

Armin Neudert  
TMN Systemberatung GmbH

Visual Studio ALM Days 2012

# DB LIFECYCLE MANAGEMENT MIT DEN SQL SERVER 2012 DATA TOOLS UND TFS 2012

# Armin Neudert

- ◉ Geschäftsführender Gesellschafter
- ◉ Schwerpunkte
  - Technical Lead SQL Server
  - Softwareprozesse / ALM
  - Consulting/Coaching
  - Projektleitung
- ◉ Community
  - Gründer/Lead SQL PASS Stuttgart
  - Redner auf Konferenzen, Usergroup Meetings
  - Diverse Demo-Videos, Webcasts
- ◉ MCTS, MCP

# Ziele dieses Talks

- ◉ Überblick über die SQL Server 2012 Data Tools
- ◉ Unterschiede zwischen Anwendungs- und DB-Entwicklung aufzeigen
- ◉ Aufzeigen, wie die DB-Entwicklung gleichgestellt werden kann – Demos!
- ◉ Integration in das Application Lifecycle Management
- ◉ Start-up-Infos und Quellen für weitere Infos

# Agenda SSDT

- ④ DB-Entwicklung & ALM
- ④ Entwicklung mit DB-Projekten
- ④ Deployment mit DAC
- ④ Connected Development
- ④ Weitere Infos und Ausblick

# Datenbankentwicklung & ALM

# Application Lifecycle Management (ALM)

- ⦿ Management des gesamten Produktlebenszyklus
- ⦿ Technisch und Business
- ⦿ Entwicklungsprozess
- ⦿ Testing, Architektur, SCC, Reporting, ...

Datenbanken häufig vernachlässigt!

# Anwendungs- vs. DB-Entwicklung

- ◉ Häufig Ungleichgewicht zwischen Anwendungs- und Datenbankentwicklung
- ◉ Codierung von Änderungen, nicht von aktuellem Zustand (wenig verbreitet: andere Tools)
- ◉ Agile Entwicklung im DB-Umfeld schwierig

Wie können uns die SQL Server Data Tools in Zusammenarbeit mit dem Team Foundation Server hierbei helfen?

