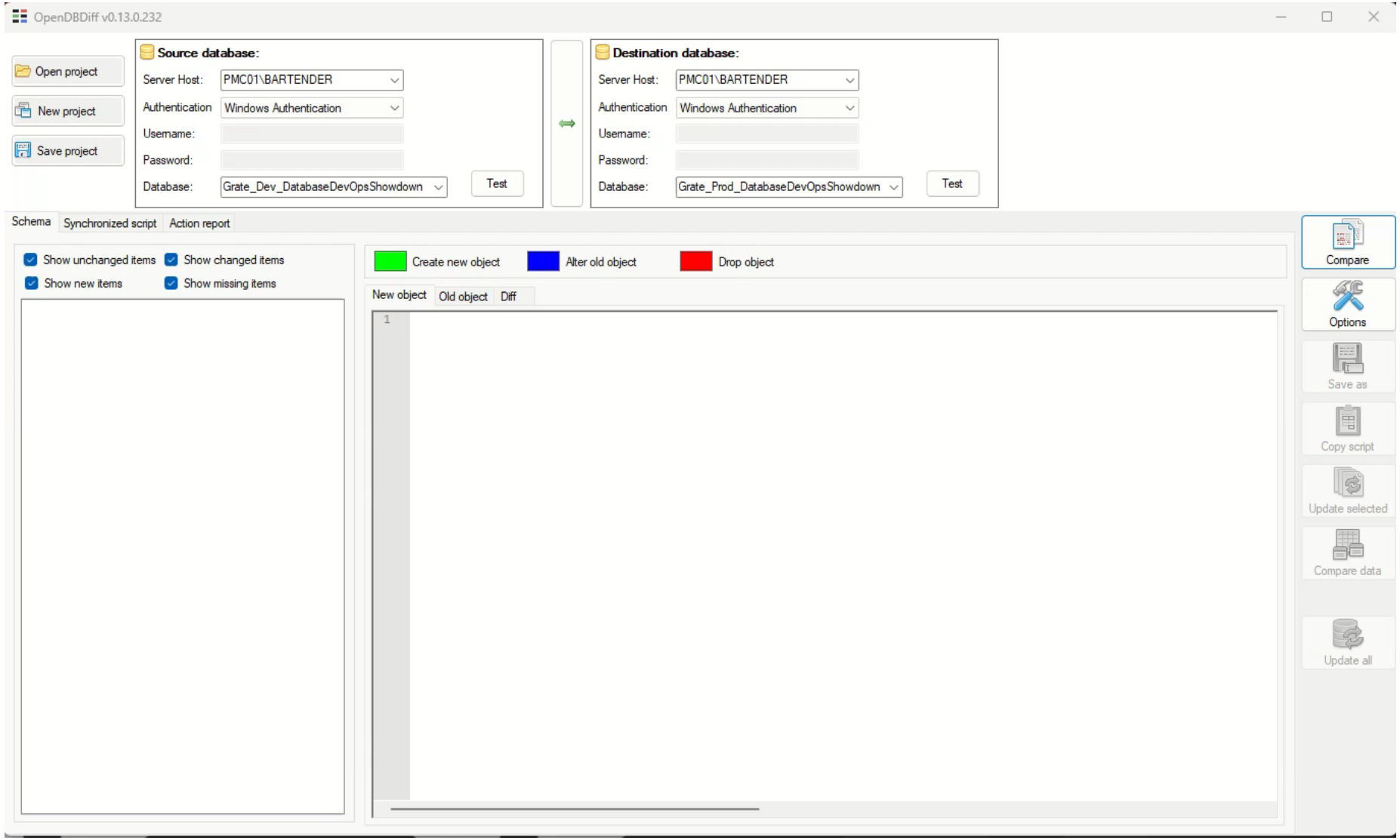
- Create a migration
  - Hand crafted
  - Via comparison tools; like OpenDbDiff
- Migration types

Migration Type	Category	Example
SQL	Versioned	V2__Add_new_table.sql
SQL	Versioned (Undo)	U2__Add_new_table.sql
SQL	Repeatable	R__Add_new_table.sql
Java	Versioned	V2__Add_new_table.java
Java	Repeatable	R__Add_new_table.java
Script	Versioned	V1__execute_batch_tool.ps1
Script	Repeatable	R__execute_batch_tool.ps1

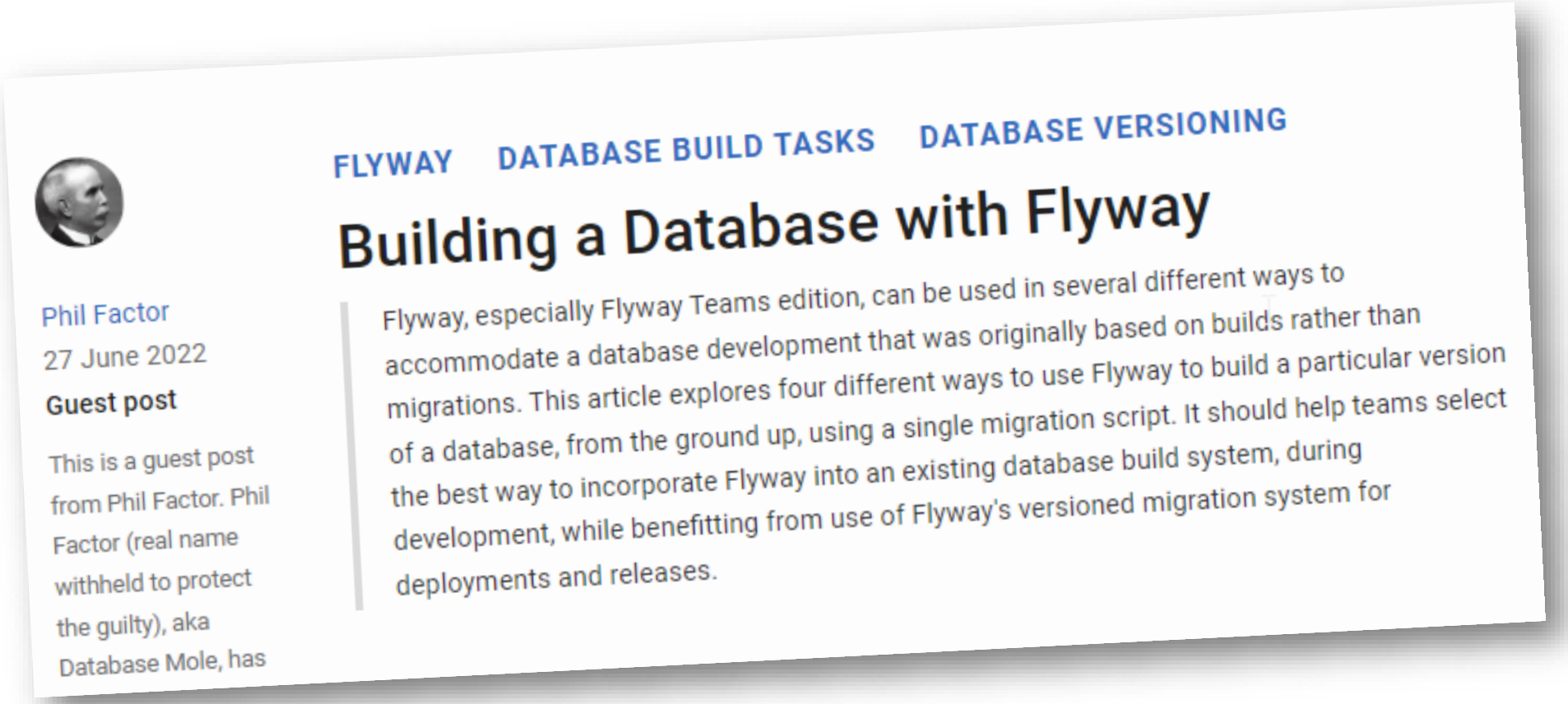
grate has three different types of scripts.

- Anytime Scripts
- Everytime Scripts
- One-Time Scripts

OpenDBDiff – Generate migration files



Alternative looks on building a Flyway DB



FLYWAY DATABASE BUILD TASKS DATABASE VERSIONING

Building a Database with Flyway

Phil Factor
27 June 2022
Guest post

This is a guest post from Phil Factor. Phil Factor (real name withheld to protect the guilty), aka Database Mole, has

