



Growing Our Community

Offline Database Development and Unit Testing with SSDT

Andrey Zavadskiy



Thank You



Gain insights through familiar tools while balancing monitoring and managing user created content across structured and unstructured sources.

www.microsoft.com

Presenting Sponsors



Unifying computer, storage, networking, and virtualization, Cisco UCS is the optimal database and business intelligence platform for SQL Server.

www.cisco.com



Solutions from Dell help you monitor, manage, protect and improve your SQL Server environment.

www.software.dell.com

Supporting Sponsors





PASS
SUMMIT 2015

October 27-30 / Seattle, WA

Planning on attending PASS Summit 2015? Start saving today!

- The world's largest gathering of SQL Server & BI professionals
- Take your SQL Server skills to the next level by learning from the world's SQL Server experts, in over 190 technical sessions
- Over 5000 attendees, representing 2000 companies, from 52 countries, ready to network & learn

Contact your Local or Virtual Chapter for an additional \$150 discount.

\$1795

until July 12th, 2015

Andrey Zavadskiy

- Solutions architect, SQL & .NET developer
- 20 years in IT industry
- Worked with SQL Server since 7.0 back in 2001
- Developed in Visual Basic, C#, ASP.NET, MVC, JavaScript, SharePoint
- MCSE, MCSD, MCT



<http://andreyzavadskiy.com>



<https://www.facebook.com/andrey.k.zavadskiy>



@AndreyZavadskiy



<https://www.linkedin.com/in/zavadskiy>

Session Goal

Provide a practical overview of how to use SQL Server Data Tools to create, test and deploy a database project

Contents

Offline database development

- Creating a database project
- Creating database objects and writing T-SQL code
- Project properties
- Deployment
- Deployment scripts for inserting data

Database unit testing

- Objects to be tested
- Unit test flow
- Creating and running unit test
- Debugging database objects in unit test
- Custom unit test conditions

Concept of SQL Server Data Tools

Focusing on the final version of code rather than on many ALTER iterations

The unique IDE for all database developer's needs

- Based on Visual Studio
- Working with connected databases (like SQL Server Management Studio)
- Project approach to code writing
- Debugging and unit testing
- Version control (via integration with Team Foundation Server)

Offline Database Development

Creating a database project

Creating database objects

- Import from existing database
- Creating from scratch

Project properties

- Project and database settings
- Code analysis

DEMO

Creating a database project

Database Deployment

Deployment in the connected database

- Publishes the script (CREATE)
- Compares against the target database
- Creates a change script (ALTER)
- Runs the change script on the target database

Deployment in the disconnected database

- Creates a DACPAC package
- Distribute and publish DACPAC
 - Compares DACPAC against the target database
 - Creates a change script
 - Runs the change script on the target database

Publishing profiles and settings

DEMO

Publishing a database

Deployment Scripts For Inserting Data

Creating scripts

- Based on the existing data
- From scratch

Various types of scripts

- Pre-deployment
- Post-deployment

DEMO

Scripting data for deployment

Database Unit Testing

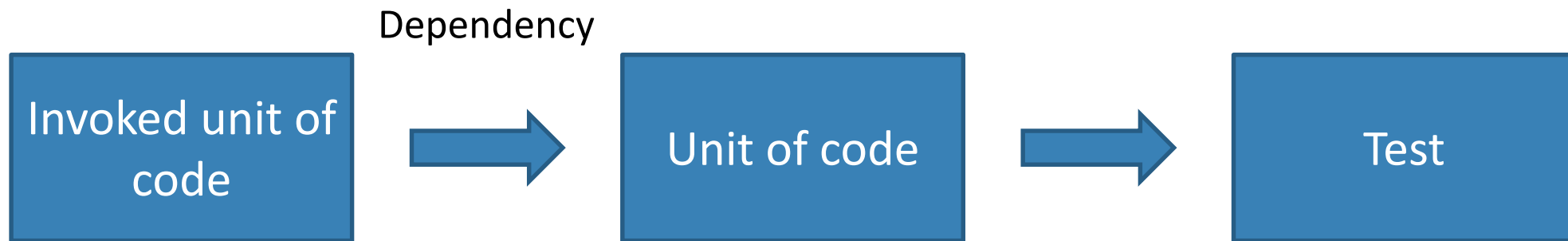
Unit test

- Is performed on a smallest piece of testable code
- Isolated from the other pieces of code
- Should be repeatable
- Gives the answer to only one question
- Usually created by developers

What for?

- Confidence in your code
- Confirms that product requirements are working
- Early error checking of code
- Instant visual feedback on errors
- Helps to check subsequent changes in code
- Provides documentation for other developers

Where are the bugs?



What can be tested?

Meta-data

- Table structure, field type and length
- Existence of objects

Constraints

- CHECK, DEFAULT, PRIMARY KEY, FOREIGN KEY, UNIQUE

T-SQL code

- Stored procedures, Functions, Triggers

Security permissions

Execution time

Data that can be tested

Scalar values

- Normal values
- Errors (incorrect values)
- Very big values
- NULL

Table values

- Rowset
- Empty rowset
- Very big rowset
- Metadata

Unit Test Flow

- Test initialize
 - Unit test
 - Pre-test
 - Test
 - Post-test
 - Test cleanup
-
- Can have more than one test condition
 - Can handle exceptions raised in database
 - Can be run within a transaction

DEMO

Creating and running a database unit tests

Debugging in unit tests

- Can debug only the T-SQL code to be tested
 - Breakpoint can be set inside the stored procedure, function or trigger
- Can't debug the T-SQL code of the unit test itself

DEMO

Debugging a code from within a database unit test

Custom unit test conditions

- Are a Visual Studio IDE extensions
- Created in a separate project as a class library
- Compiled to a DLL
- How-to in MSDN article “Custom Test Conditions for SQL Server Unit Tests”
 - [https://msdn.microsoft.com/en-us/library/jj860449\(v=vs.103\).aspx](https://msdn.microsoft.com/en-us/library/jj860449(v=vs.103).aspx)
- Example in my CodePlex project
 - <https://ssdtconditions.codeplex.com/>

DEMO

Using a custom test conditions

References

- **MSDN: SQL Server Data Tools**

[https://msdn.microsoft.com/en-us/library/hh272686\(v=vs.103\).aspx](https://msdn.microsoft.com/en-us/library/hh272686(v=vs.103).aspx)

- **SSDT Team Blog**

<http://blogs.msdn.com/b/ssdt/>

- **MSDN Forum**

<https://social.msdn.microsoft.com/Forums/sqlserver/en-US/home?forum=ssdt>



Growing Our Community

Questions?



Growing Our Community

Thank You for Attending



Follow [@pass24hop](https://twitter.com/pass24hop)

Share your thoughts with hashtags
[#pass24hop](https://twitter.com/hashtag/pass24hop) & [#sqlpass](https://twitter.com/hashtag/sqlpass)



Growing Our Community

Coming Up Next ...

SQL Server Agent: The Life Preserver for the Drowning DBA

Lance Tidwell