# Was ist „in der Box“?

## SQL Server Data Tools - Development

SQL Server  
Object  
Explorer

SQL Static  
Code Analysis

Database  
Publishing

SQL  
Language  
Services

Buffered  
Declarative  
Editing

Table  
Designer

Schema  
Comparison

Isolated Local  
Database  
Runtime

SQL/CLR  
Project  
System

SQL  
Debugging

## Verfügbar nach Visual Studio 2012 RTM

Database  
Unit Testing

Data  
Generation

Data Compare

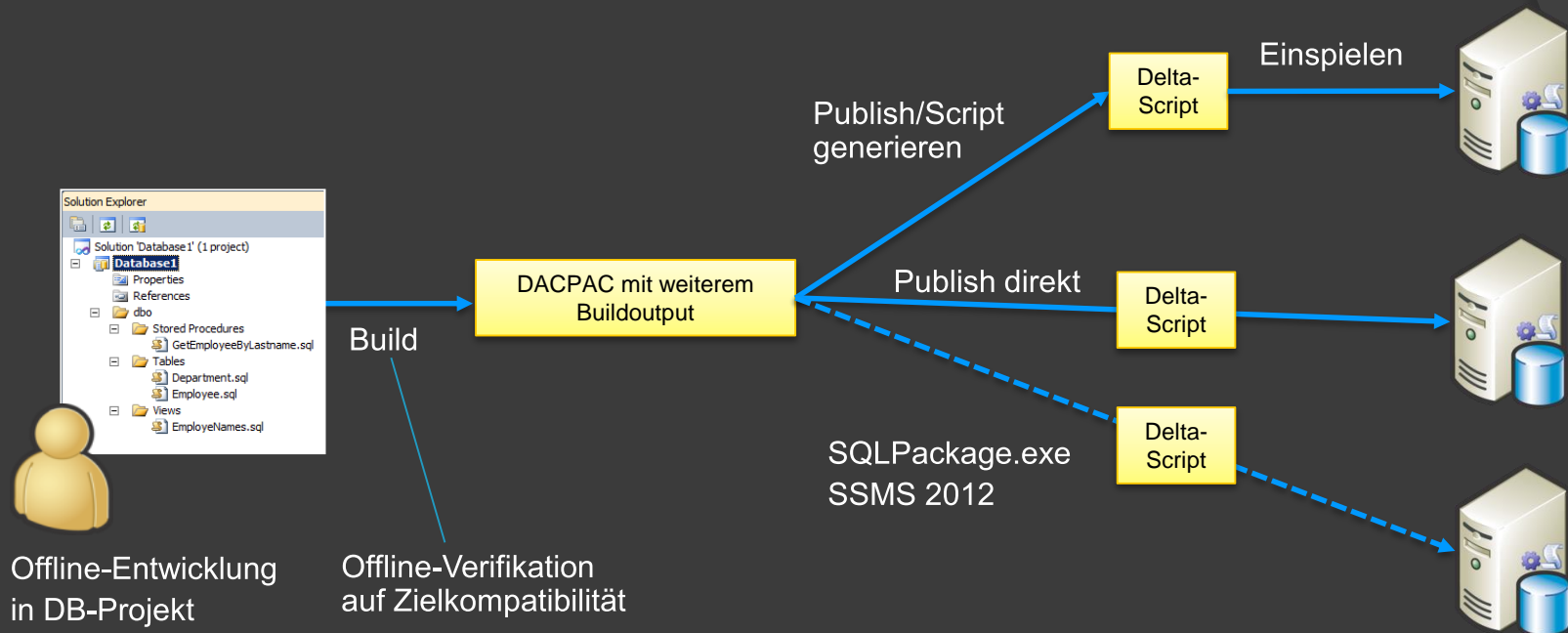


# Entwicklung mit DB-Projekten

# Demo! Was werden Sie sehen?

- ◉ Schema-Import einer existierenden Datenbank
- ◉ Das Datenbankprojekt, Build, Buildoutput
- ◉ Schema-Änderungen und Script-Deployment
- ◉ Änderung in Sandbox-DB und Synch. in das Projekt
- ◉ Statische Codeanalyse
- ◉ Refactoring
- ◉ Go To Definition, Find All References
  
- ◉ Deployment auf LocalDB
- ◉ Debugging

# Build und Publish/Deploy



- Ziel DBs in beliebiger Version
- Green und Brown Field Deployment
- Ziel-SQL-Server >= 2005, SQL Azure

# Weitere Anmerkungen

- ⦿ „Externe“ Änderungen sind möglich und verwaltbar, Commandline-Tool/SSMS
- ⦿ Pre- und Postdeploy
- ⦿ „NULL -> NOT NULL“ – Änderungen
- ⦿ Nicht nur für DB-bezogene Objekte