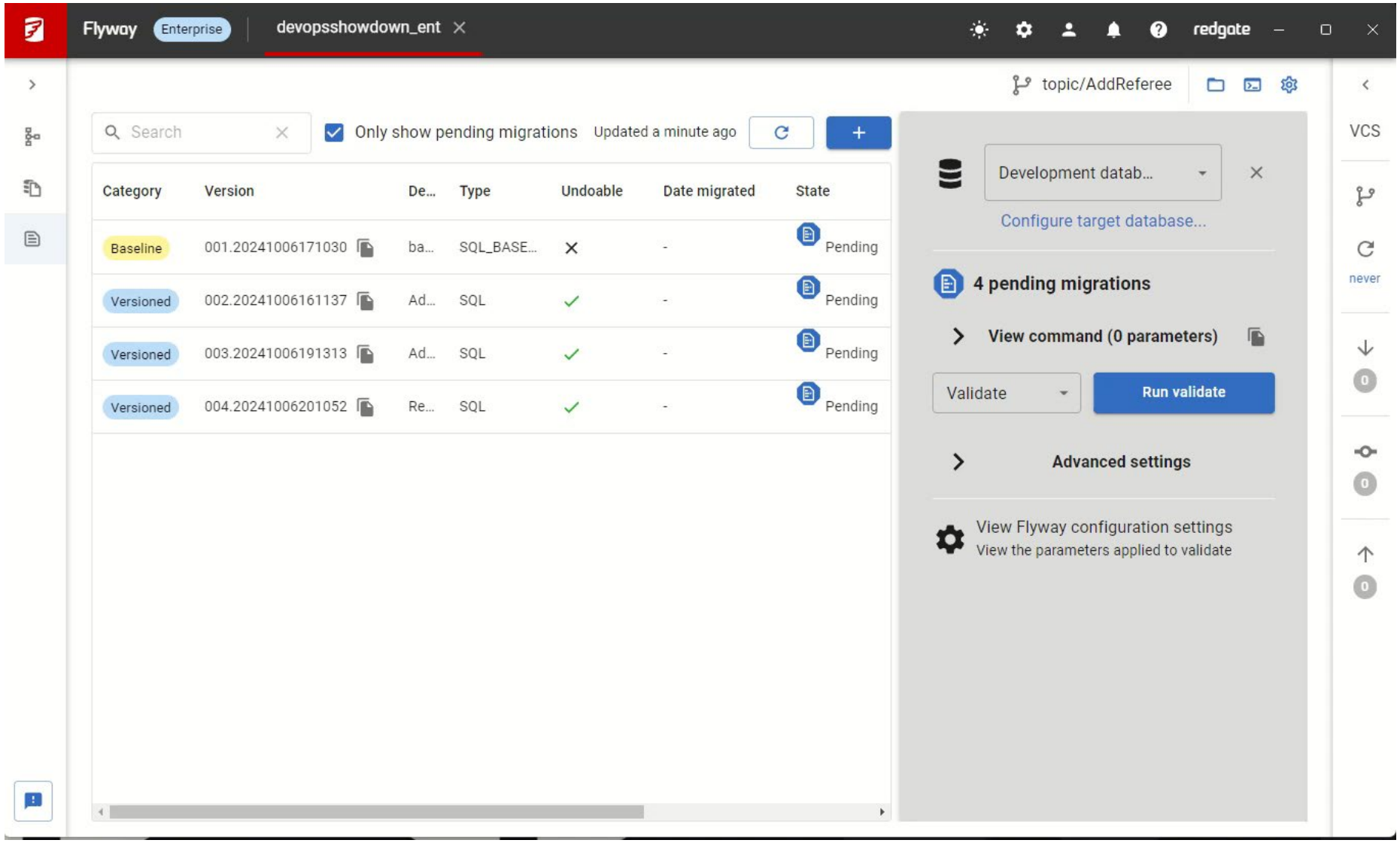
> Flyway, especially Flyway Teams edition, can be used in several different ways to accommodate a database development that was originally based on builds rather than migrations. This article explores four different ways to use Flyway to build a particular version of a database, from the ground up, using a single migration script. It should help teams select the best way to incorporate Flyway into an existing database build system, during development, while benefitting from use of Flyway's versioned migration system for deployments and releases.

