

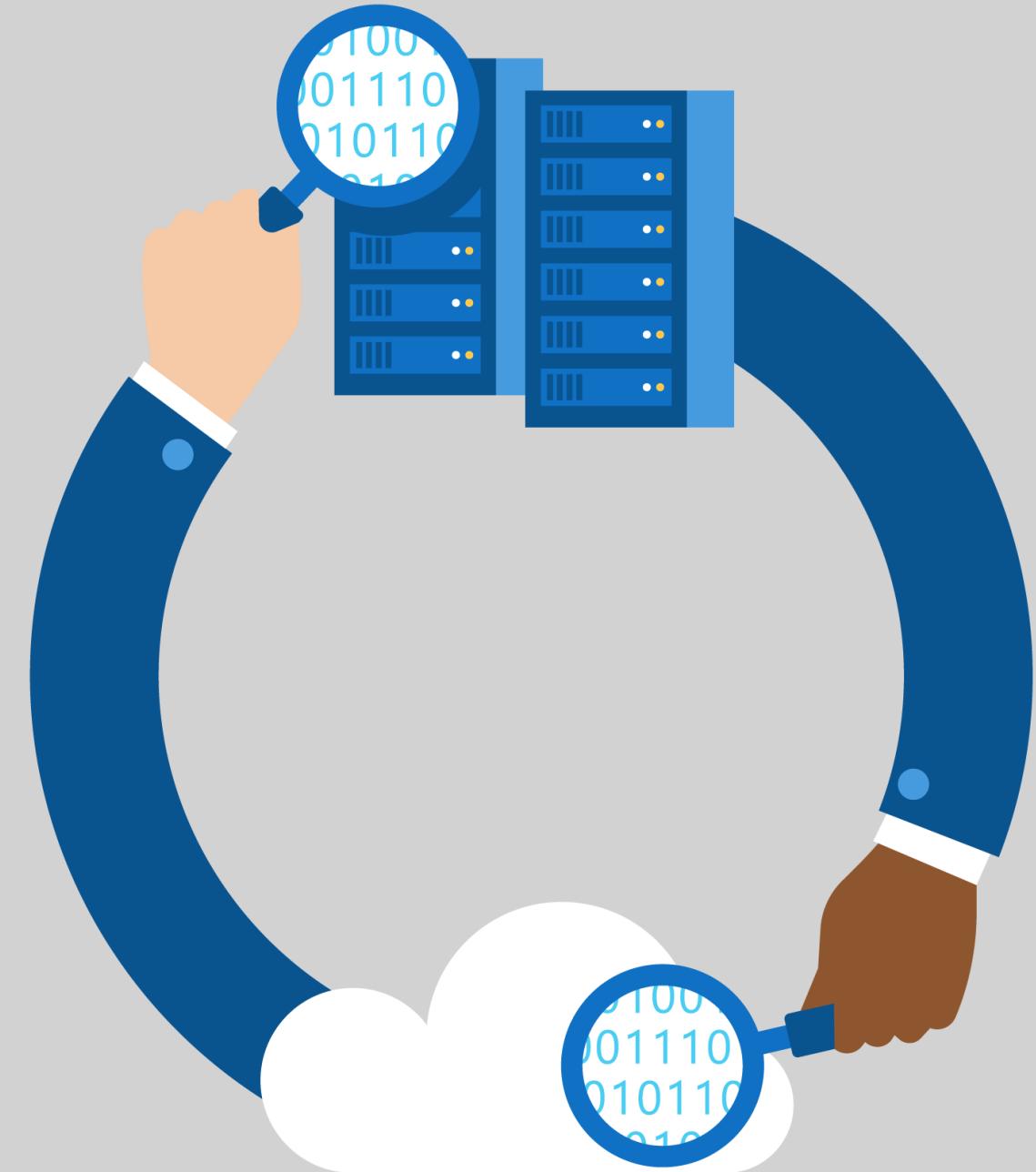


# Database DevOps



## SQL Server Testing

Franck Théolade  
[frtheola@microsoft.com](mailto:frtheola@microsoft.com)



# Introductions

## Franck Théolade

Application Development Manager at  
Microsoft UK  
Data Engineering  
Azure DevOps



# Introductions



What is your name?



What is your job role?



What would you like  
to learn about?

# Agenda

SQL Testing  
Core Concepts  
and tSQLt



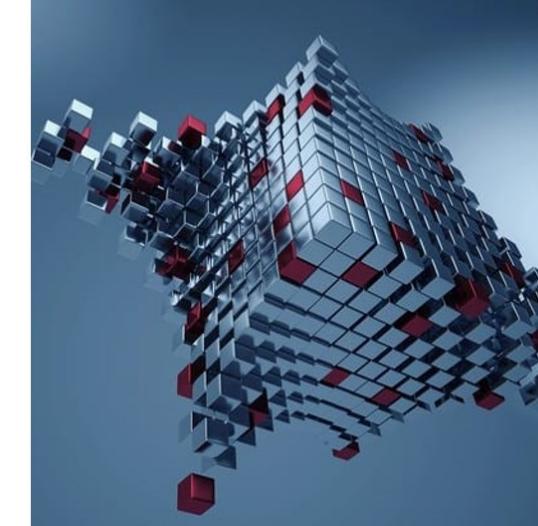
Test Driven Design  
for SQL Server



Hands-on Lab #1  
tSQLt Unit testing



SQL Server  
Management Studio



BREAK



Azure  
Pipeline  
Integration



Hands-on Labs #2  
Deploying & Testing  
Database Changes  
with Azure DevOps



LUNCH



Your Feedback Matters...



[aka.ms/RedBullDBTesting](http://aka.ms/RedBullDBTesting)

# SQL Testing Core Concepts and tSQLt



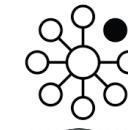
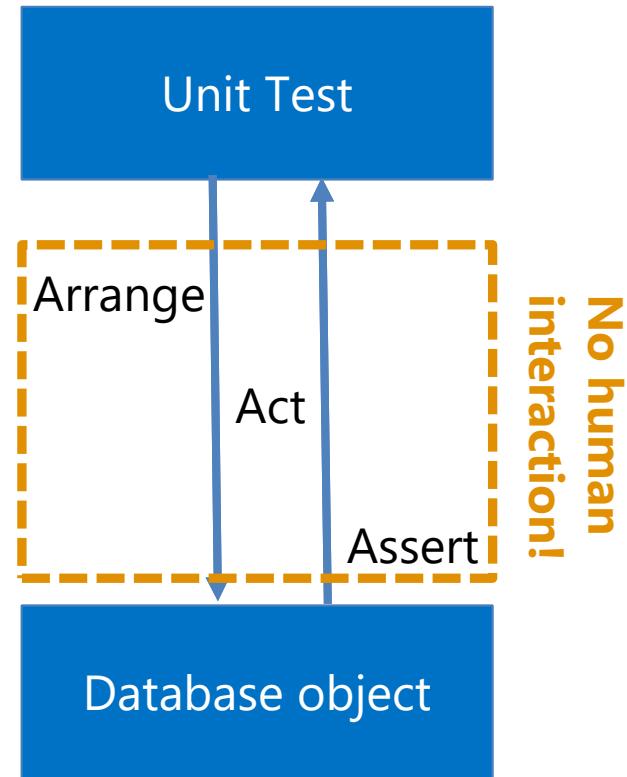
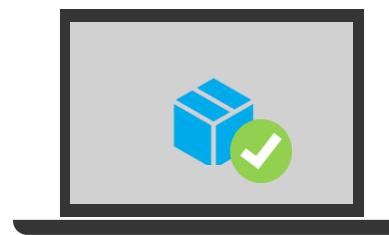
# What is SQL Unit Testing?