# ALM-Integration

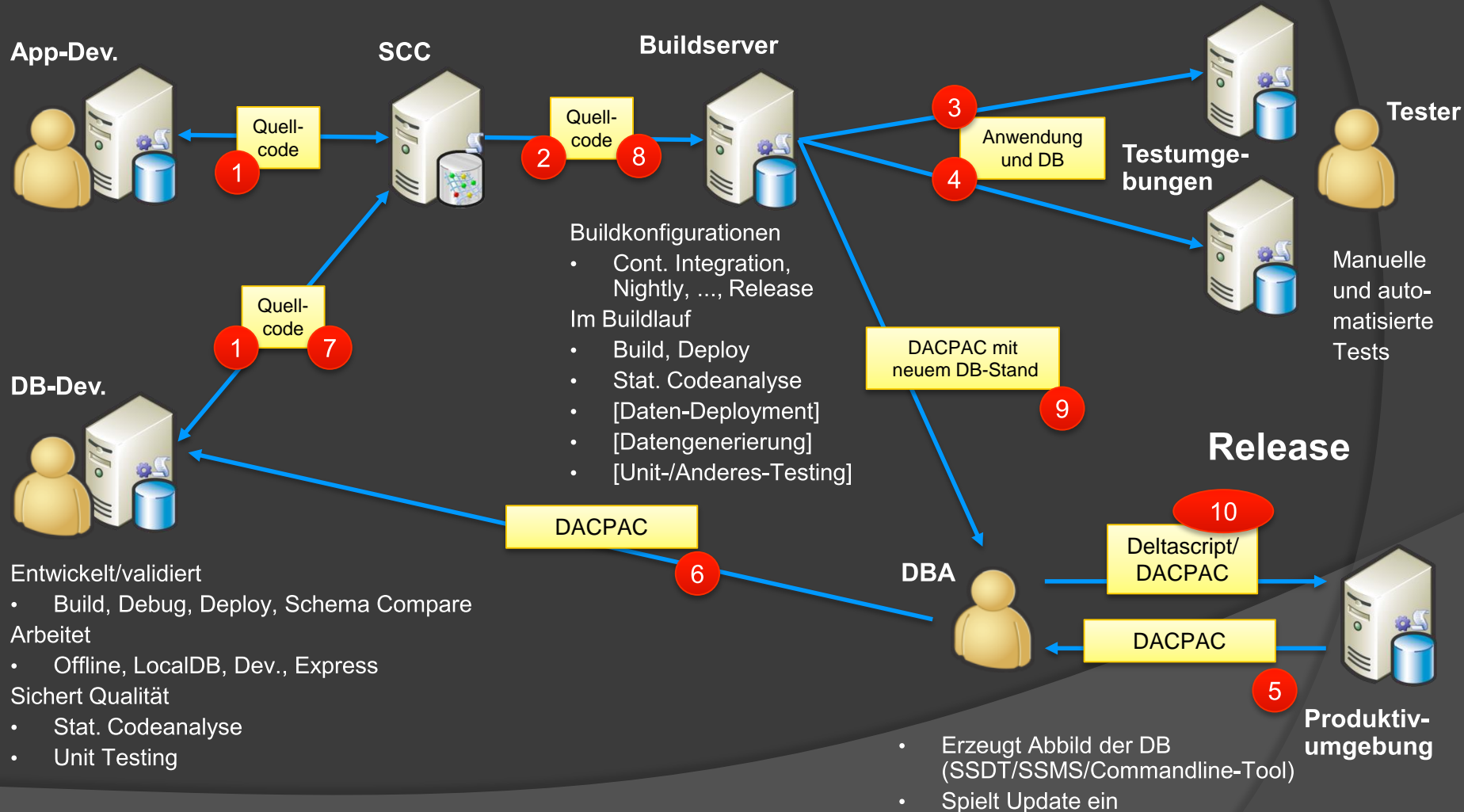
- ⦿ Snapshots – eine ganz einfache Möglichkeit
- ⦿ Quellcodeverwaltung – eine gute Sache
  - Anbindung über MSSCCI-Schnittstelle
- ⦿ Team Foundation Server – richtig gut!
  - SCC mit Changesets, Branching/Merging, globale DB-Elemente mit Dependency Management, ...
  - Arbeitsaufgaben
  - Team Build, Testlabs, ...

# ALM-Integration

## Entwicklung

## Validierung/Integration

## Test



# Agile DB-Entwicklung

- ⦿ Ist möglich!
  - ⦿ Frühzeitige Validierung / Integration
  - ⦿ Testautomatisierung, Codeanalyse
  - ⦿ Refactoring
- 
- ⦿ Continuous Integration, Delivery und Deployment