Deploy

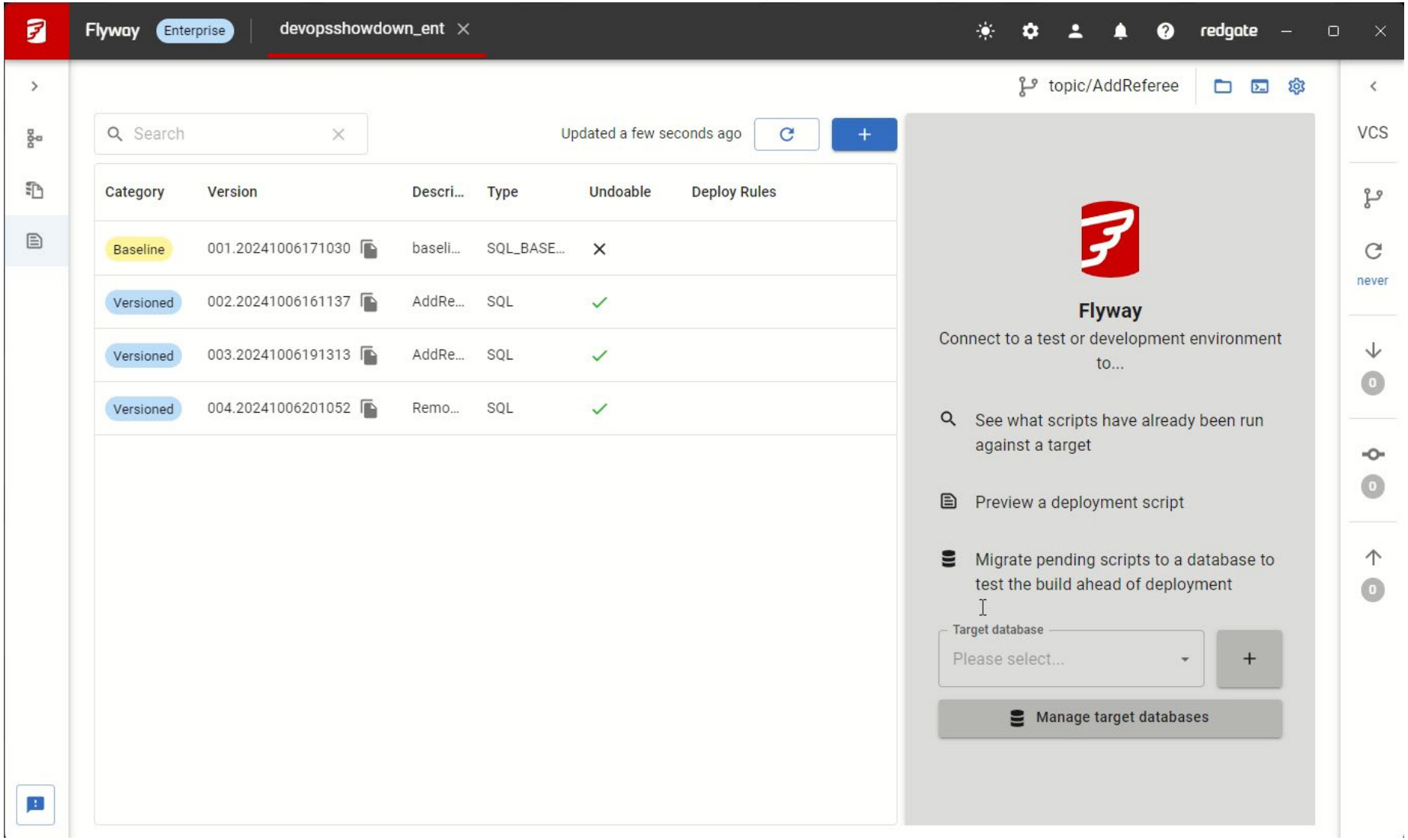
With Flyway

A man is doing a handstand on another man photo – Kung fu Image on Unsplash

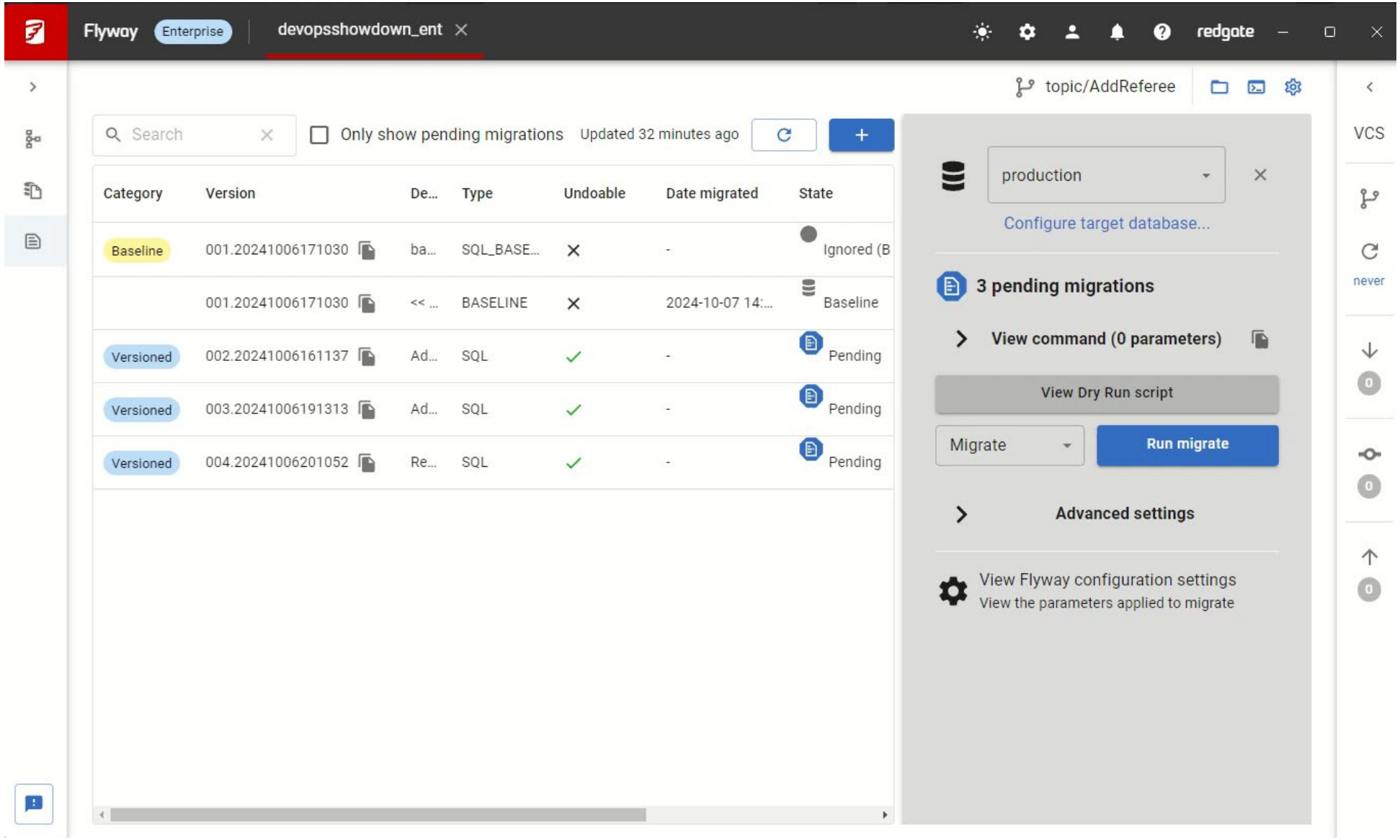
Setting up target environments



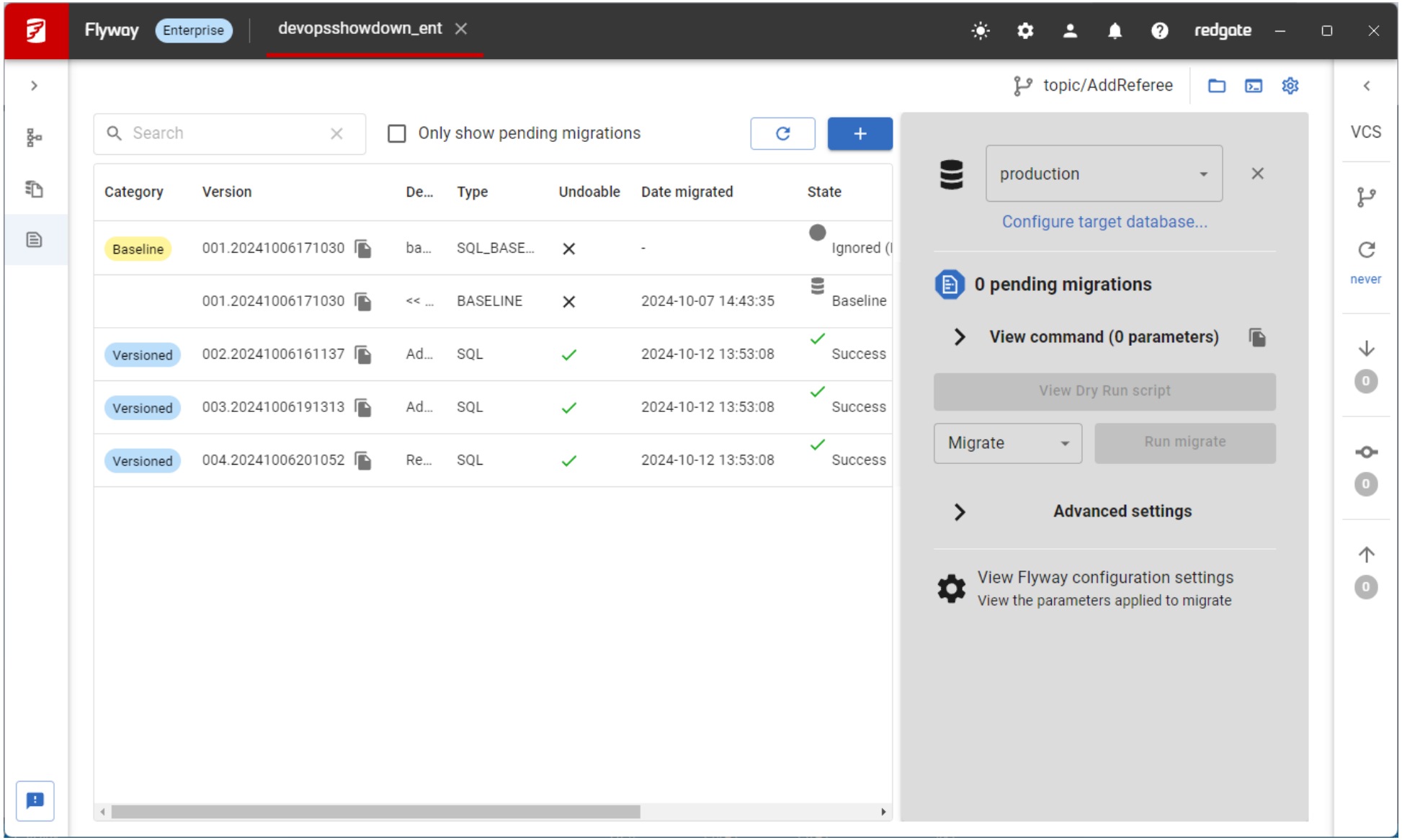
First “Activate” Flyway on production



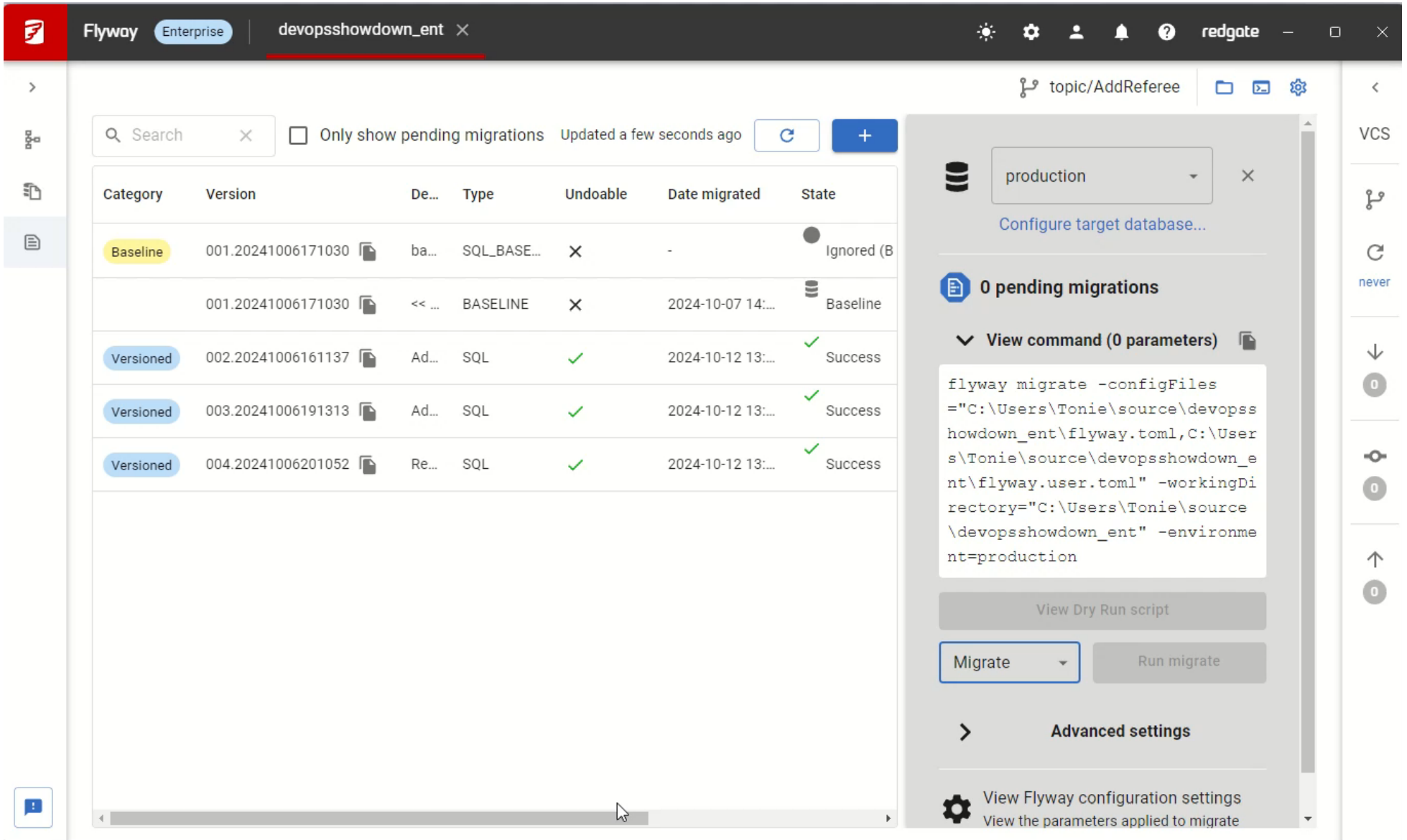
Migrate all


Flyway
Enterprise
devopsshowdown_ent
redgate
topic/AddReferee
Search
Only show pending migrations
Updated 32 minutes ago
Category
Version
De...
Type
Undoable
Date migrated
State
Baseline
001.20241006171030
ba...
SQL_BASE...
Ignored (B
001.20241006171030
<< ...
BASELINE
2024-10-07 14:...
Baseline
Versioned
002.20241006161137
Ad...
SQL
Pending
Versioned
003.20241006191313
Ad...
SQL
Pending
Versioned
004.20241006201052
Re...
SQL
Pending
production
Configure target database...
3 pending migrations
View command (0 parameters)
View Dry Run script
Migrate
Run migrate
Advanced settings
View Flyway configuration settings
View the parameters applied to migrate
VCS
never