“

*...Is the process of looking for discreet chunks of code that make for convenient “**units of work**”, to which to **apply automated low-level tests** both for existing **legacy** and **greenfield applications**.*”



Database Developers



Isolation



Low-level



Database Units



Prevents regression

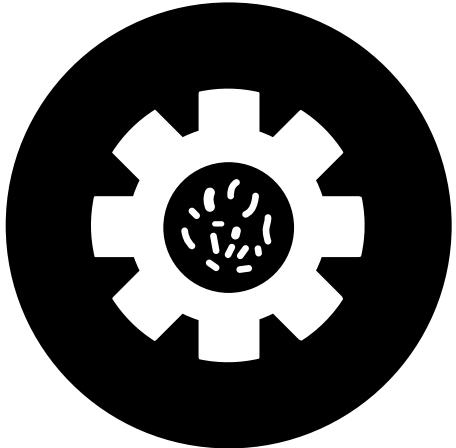


Safety net

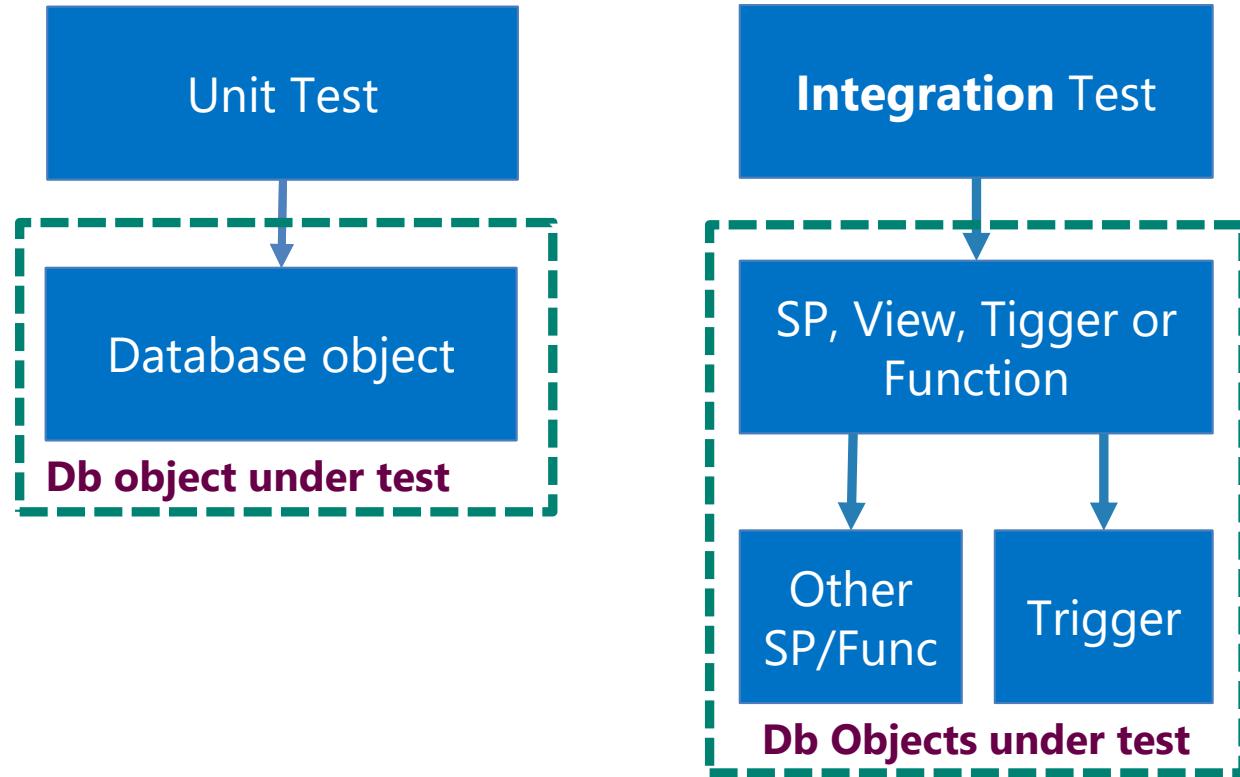


Executable Specification

# Flavours of Unit Tests



# What's different between unit & integration tests?



Db Objects integration



Bigger Picture