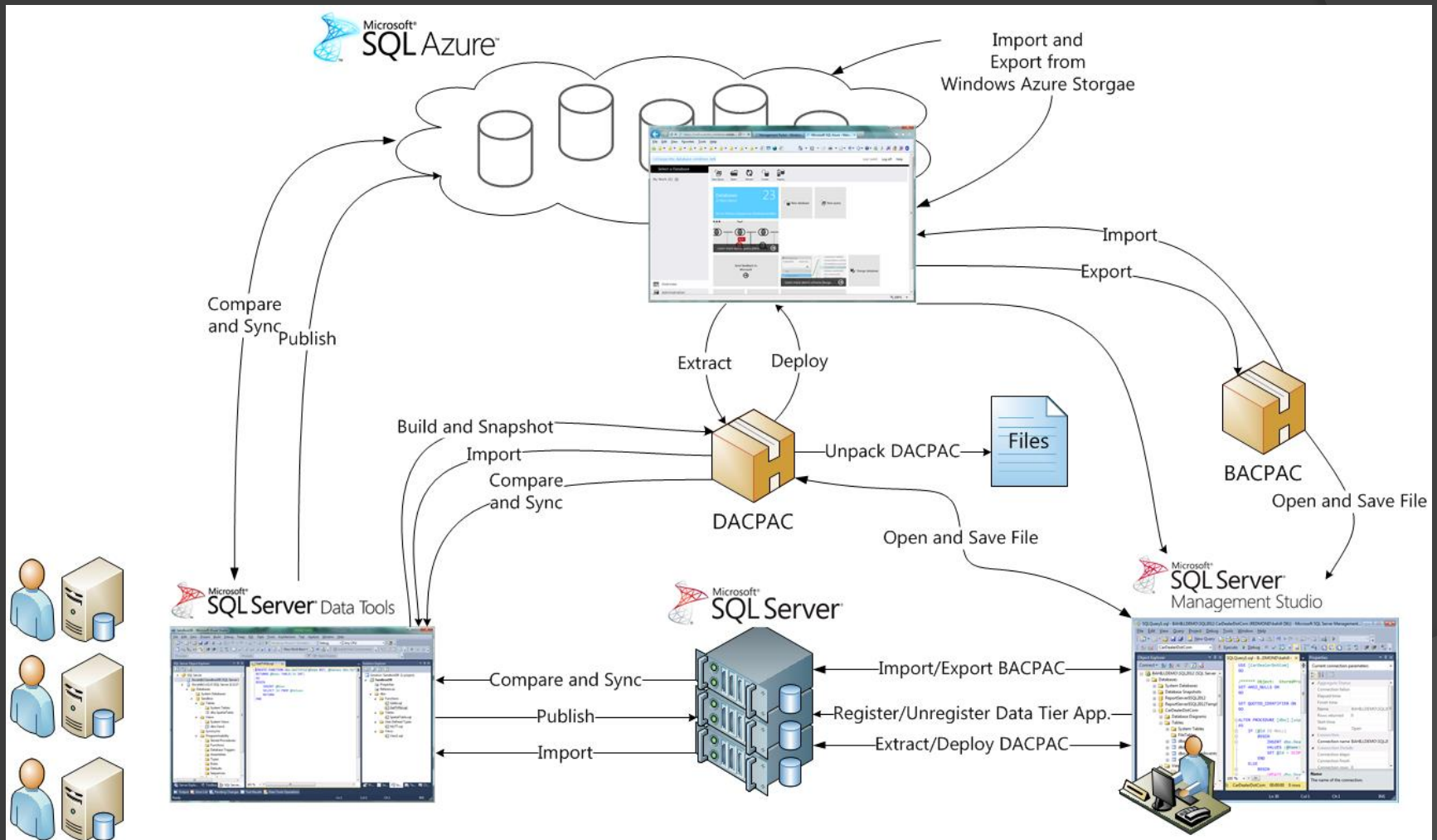
**Mein DAC, Dein BAC**  
**Deployment mit DAC**



# Überblick DAC

- ⦿ DAC = Data-tier Application Component (Datenebenenanwendung)
- ⦿ Nichts neues, aber jetzt deutlich besser/erweitert
- ⦿ Auslieferungseinheit: .dacpac
  - Zip-Datei
  - XML-Deklarationsdateien und –Schemabeschreibung
  - Können Daten enthalten Daten (nicht im Build-Output)
  - Wenn aus SSDT-Projekt erstellt: Refactorlog, Pre- und Postdeploy-Scripts
- ⦿ BACPAC = DACPAC + Daten
  - Wo ist der Unterschied?
- ⦿ Bevorzugte Deployment- und Sync-Methode für SQL Azure