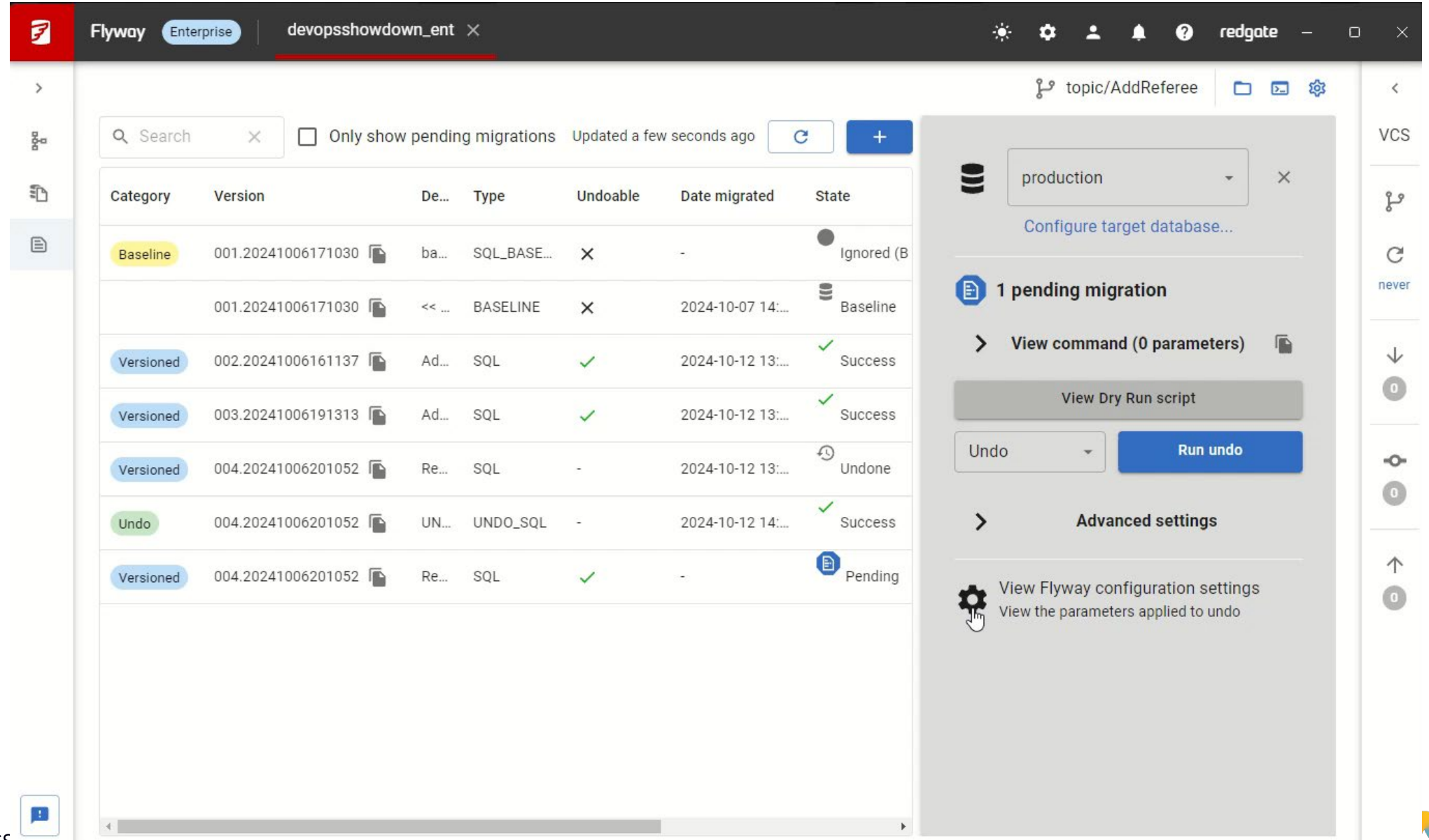
Migrate all – the result



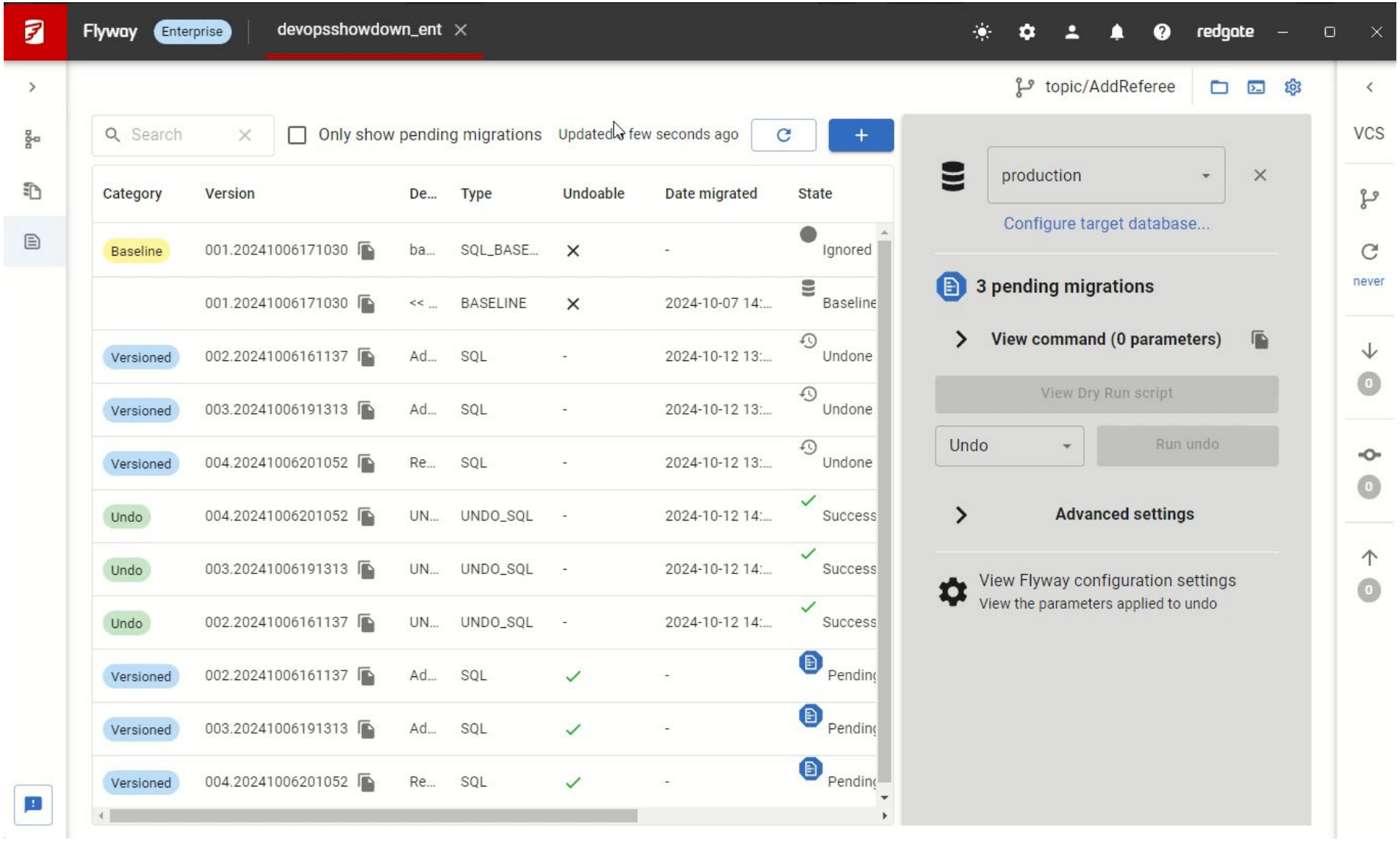
Undo a migration



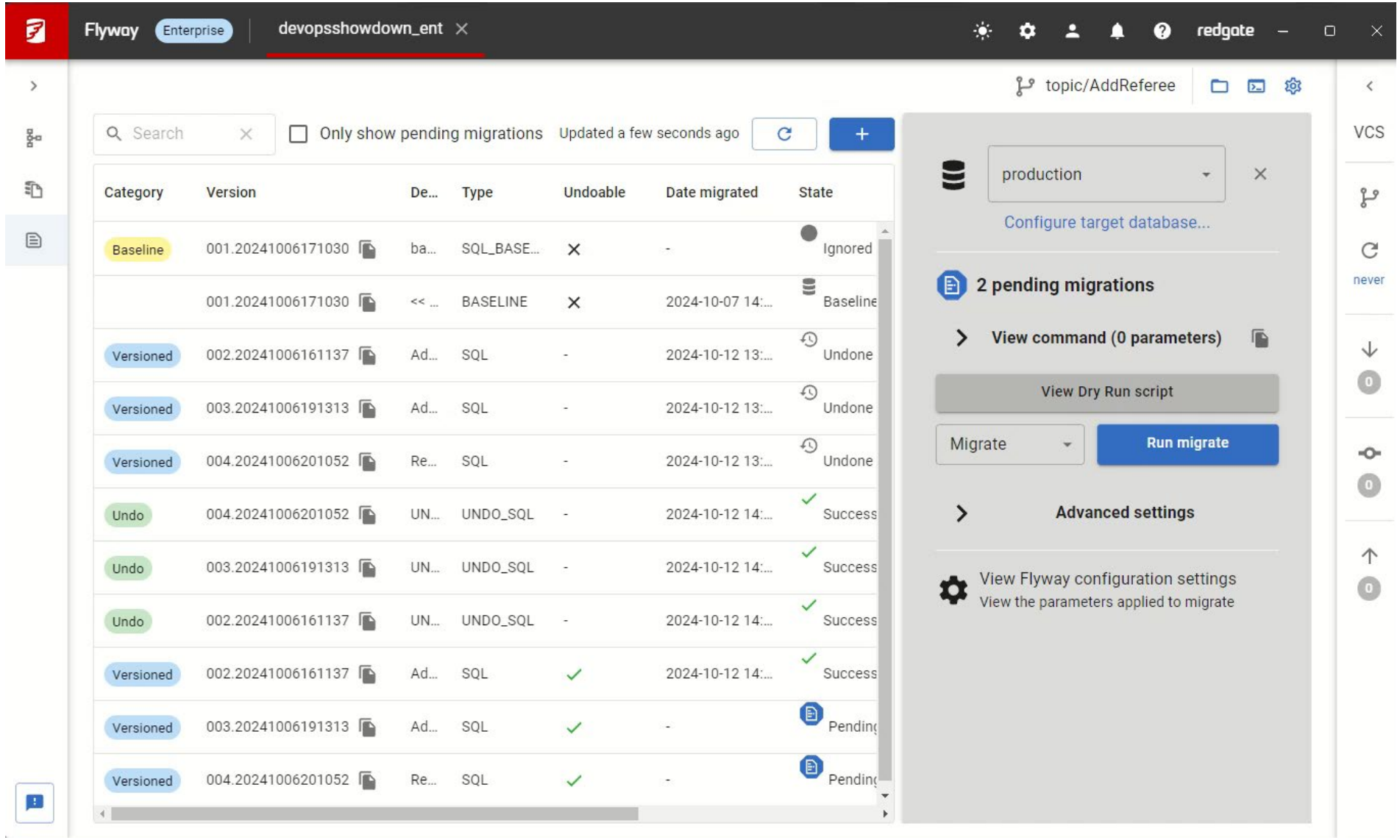
Undo multiple migration (cherrypick)