External interactions

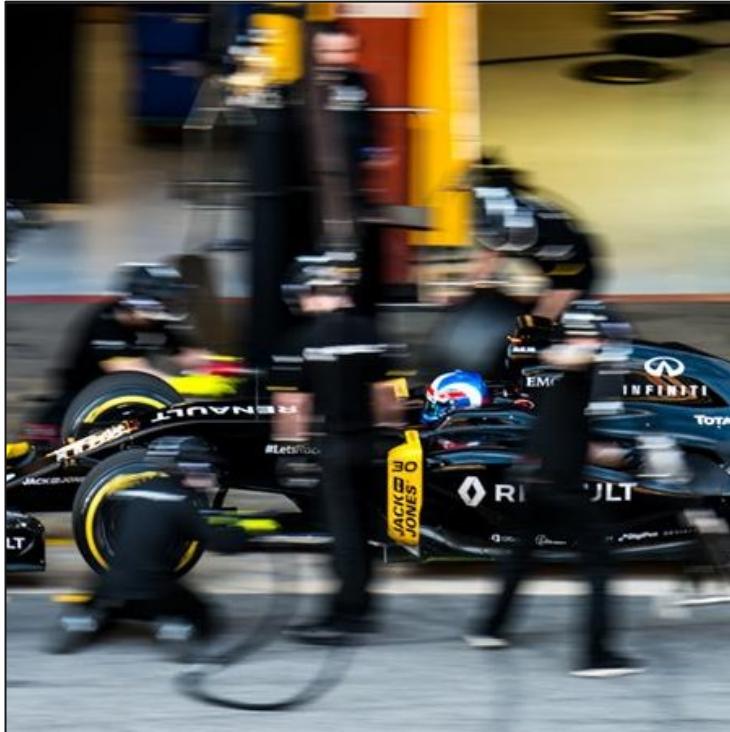


Slower to run

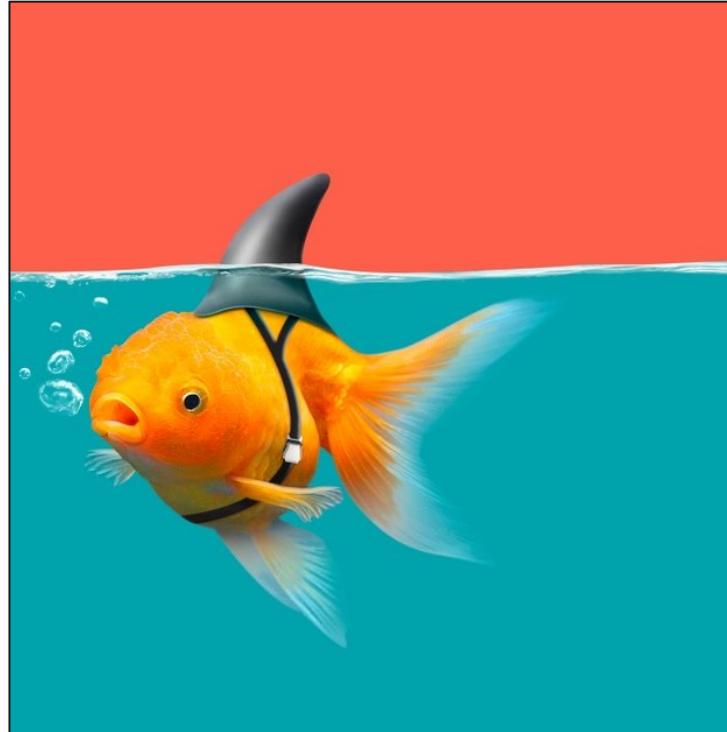


Configuration &  
Maintenance

# SQL Unit Testing Best Practices



F.I.R.S.T. & Best Practices

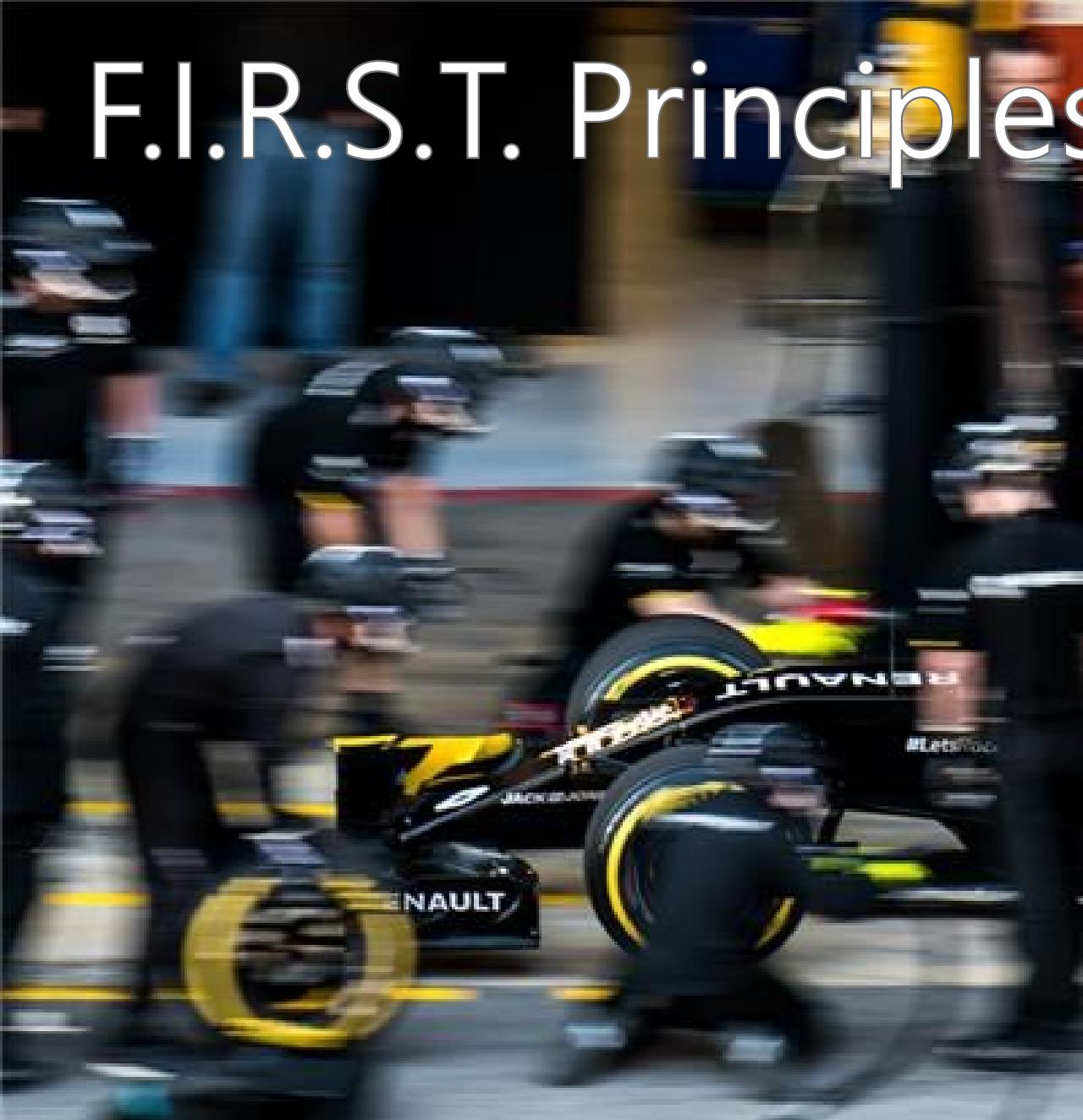


Test Doubles &  
Test Data generation



Pipeline CI/CD & IDE  
Integration

# F.I.R.S.T. Principles



Fast / False  
Outcome

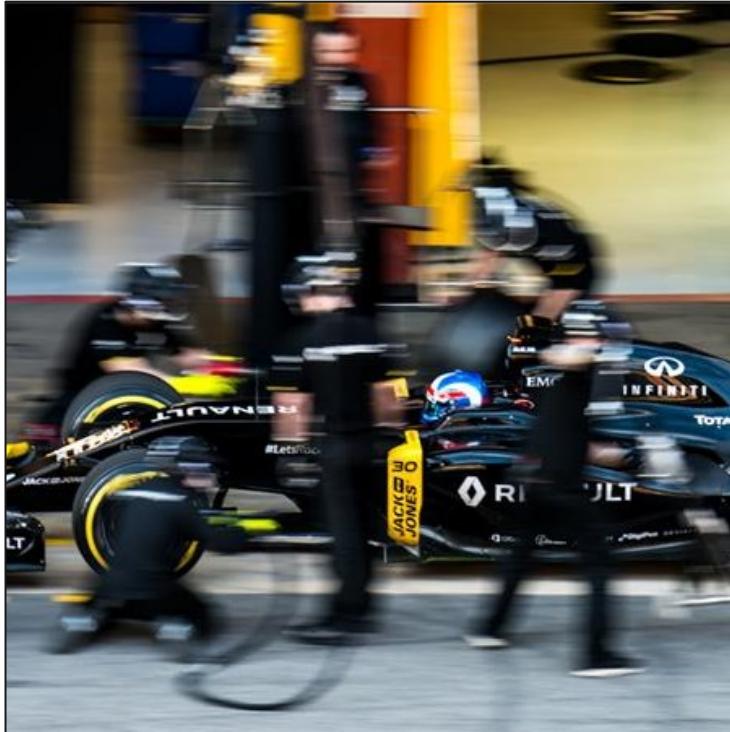
Independent

Repeatable /  
Readable

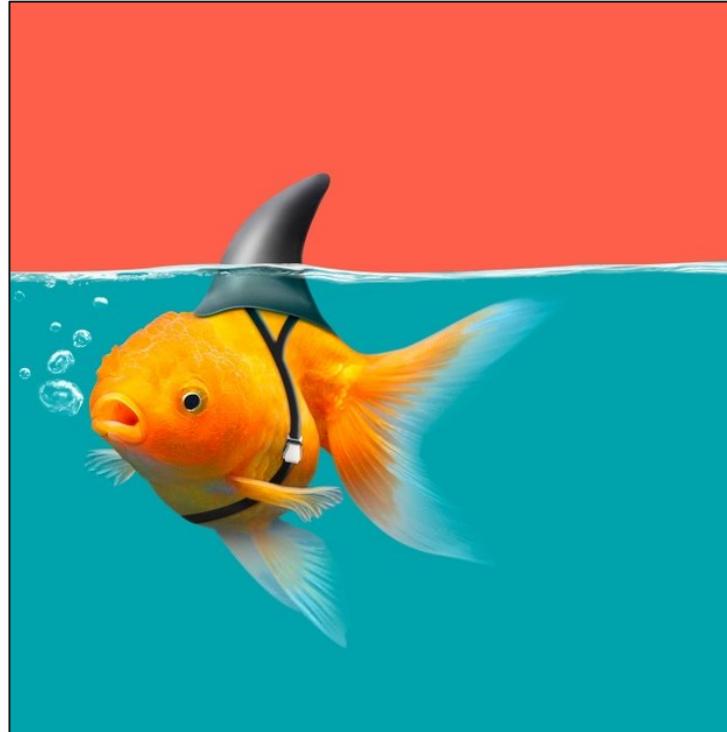
Self-validating

Timely / TDD

