Declarative Database Development with SSDT

Randy Pagels
Sr. Developer Tools Specialist
rpagels@microsoft.com
Agenda

Database ALM Challenges
SQL Server Data Tools (SSDT)
Connected Development
Project Based Development
Application Lifecycle & Tools
Comparison & Publishing
DAC Framework

Q&A
Too Many Details
Need to preserve data puts focus on ALTER scripts
Dependencies make even simple tasks complex

Error Detection
Errors detected late
Drift has to be accommodated

Keeping in Sync
Hard to synchronize app and database versions
Target different editions, cloud
There is a Better Way...

**Declarative**
Declarative not scripted
Model-based design and validation

**Connected or Offline**
Same tools when connected or offline
Use schema differencing for
  Incremental import and deployment
  Drift detection and reconciliation

**Application Lifecycle**
Schema and app under source code control
Test locally, deploy to SQL Server or SQL Azure
SQL Server Data Tools (SSDT)

Developer-focused toolset
Build SQL Server and SQL Azure databases
Compile refactor, and deploy databases
Single cohesive environment

Core Experiences
Connected Development
Project Based Development
Application Lifecycle & Tools

Fundamentals
Target SQL Server and SQL Azure
Detect and manage database drift
Declarative, model based database development
Integrated tools with modern language services
Connected and offline with local testing
Introducing SQL Server Data Tools

SQL Server Object Explorer

SQL Language Services

Schema Comparison

Isolated Local Database Runtime

SQL Server Data Tools

SQL Static Code Analysis

Buffered Declarative Editing

Database Publishing

Table Designer

SQL/CLR

SQL Debugging
Introducing SQL Server Data Tools

Will be integrated as quickly as possible post Visual Studio 11 RTM

- Database Unit Testing
- Data Generation
- Data Compare
- Schema Comparison
- Isolated Local Database Runtime
- SQL/CLR
- SQL Debugging
Connected Development

**SSOX**
SQL Server Object Explorer (SSOX)
Modeled after SSMS object explorer

**Modern Experience**
Modern T-SQL Coding Experience
Buffered Declarative Object Editing
Model based with error detection
Imperative Script Execution
T-SQL Intellisense

**Modern Design**
Code-behind Table Designer
View/Edit/Script Data (incl. copy/paste)
Execute/Debug T-SQL Procedures, Functions
Demo

Connected or Online Development
Visual Studio IDE
Database definition managed in VS project
Multi-target to SQL Server {2005-2012} or SQL Azure

Advancements
Language Services for T-SQL
Go To Definition/Find All References/Refactoring
F5 debugging with LocalDB

Tightly Integrated
Visualize schema differences and migrate changes
Publish direct to database or via SQL script or DACPAC
Integrated database drift detection
Point-in-time Snapshot
Drag & Drop import from SQL Server Object Explorer
Demo

Project Based Development
Application Lifecycle & Tools

MSBuild Tasks
Build
Publish
T-SQL Static Code Analysis

MSBuild Utilities
Redist SSDTBuildUtilities.msi
DB projects in build server environment
No need to install full copy of VS

Standard Integration
SSDT integrates with all standard VS SCCS providers
DACFX v3
Schema compare
Publish
Publishing your Database Changes

Target Version Aware
SQL Server 2005
SQL Server 2008
SQL Server 2008 R2
SQL Server 2012
SQL Azure

Publish
Direct
via SQL script
DACPAC snapshots
Develop, Compare & Deploy

Visual Studio DB Project (Source)
- Model
- Objects
- Write T-SQL

SQL Server Database (Target)
- Incremental database deployment
- Model Delta
- Deployment Engine
- Deployment Contributors

Schema Model
- Model
- Model Comparison

Database Reverse Engineered
- SQL
- Model
- Object Query Language (T-SQL)

Model Delta
- SQL
- Plan

Schema Model
- SQL
- Model

Properties
- SQL
- Model
- Object Query Language (T-SQL)

Model
- SQL
- Model
- Object Query Language (T-SQL)

Object Query Language (T-SQL)
- SQL
- Model
- Object Query Language (T-SQL)

SQLite
- SQL
- Model
- Object Query Language (T-SQL)

Dbschema
- SQL
- Model
- Object Query Language (T-SQL)

Model Delta
- SQL
- Plan

Deployment Engine
- SQL
- Plan

Deployment Contributors
- SQL
- Plan

Incremental database deployment
Demo

Connected or Online Development

Project Based Development

Comparison & Publishing
DAC Framework v3.0

DAC Framework
DAXFX is the core SQL redist providing modeling, reverse engineering, and deployment pipeline capabilities.

Managed Public API
Exposes verbs for DACPAC and BACPAC operations.

Command-line Tool
Exposes DACPAC verbs, project publish.

DAC Tooling
DACUnpack.exe
Windows file handler for unpacking DACPAC to disk.
DACFX Clients
SSMS, SSDT, SAMP, I&E, VS Web and DB Publishing
Comparison complete. Differences detected.
# Project Compatibility & Migration

<table>
<thead>
<tr>
<th>VS 2010 Database Projects</th>
<th>Optional conversion to SSDT Project</th>
<th>Mandatory upgrade to SSDT Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSDT SQL Server Database Projects</td>
<td>Project and Assets Roundtrip without conversion</td>
<td></td>
</tr>
</tbody>
</table>

**Project Comparison & Migration Details**
SSDT Power Tools

The Data tooling team’s vehicle to push incremental value to customers.

Currently
- New node named “Projects” inside the SQL Server Object Explorer. You can use this tree to navigate your schema, edit objects, refactor them, and add new objects.

Coming up next
- **Group by Schema**: The tool only has top-level folders for object types. We plan on adding the option to the SQL Server Object Explorer in general, in order to group objects by schema, then object type
- **Filtering Built-In Objects and External Objects**: Checkboxes to filter out built-in and external objects from the tree
- **Auto-refresh**: Detect drift from project and refresh the tree automatically.

SSDT Power Tools Website
http://tinyurl.com/7o74qzg
Summary

Developer Focused

Toolset to Author, Debug, and Publish Databases
Target SQL Server & SQL Azure
BIDS tooling installs as part of SQL 2012 when SSDT is selected
SQL CLR Development supported

Supports SQL Platform

Free web updates for SQL Server and SQL Azure releases
Works in concert with other SQL Server tooling (SAMP, SSMS)

Compatibility

Supports your development environment
Used with Visual Studio 2010 SP1 & Visual Studio 11
Easily migrates VS 2010 database projects (*.dbproj)
Visit Team System Café Today!

The door is open 24/7 at the café! Stop by for a cup of joe and delicious tidbits on AppDev topics!

http://www.teamsystemcafe.net
Resources

Online Setup
http://tinyurl.com/878lonn ➔ Get It Today!

Team blog
http://tinyurl.com/7dak35o

MSDN Forum
http://tinyurl.com/7ox5tf2

SSDT vs. VS2010 Database Projects
http://tinyurl.com/76nukaz

Articles
MSDN Magazine Sept 2011
The "Juneau" Database Project
http://tinyurl.com/86bhsko