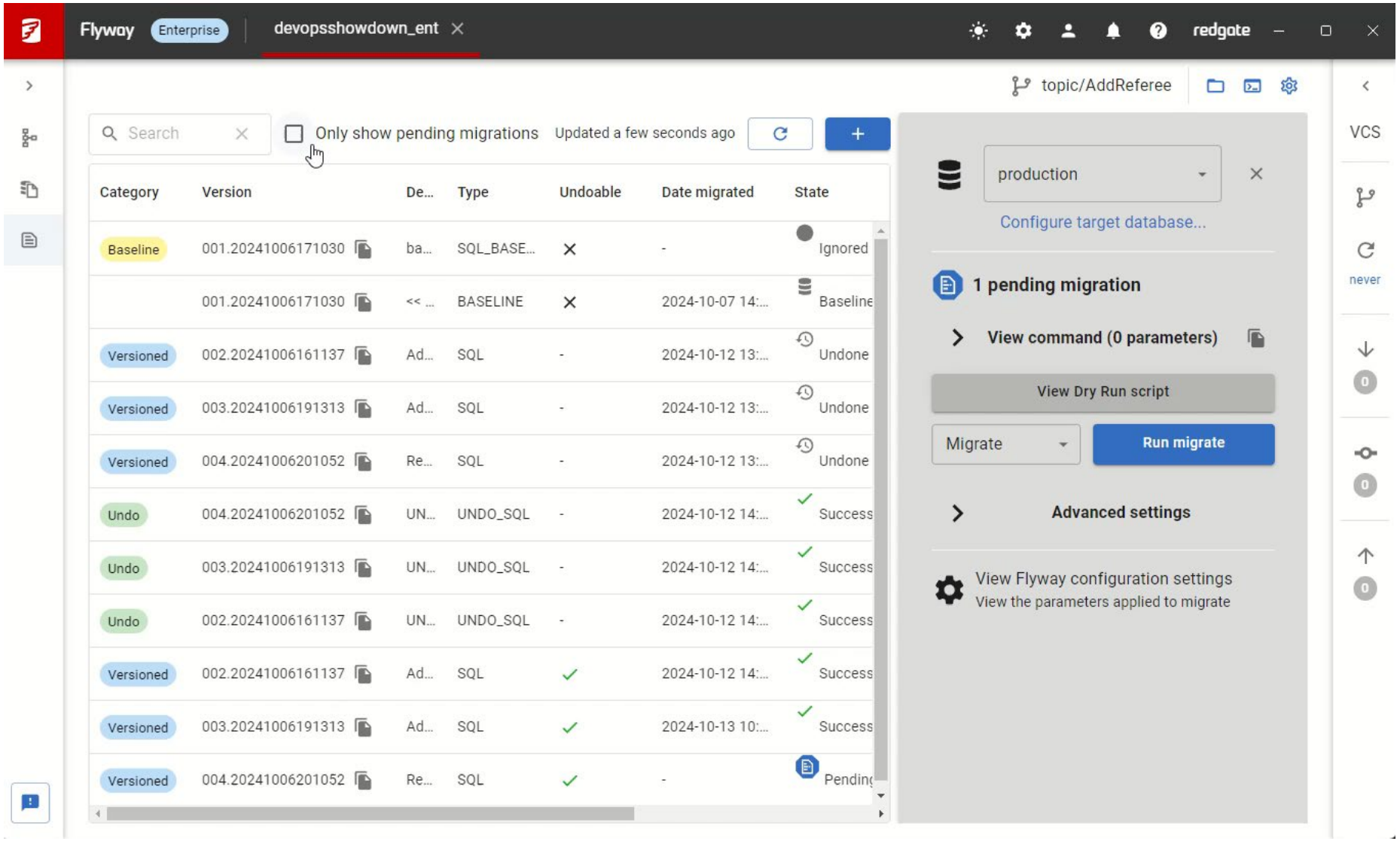
Migrate just one version (cherry pick)



Migrate to a version (target)



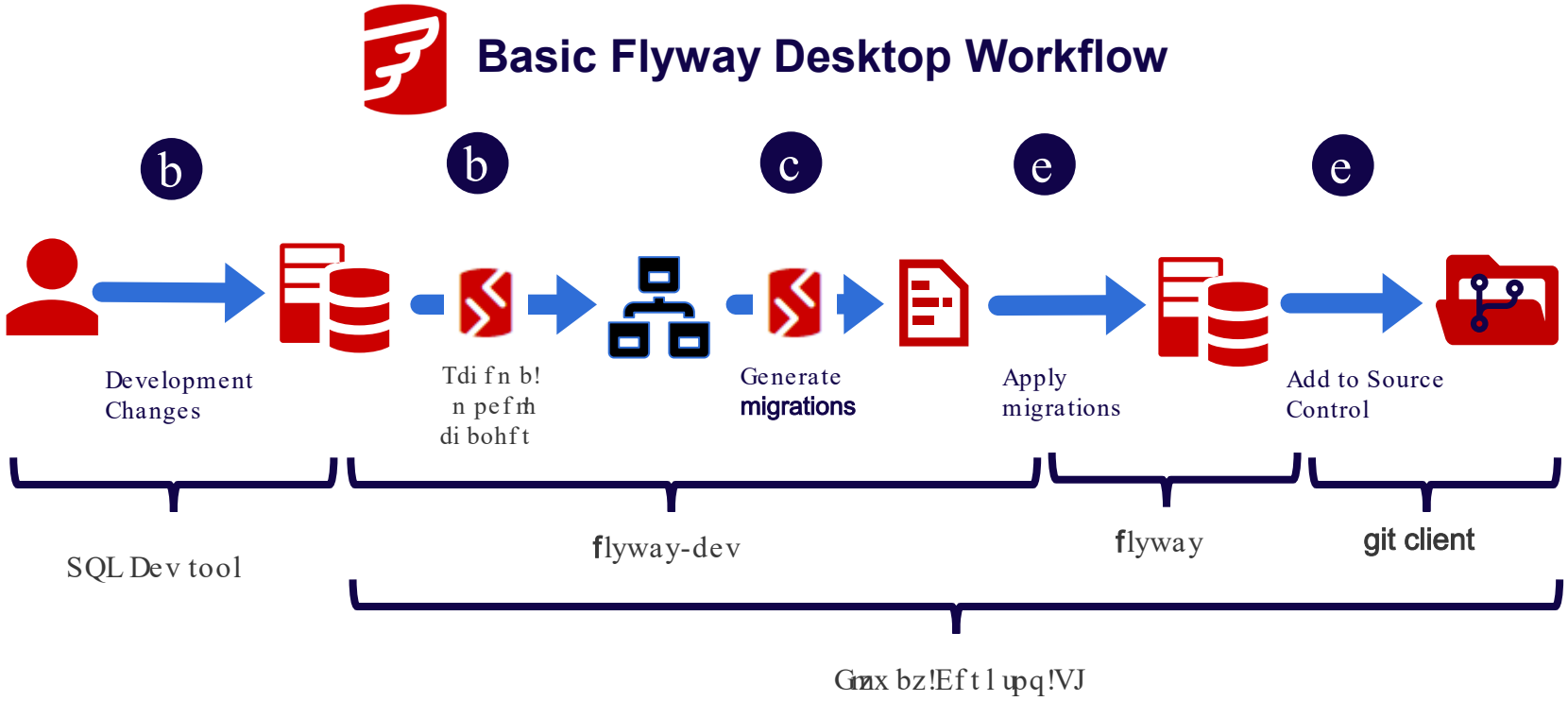
Undo to a version (target)