# SQL Unit Testing Best Practices



F.I.R.S.T. Principles



Test Doubles &  
Data generation



Pipeline CI/CD & IDE  
Integration

# Mocks aren't stubs!



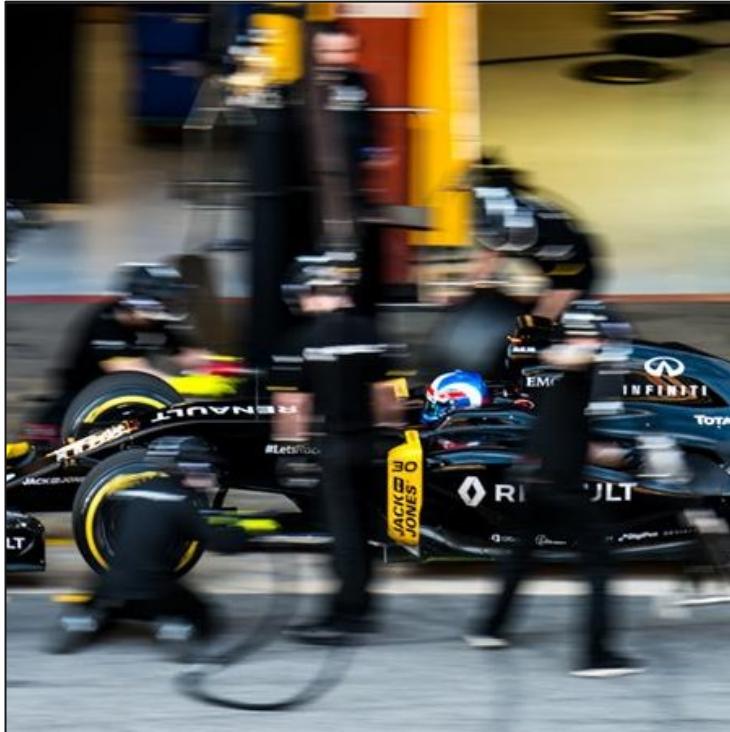
Dummies

Stubs

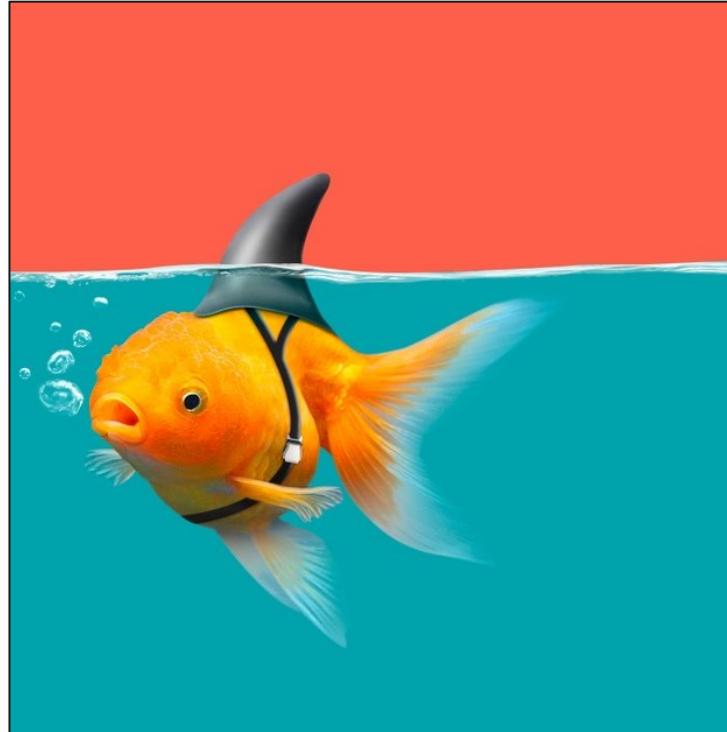
Spies

Mocks

# SQL Unit Testing Best Practices



F.I.R.S.T. Principles



Test Doubles &  
Data generation



Pipeline CI/CD & IDE  
Integration



Source Control

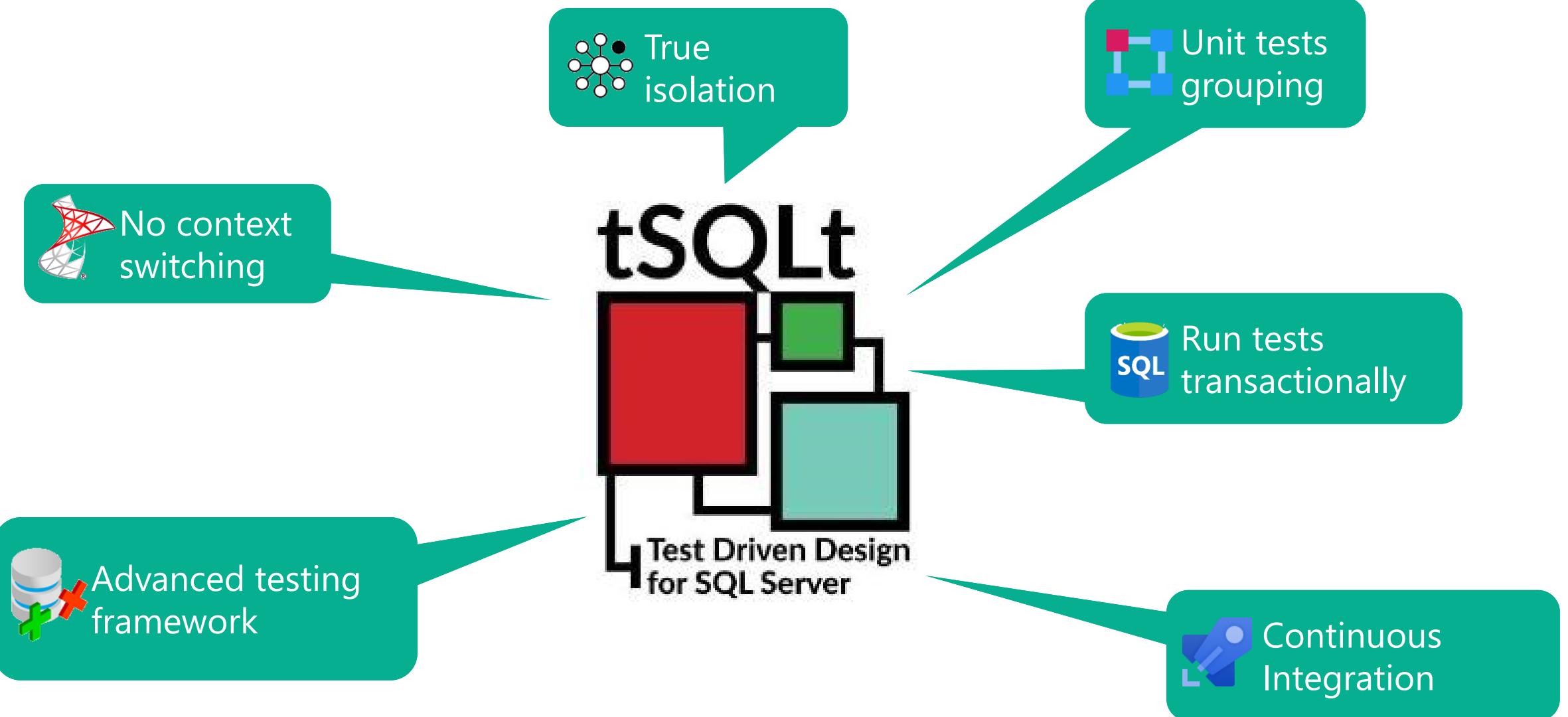
Setup Test Data

Run Tests

Tests result  
Reporting

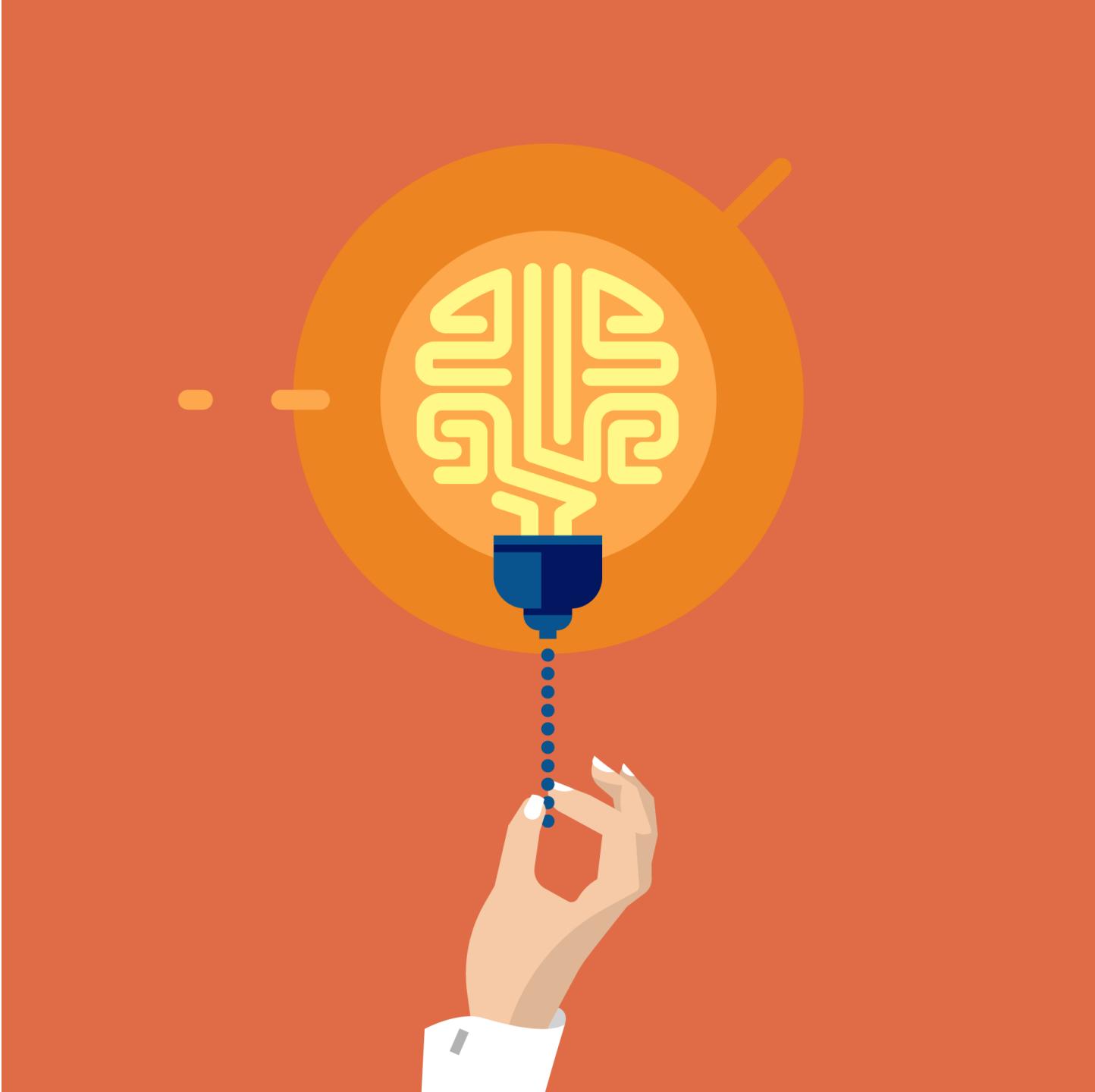
