

# Migration der Logik von Access nach SQL Server

AEK 17 – Herbst 2014  
Bernd Jungbluth

[www.berndjungbluth.de](http://www.berndjungbluth.de)

## ■ Zur Person

### ■ Bernd Jungbluth

### ■ IT-Erfahrung

- SQL und Datenbanken seit 1991
- SQL Server seit Version 7.0

### ■ Freiberuflicher Berater und Entwickler

- Administration SQL Server
- Entwicklung und Optimierung von SQL Server-Datenbanken
- Datawarehouse-Systeme nach Bill Inmon
- SQL Server Reporting Services
- SQL Server Integration Services
- Migration Access nach SQL Server

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Migration einer Access-Applikation

#### ■ Logik einer Access-Applikation

- Tabellen mit Datentypen und Tabellenfunktionalität
- Basisfunktionalität zur Datenverwaltung in Tabellen und Formularen
- SQL-Anweisungen in Access-Abfragen und Datenquellen
- VBA in Formularen, Berichten und Modulen
- Makros

#### ■ Migration einer Access-Applikation

- Tabellen und Daten einer Access-Datenbank
- Datentypen
- Gültigkeitsregeln und Standardwerte
- Eingabe erforderlich und Leere Zeichenfolge
- Primärschlüssel und Fremdschlüssel-Beziehungen
- Eindeutige Indizes

### ■ Migration einer Access-Applikation / SQL Server Migrations-Assistent für Access

#### ■ SQL Server Migrations-Assistent für Access

- Kostenfreier Download mit notwendiger Lizenzierung
- Migriert ab Access 97 für SQL Server 2005 – 2014 und SQL Azure
- Aktuelle Version 6.0

#### ■ Möglichkeiten

- Migration aller sowie einzelner Tabellen und Abfragen
- Einbinden der erstellten SQL Server-Tabellen in Access-Datenbank
- Ergebnisbericht mit entsprechenden Auswertungsmöglichkeiten
- Simulation der Migration mit Ergebnisbericht und Aufwandskalkulation

#### ■ Hinweise

- Migriert nur einfache Auswahlabfragen ohne Access-Funktionen
- Setzt Standardwert bei »Ja/Nein«-Feldern ohne Standardwert auf »Nein«

- Migration einer Access-Applikation / SQL Server Migrations-Assistent für Access

- Demo

- Migration einer Access-Datenbank  
mit SQL Server Migrations-Assistent für Access

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen

#### ■ Access und SQL Server-Tabellen

- Einbinden der SQL Server-Tabellen in Access
  - Manuell per Tabellenverknüpfungsmanager
  - Automatisiert mittels eigener VBA-Routine
- ODBC-Verbindung erforderlich

#### ■ ODBC-Verbindung

- Installation des ODBC-Treibers auf jedem Client notwendig
- Konfiguration einer ODBC-Datenquelle – »DSN«
- Beinhaltet die Verbindungszeichenfolge – »connectionstring«
  - Konfiguration der ODBC-Datenquelle auf jedem Client erforderlich
- Alternativ Konfiguration ohne ODBC-Datenquelle – »DSN-less«
  - Verbindungszeichenfolge zur ODBC-Datenquelle in lokaler Tabelle im Frontend der Access-Applikation

### ■ Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen / ODBC-Treiber

#### ■ »SQL Server«

- Treiber für SQL Server 2000
- Funktioniert mit SQL Server-Versionen ab SQL Server 2000
- Unterstützt keine Funktionen der neueren SQL Server-Versionen

#### ■ »SQL Server Native Client x.0«

- SQL Server Native Client 9.0 für SQL Server 2005
- SQL Server Native Client 10.0 für SQL Server 2008 und 2008 R2
- SQL Server Native Client 11.0 für SQL Server 2012 und 2014
- Jede Version abwärtskompatibel bis SQL Server 2005

#### ■ »Microsoft ODBC Driver 11 for SQL Server«

- Aktueller Treiber seit 2013
- Abwärtskompatibel bis SQL Server 2005

- Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen

- Demo

- Eingebundene SQL Server-Tabellen in Access

### ■ Analyse der Access/SQL Server-Applikation / Aussagen und Gegendarstellung

- »Die Migration ist schnell erledigt.«
  - Eher selten.
  - Bei eingebundenen Tabellen ist die Performance in der Regel schlecht.
  - Nicht alle Funktionen werden weiterhin fehlerfrei funktionieren.
  
- »Access und SQL Server stammen von einem Hersteller.«
  - Stimmt.
  - Hat aber keinerlei Bedeutung oder gar Auswirkung.
  
- »Access und SQL Server sind aufeinander abgestimmt.«
  - Stimmt nicht.
  - Access arbeitet mit per ODBC eingebundenen Tabellen.
  - Der Rest ist Sache des ODBC-Treibers.