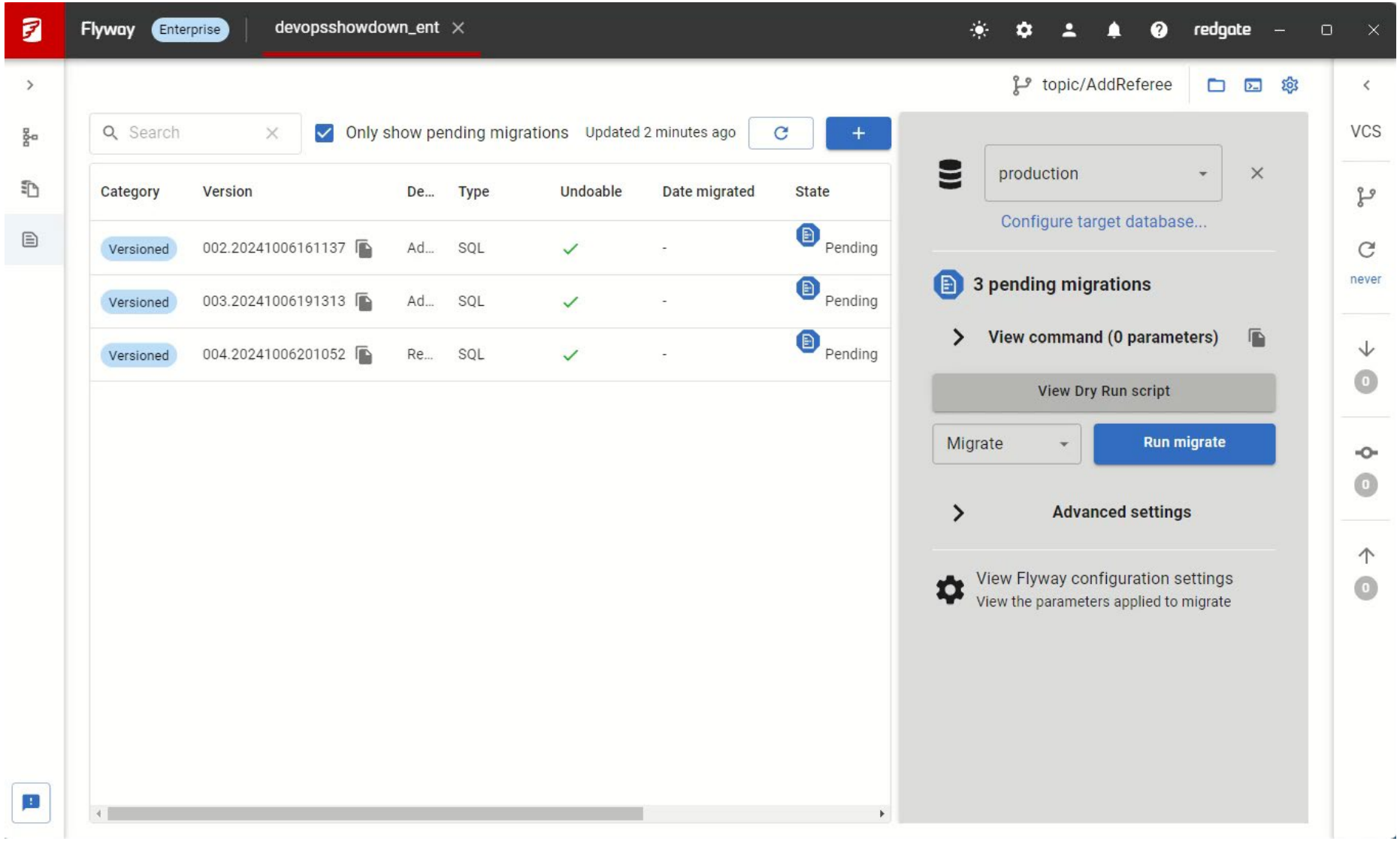
Flyway target vs cherrypick

- Execution Flow
  - target: applies all migrations up to a certain point
    - in order
  - cherrypick: allows you to choose specific version(s) to run
    - potentially out of order
- Use Cases:
  - target: is often used for controlled, stepwise update / rollback
  - cherrypick: applying /undoing specific migrations in isolated scenarios

What just happened?



Flyway CLI – Migrate



Flyway CLI

- flyway
  - migrate
  - -environment=production
  - -configFiles=
    - “C:\Users\Tonie\source\devopsshowdown_ent\flyway.toml,
    - C:\Users\Tonie\source\devopsshowdown_ent\flyway.user.toml”
  - -workingDirectory= “C:\Users\Tonie\source\devopsshowdown_ent”

Flyway CLI

- flyway
  - migrate
  - -environment=
  - -configFiles=
    - “C:\Users\To
    - C:\Users\Ton
  - -workingDirectory= “C:\Users\Tonie\s

- Auth
- Baseline
- Check Command
- Clean
- Info
- List Engines
- Migrate
- Repair
- Snapshot
- Undo
- Validate

Parameters

Page last updated 03 O 2024

Connection

- url
- user
- password
- driver
- connectRetries
- connectRetriesInterval
- initSql
- jdbcProperties

General

- batch
- callbacks
- cherryPick Flyway Teams
- configFileEncoding

Advanced Flyway Deployment Configuration

- Callbacks
  - Not only for migrate
- Useful for:
  - Data (un)loading
  - Pre / post processes
  - Custom RDBMS calls
- Grate alternative:
  - “Everytime” scripts

Migrate	Execution
beforeMigrate	Before Migrate runs
beforeRepeatables	Before all repeatable migrations during Migrate
beforeEachMigrate	Before every single migration during Migrate
beforeEachMigrateStatement	Before every single statement of a migration during Migrate
afterEachMigrateStatement	After every single successful statement of a migration during Migrate
afterEachMigrateStatementError	After every single failed statement of a migration during Migrate
afterEachMigrate	After every single successful migration during Migrate
afterEachMigrateError	After every single failed migration during Migrate
afterMigrate	After successful Migrate runs
afterMigrateApplied	After successful Migrate runs where at least one migration has been applied
afterVersioned	After all versioned migrations during Migrate
afterMigrateError	After failed Migrate runs

Lab

Deploy your changes

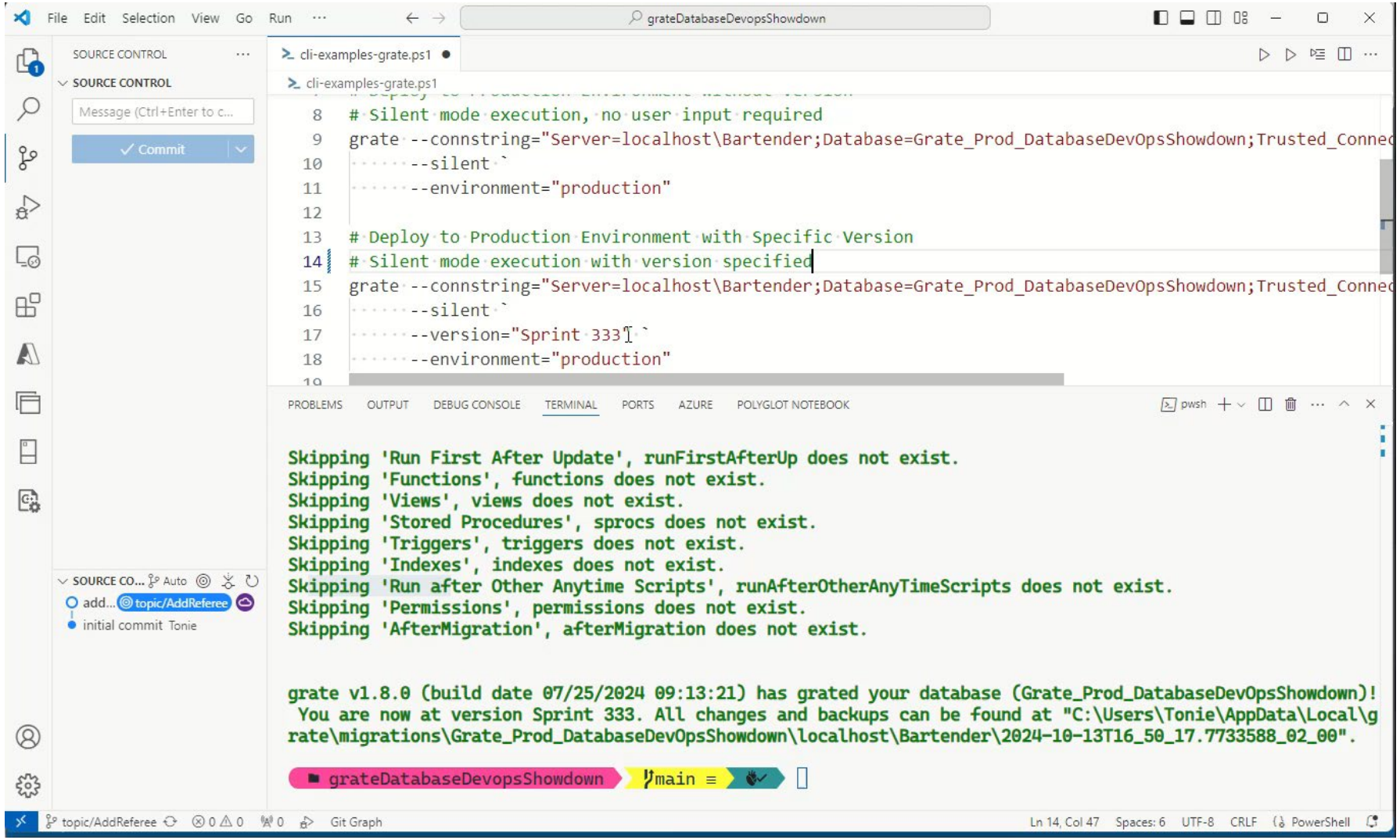
Person wrapping bandage photo – Free United kingdom Image on Unsplash