Code Coverage  
Reporting

# Introducing tSQLt

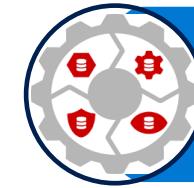
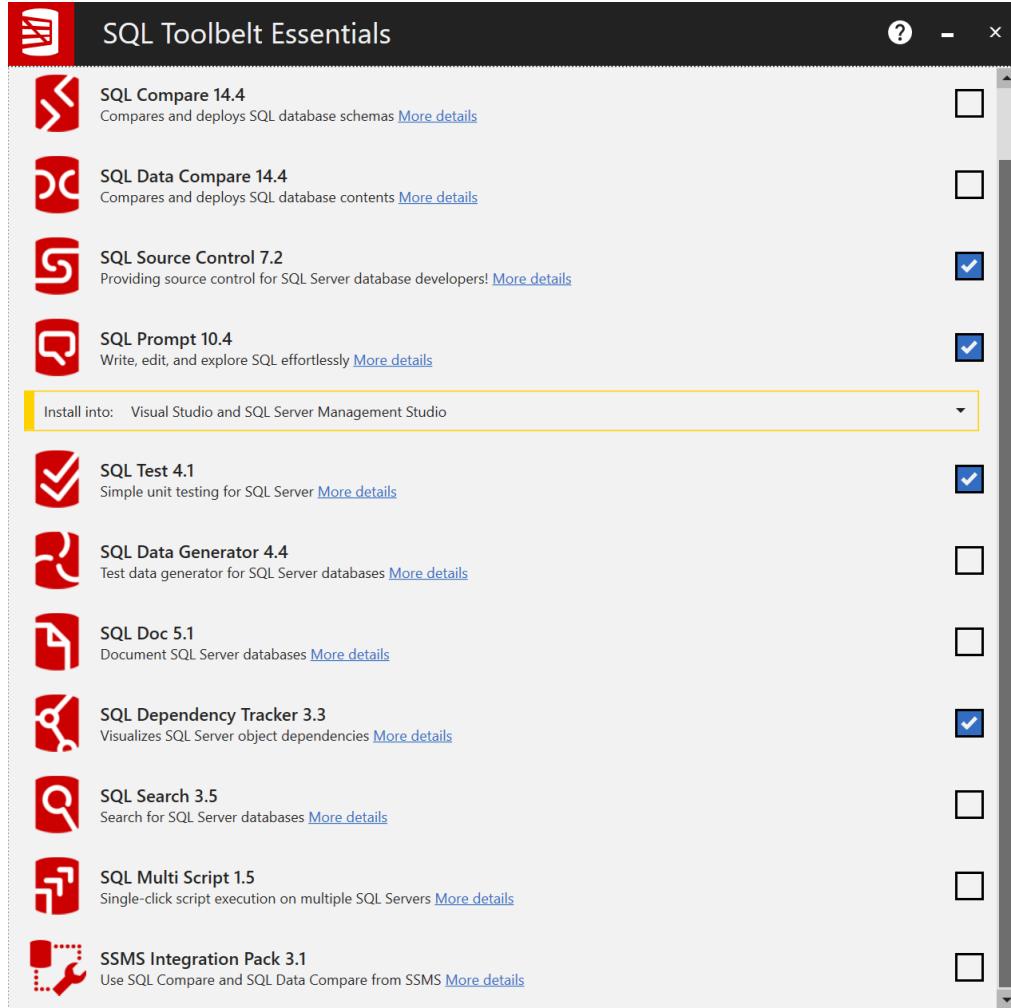


# tSQLt Demo

<https://tsqlt.org/user-guide/quick-start/>



# Enhancing SSMS Developer experience



Nice to have 3<sup>rd</sup> party licenced tool



Better IDE Developer Experience & Adoption



Better Test management

<https://www.red-gate.com/products/sql-development/sql-test>

<https://www.devart.com/dbforge/sql/unit-test/>



# Enforcing development standards

Identify  
consistent  
conventions

Write a tSQLt  