# Überblick DAC



# Connected Development

# Neue Funktionalitäten in Visual Studio

- ◎ SQL Server Object Explorer
- ◎ Verbesserter T-SQL-Editor
  - Intellisense, Go To Definition, ...
- ◎ Direkte „live“-Änderungen gepuffert
  - Auswirkungenanalyse / Validierung
  - Skript-Generierung
- ◎ Applikationsentwickler bekommen das Tooling des Management Studios für T-SQL

# Quick Start und weitere Infos

# Quick Start

- ⦿ Zwei „Versionen“ - für VS 2010 und 2012
- ⦿ SSDT integrieren sich in IDE, bzw. bringen die VS IDE mit
- ⦿ Projekte sind in VS 2010/2012 wechselseitig bearbeitbar
- ⦿ Wenn SSDT schon installiert sind:  
Update auf Nov. 2012 Release
- ⦿ Releasezyklus
  - Asynchron zu SQL Server Releases
  - Jedes neue SQL Azure Feature wird unterstützt, sobald es in live geht

# Quick Start

- ① Projektübernahme aus VSDB möglich
- ① SSDT Power Tools installieren
- ① Headless Buildserver Support

- ◉ Developer Center/Download SSDT
  - <http://msdn.microsoft.com/en-us/data/tools.aspx>
- ◉ Blog SQL Server Data Tools Team
  - <http://blogs.msdn.com/b/ssdt/>
- ◉ Blog Gert Drapers, Principal Group Program Manager SSDT
  - <http://blogs.msdn.com/b/gertd/>
- ◉ MSDN-Forum
  - <http://social.msdn.microsoft.com/Forums/en-US/ssdt/threads>
- ◉ Whitepaper zu VSDB im ALM-Kontext
  - <http://vsdatabaseguide.codeplex.com/>
- ◉ Folgen Sie mir mir ... <http://blogs.tmn-net.com/an/>



# Unser Resumee

- ◎ Unsere Erfahrungen / Kundenerfahrungen
- ◎ Die 80/20-Regel gilt auch hier ;-)
- ◎ Mehrwert überwiegt deutlich

# Fragen



Mail: neudert ( at ) tmn-systemberatung.de

Blog: <http://blogs.tmn-net.com/an>

Xing: [www.xing.com/profile/Armin\\_Neudert](http://www.xing.com/profile/Armin_Neudert)

LinkedIn: [de.linkedin.com/in/aneudert](http://de.linkedin.com/in/aneudert)

FB: [www.facebook.com/aneudert](http://www.facebook.com/aneudert)

Twitter: [www.twitter.com/aneudert](http://www.twitter.com/aneudert)

Web: [www.tmn-systemberatung.de](http://www.tmn-systemberatung.de)

# Die SSDT sind für alle da!

- ◎ Für alle Zielgruppen
  - Offline-Entwicklung und –Validierung
  - Viele unterstützende Features: Language Services, ...
  - ALM-Integration
- ◎ T-SQL-Entwickler / DBAs
  - Durchführen von gepufferten Änderungen
  - Professionelle projektorientierte Entwicklungsumgebung
  - Management Studio dennoch nicht überflüssig
- ◎ Anwendungsentwickler
  - Anwendungs- und DB-Projekte in einer Solution
  - Visual Studio IDE muss (fast) nie verlassen werden

# Resumee

- ⦿ Eine völlig andere Dimension der DB-Entwicklung
- ⦿ Kein Bruch mehr zwischen Anwendungs- und DB-Entwicklung
- ⦿ Agile DB-Entwicklung ist möglich
- ⦿ Mit TFS volle ALM-Integration

# Vielen Dank!



Armin Neudert  
TMN Systemberatung GmbH