Deploy

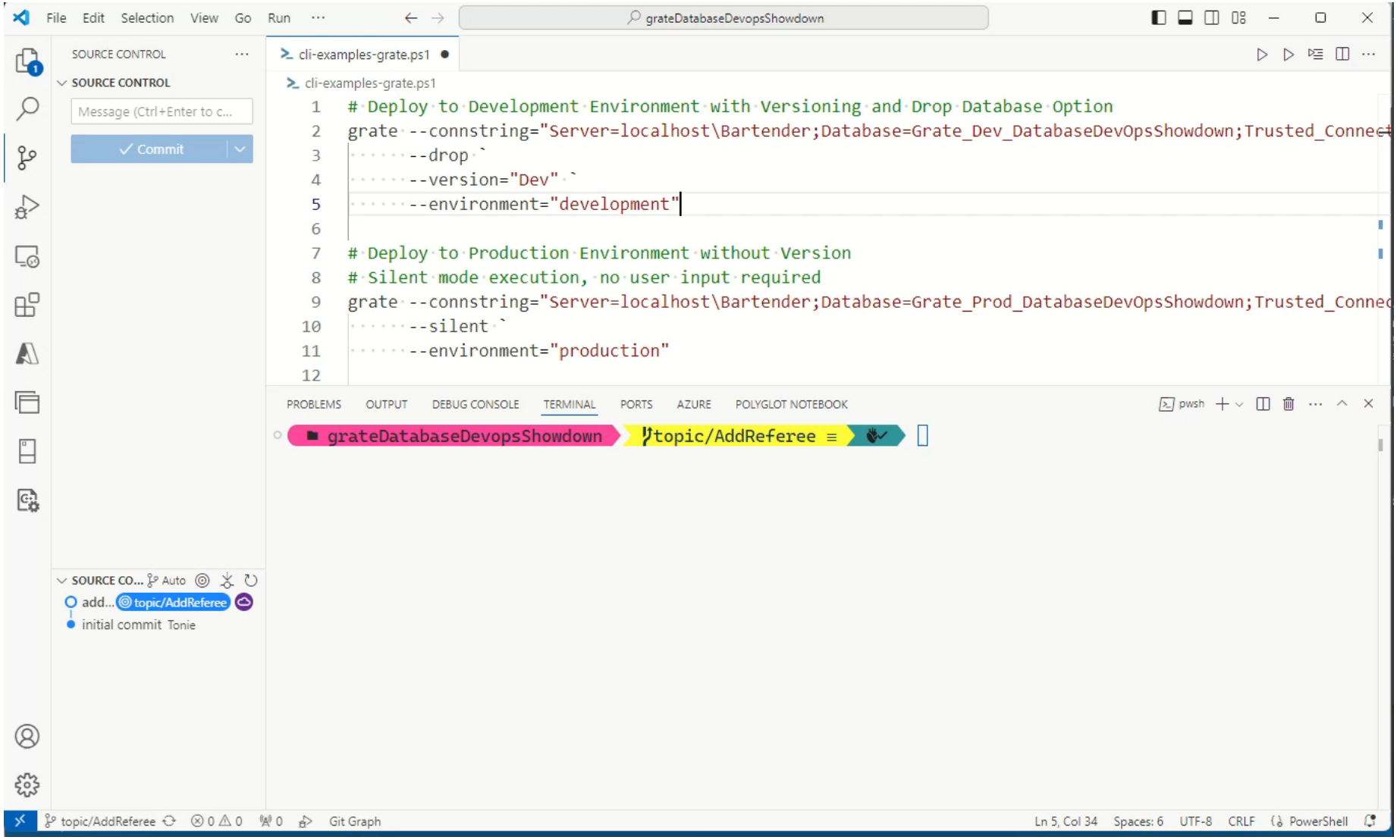
With grate

A man is doing a handstand on another man photo – Kung fu Image on Unsplash

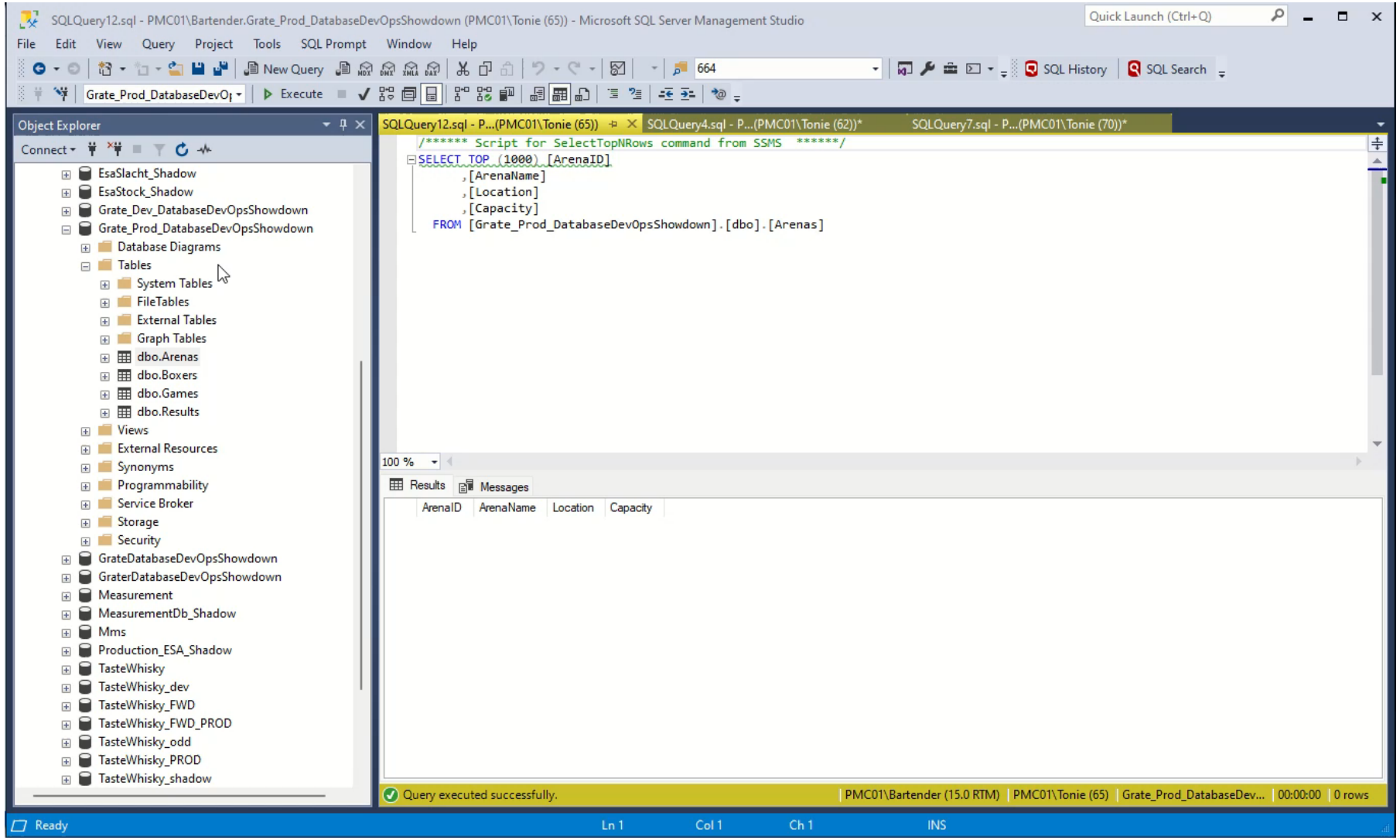
Deploy to production with version



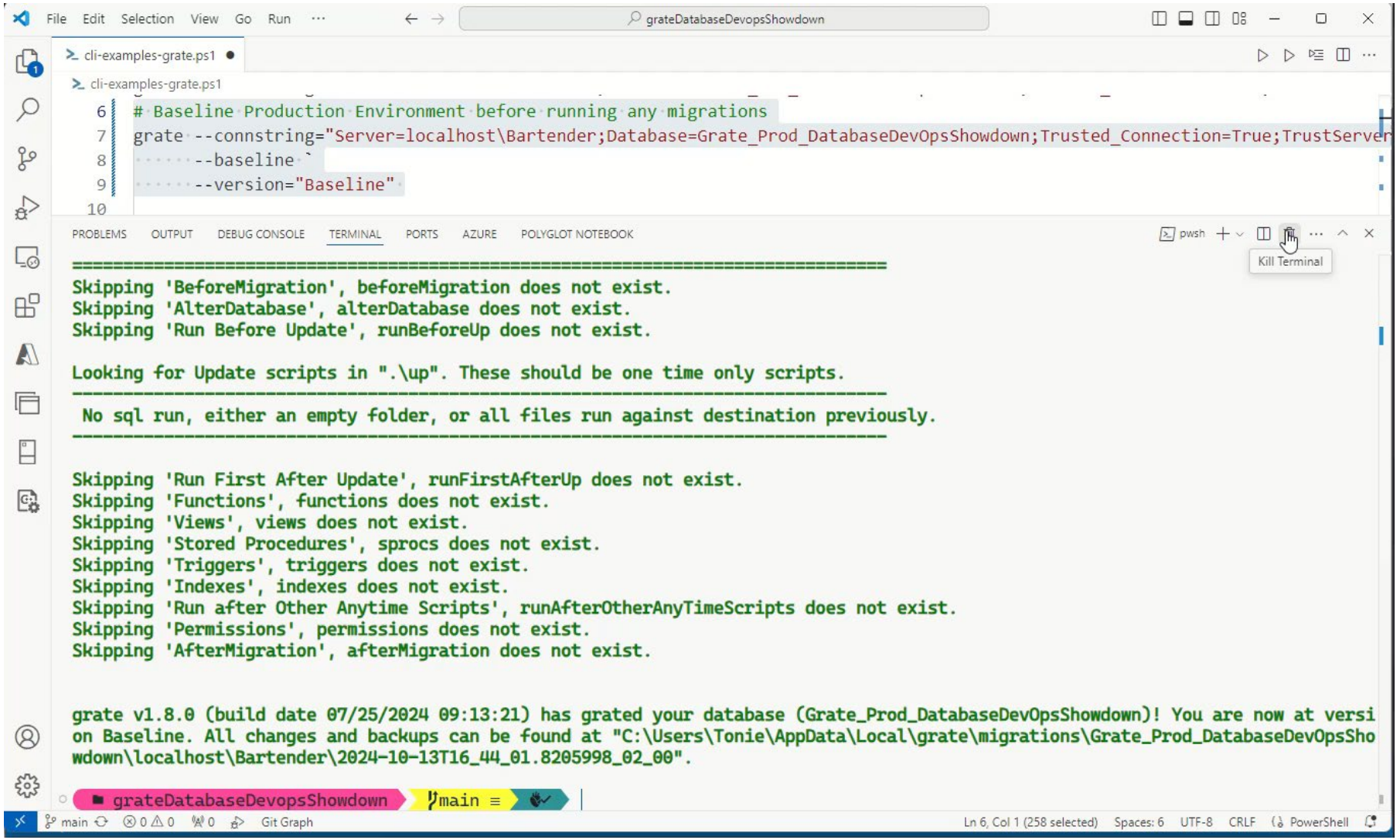
Deploy to dev, but drop the db first



First "Activate" grate on production

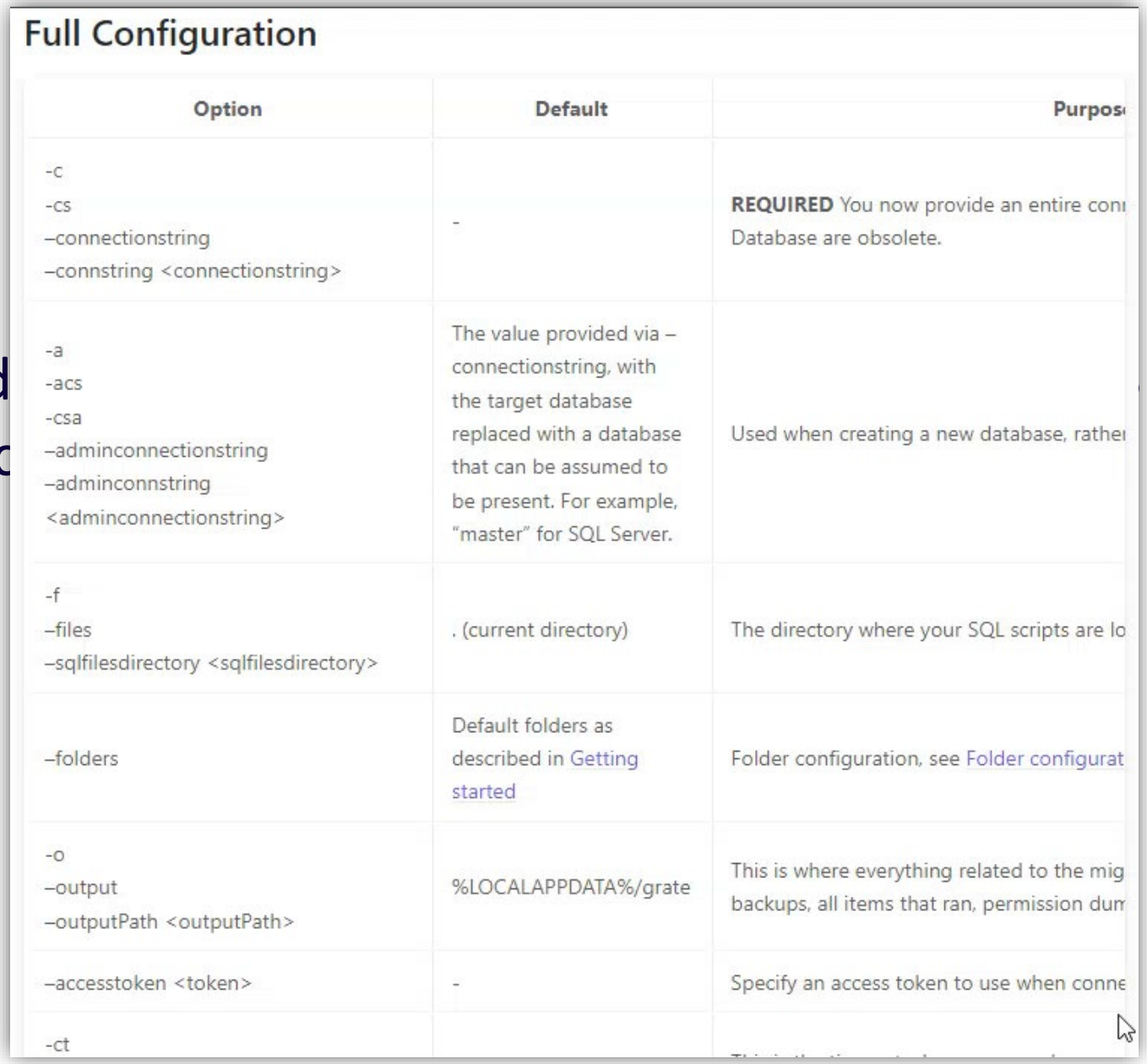


First "Activate" grate on production



Grate CLI

- grate
  - --connstring="Server=localhost\Bartend
    howdown;Trusted_Connec
  - --version=2.0
  - --silent
  - --drop



Full Configuration

Option	Default	Purpos
-c -cs –connectionstring –connstring <connectionstring>	-	**REQUIRED** You now provide an entire con Database are obsolete.
-a -acs -csa –adminconnectionstring –adminconnstring <adminconnectionstring>	The value provided via –connectionstring, with the target database replaced with a database that can be assumed to be present. For example, "master" for SQL Server.	Used when creating a new database, rathe