unit test to  
enforce it

```
EXEC tSQLt.NewTestClass @ClassName = N'DevelopmentStandards'
GO
CREATE PROC DevelopmentStandards.[test Naming Convention - test
classes correspond to object in database] AS
--Assemble
--Act
SELECT Name,
SchemaID,
OBJECT_ID('dbo.' + name) AS ObjectID
INTO DevelopmentStandards.Actual
FROM tSQLt.TestClasses
WHERE OBJECT_ID('dbo.' + name) IS NULL
AND NAME != OBJECT_SCHEMA_NAME(@@ProcID)
ORDER BY NAME

--Assert

EXEC tSQLt.AssertEmptyTable 'DevelopmentStandards.Actual'

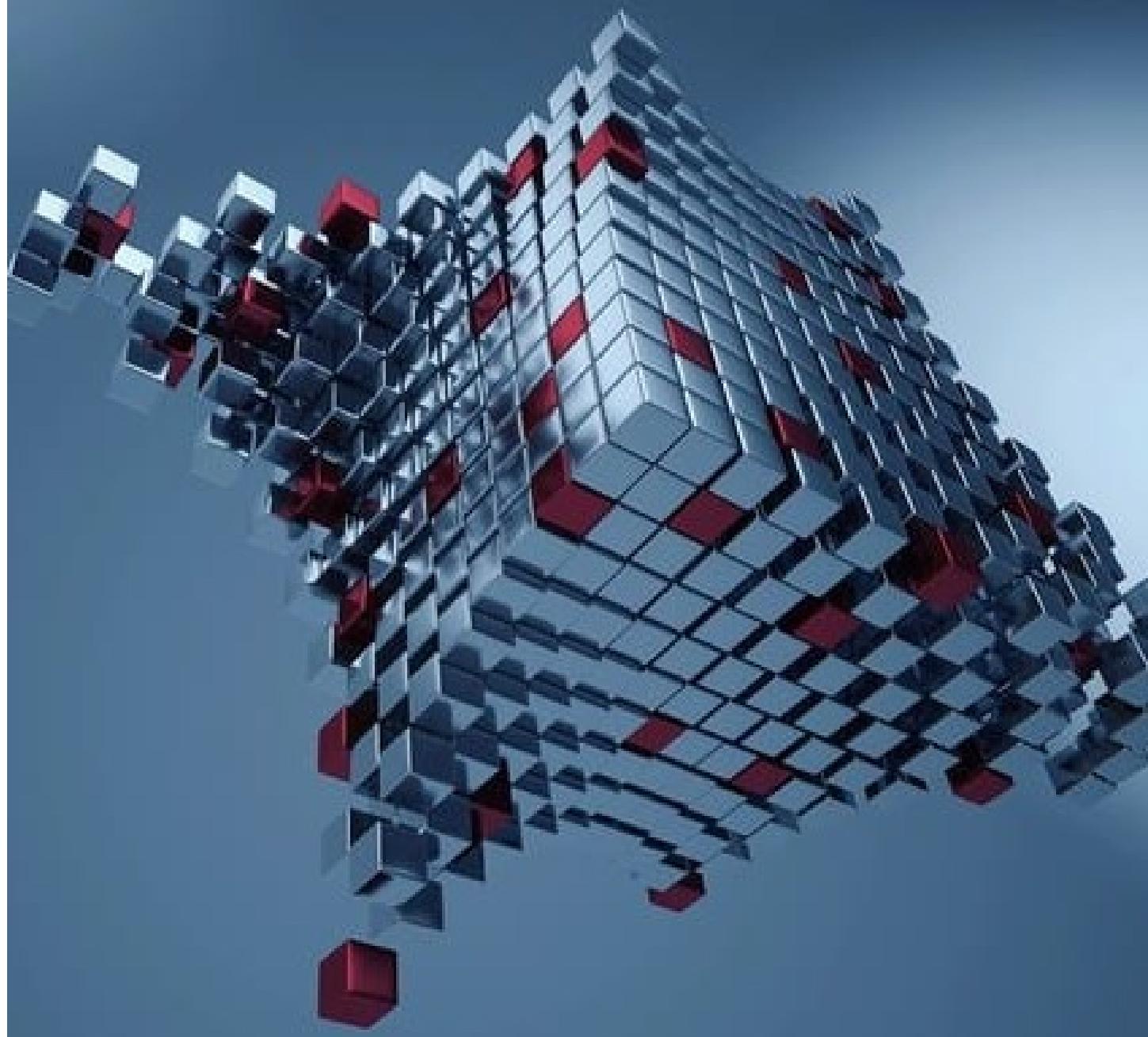
GO
```

# Hands-on Lab #1

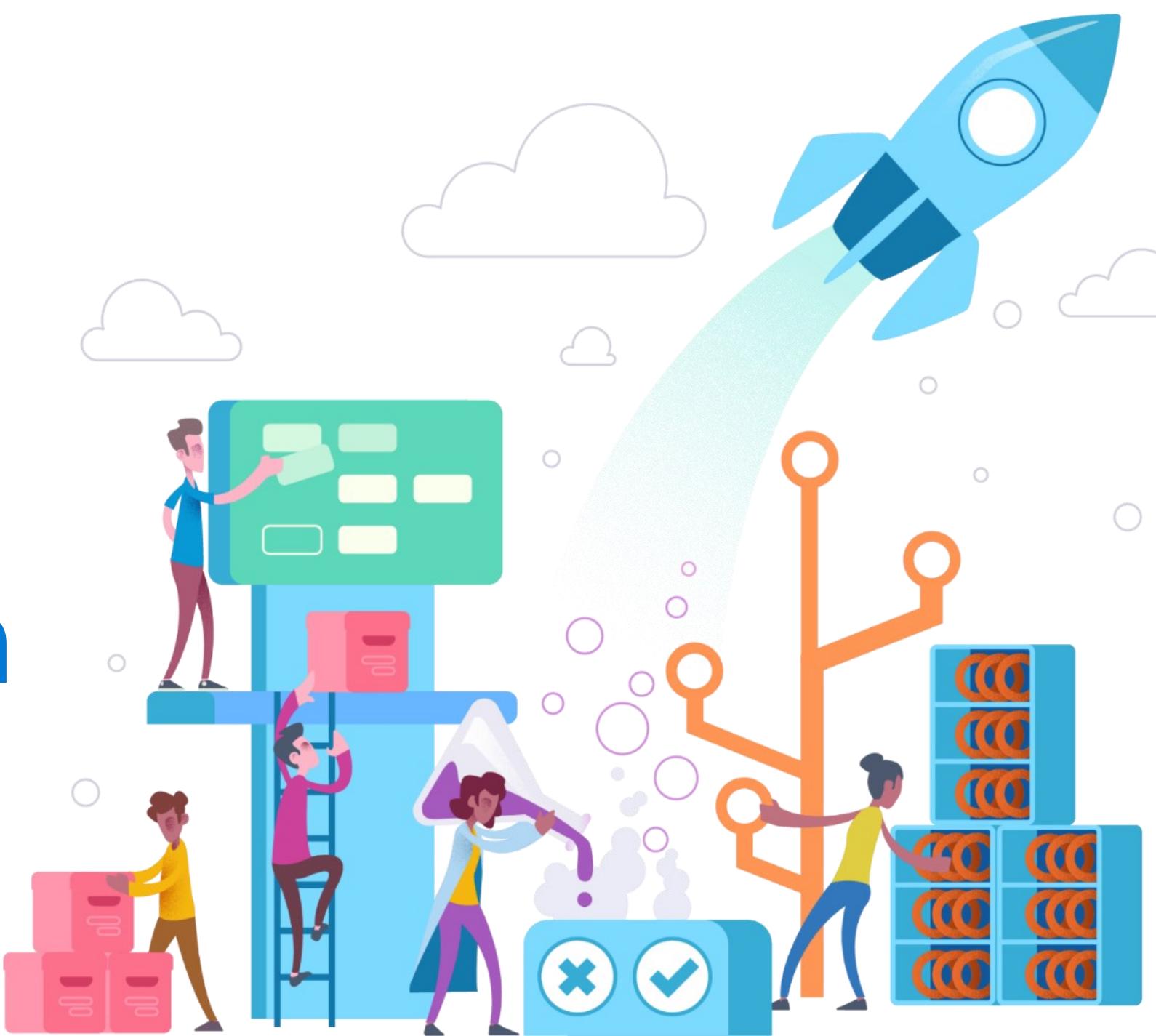
## tSQLt Unit testing



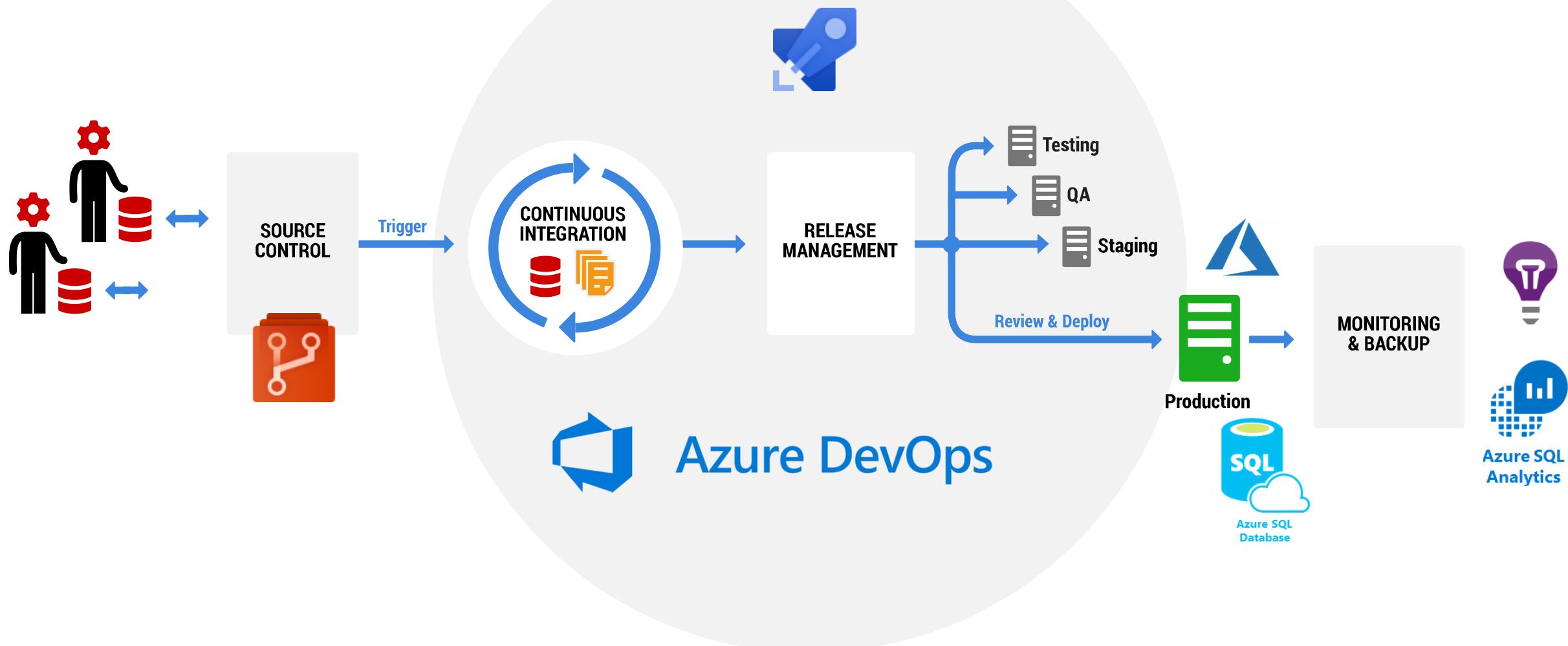
SQL Server  
Management Studio



# Azure Pipeline Integration



# Bringing the Database to Azure DevOps





# Hands-on Labs #2

## Deploying & Testing

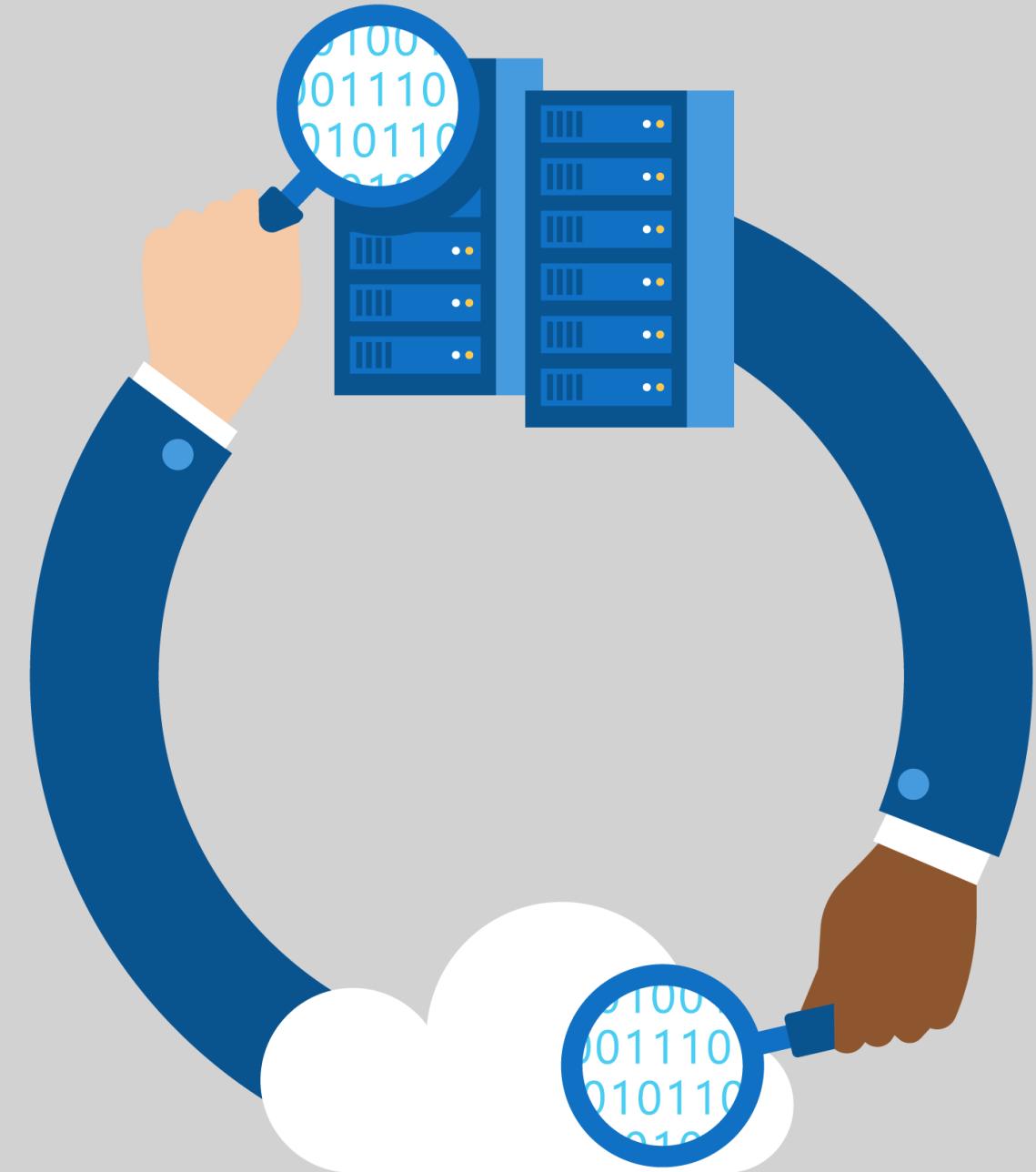
## Database Changes

## with Azure DevOps

# Your Feedback Matters...



[aka.ms/RedBullDBTesting](https://aka.ms/RedBullDBTesting)



Q&A

You have  
Questions      We have  
Answers

# Thank You

ευχαριστώ Salamat Po متشرّم شكرًا Grazie

благодаря ありがとうございます Kiitos Teşekkürler 谢谢

ឧបម្ពុណទំរង់ Obrigado شكريه Terima Kasih Dziękuję

Hvala Köszönöm Tak Dank u wel дякую Tack

Mulțumesc спасибо Danke Cám ơn Gracias

多謝晒 Ďakujem הודה දෙකුඩී Děkuji 감사합니다