-f –files –sqlfilesdirectory <sqlfilesdirectory>	. (current directory)	The directory where your SQL scripts are lo
–folders	Default folders as described in Getting started	Folder configuration, see Folder configurat
-o –output –outputPath <outputPath>	%LOCALAPPDATA%/grate	This is where everything related to the mig backups, all items that ran, permission dum
–accesstoken <token>	-	Specify an access token to use when conne
-ct		

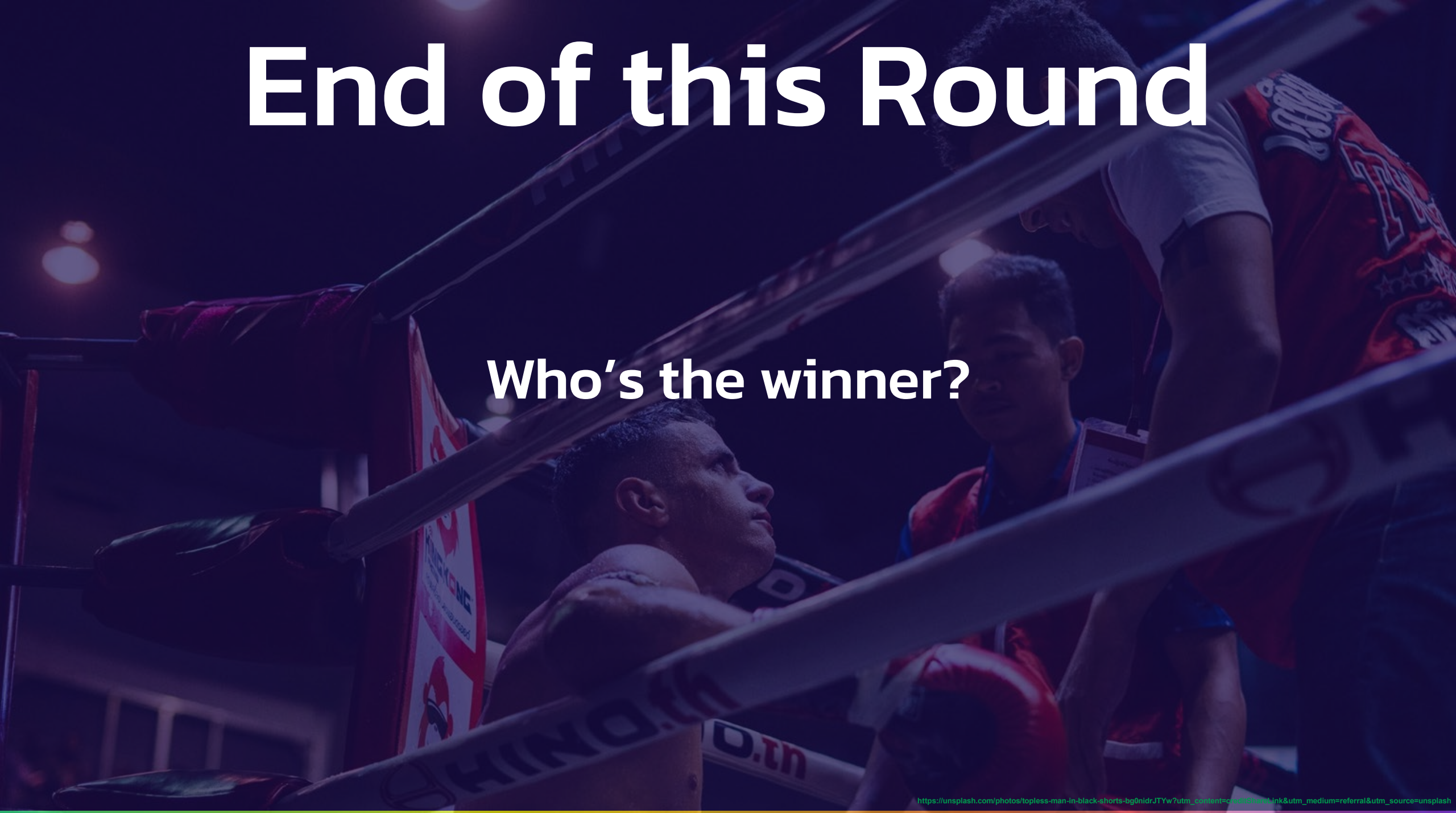
Lab

Deploy your changes

Person wrapping bandage photo – Free United kingdom Image on Unsplash

End of this Round

Who’s the winner?



https://unsplash.com/photos/topless-man-in-black-shorts-bg0nidrJTYw?utm_content=creditCopyText&utm_medium=referral&utm_source=unsplash

PASS24
Data Community Summit