### ■ Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen

#### ■ Eingebundene Tabellen

- Tabellen ohne Primärschlüssel als Snapshot
- Tabellen mit Primärschlüssel als Dynaset

#### ■ Snapshot

- Ändern der Daten nicht möglich
- Liest zur Ausgabe von Daten alle Datensätze der Tabelle
- Keine automatische Aktualisierung der Ausgabe

#### ■ Dynaset

- Ändern der Daten möglich
- Liest zur Ausgabe von Daten zuerst die Primärschlüssel der Datensätze
- Liest dann die restlichen Datenspalten der Datensätze
- Automatische Aktualisierung der Ausgabe

### ■ Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen und Sperren

#### ■ Eingebundene Tabellen und Sperren

- Sperren bei Lesezugriffen an SQL Server-Tabellen
- Lesesperren auf verschiedenen Sperrebenen

#### ■ Sperrebenen

- Datensatz – »Row Lock«
- Datenseite – »Page Lock«
- Tabelle – »Table Lock«
- Sperrausweitung von Datensatz auf Datenseite auf Tabelle
- Sperrausweitung – »Lock Escalation«

#### ■ Datenseiten

- 8096 Byte große Einheit für Lese- und Schreibvorgänge
- Beinhaltet Daten sowie 96 Byte Informationen zur Datenseite

### ■ Analyse der Access/SQL Server-Applikation / Eingebundene Tabellen und Sperren / Sperrtypen

#### ■ Shared Lock – S

- »Freigegebene Sperre«
- Aktiviert für lesenden Zugriff
- Lesende Zugriffe anderer Transaktionen möglich
- Schreibende Zugriffe anderer Transaktionen nicht möglich

#### ■ Intent Shared Lock – IS

- »Beabsichtigte freigegebene Sperre«
- Aktiviert vor Platzierung eines Shared Locks
- Auf Ebene der Tabelle und der Datenseite
- Ermöglicht eine Sperrausweitung auf Datenseite oder Tabelle
- Lesende Zugriffe anderer Transaktionen möglich
- Schreibende Zugriffe anderer Transaktionen nicht möglich

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Analyse der Access/SQL Server-Applikation / Analyse mit Extended Events

#### ■ Extended Events

- »Erweiterte Ereignisse« oder »XEvents«
- Nachfolger von SQL Server Profiler
- Sammeln von Informationen zu ausgewählten Ereignissen
- Auswerten der gesammelten Informationen
- Zur Fehlererkennung, Überwachung, Performance-Analyse, u.a.
- Benutzeroberfläche im SQL Server Management Studio

#### ■ Ergebnis

- Ein XML-Dokument
- Ausgabe in bis zu sechs verschiedene Ziele
- Direkte Ausgabe im SQL Server Management Studio
- Auswerten der Ergebnisdaten im SQL Server Management Studio
- Auswerten der Ergebnisdaten per T-SQL und XQuery

- Analyse der Access/SQL Server-Applikation / Analyse mit Extended Events

- Demo

- Analyse des Sperrverhaltens bei eingebundenen Tabellen mit den Extended Events

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Analyse der Access/SQL Server-Applikation / Access-Abfragen mit eingebundenen Tabellen

#### ■ Access-Abfragen mit eingebundenen Tabellen

- Performance abhängig vom verwendeten Treiber
- In der Regel schlechte Performance
- Betrifft Access-Abfragen mit Access-Funktionalität
- Betrifft Access-Abfragen mit besonderer SQL-Syntax

#### ■ Ursache

- Übermittlung der SQL-Anweisung von Access an SQL Server
- Abhängig von der Komplexität der SQL-Anweisung und vom Treiber

#### ■ Mögliche Arten der Übermittlung

- Übergabe der unveränderten SQL-Anweisung an SQL Server
- Aufteilen der SQL-Anweisung in mehrere SQL-Anweisungen
- Anfordern der Daten aller in der Abfrage enthaltenen Tabellen

- Analyse der Access/SQL Server-Applikation / Access-Abfragen mit eingebundenen Tabellen

- Demo

- Analyse  
Übermittlung der SQL-Anweisung einer Access-Abfrage  
von Access an SQL Server

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Gründe zur Migration der Logik

#### ■ Gründe zur Migration der Logik auf SQL Server

- Sperrverhalten bei eingebundenen Tabellen
- Abfrageverhalten bei Access-Abfragen
- Fehler wegen abweichendem Verhalten
- Eventuelle Fehler wegen unterschiedlicher Syntax

#### ■ Aufgaben

- Fehlerbehebung durch Anpassen von Abläufen und Syntax
- Reduzieren der Lesevorgänge auf das Notwendigste

#### ■ Mögliche Varianten der Migration

- »Pragmatischer Ansatz«
- »Königsweg«

### ■ Gründe zur Migration der Logik / Pragmatischer Ansatz vs. Königsweg

#### ■ »Pragmatischer Ansatz«

- Basiert auf eingebundenen Tabellen
- Prüfen der Applikation auf Funktionalität und Performance
- Optimierung, Fehlerbehebung und Neuentwicklung auf SQL Server
  - Aufwand nicht kalkulierbar
  - Applikation schnell wieder verfügbar

#### ■ »Königsweg«

- Keine eingebundenen Tabellen
- Verlagern der kompletten Logik auf SQL Server
- Datenermittlung und Datenverarbeitung per Gespeicherte Prozeduren
  - Hoher Migrationsaufwand
  - Aufwand kalkulierbar
  - Flexibel für die Zukunft

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Datenverarbeitung mit Gespeicherten Prozeduren

#### ■ Gespeicherte Prozeduren

- Daten ermitteln und ausgeben
- Daten hinzufügen, ändern und löschen
- Sehr gute Performance durch Ausführungspläne

#### ■ Entwicklung

- Kann mehr als nur eine SQL-Anweisung enthalten
  - Steuerung vom Programmfluss, Schleifenverarbeitung
  - Ein- und Ausgabeparameter, Variablen, Fehlerbehandlung
- Programmierung in T-SQL

#### ■ Möglichkeiten

- Gewährleistung der Datenkonsistenz
- Einhaltung von Geschäftsregeln

### ■ Datenverarbeitung mit Gespeicherten Prozeduren / Ausführungsplan

- Erste Ausführung einer Gespeicherten Prozedur
  - SQL-Syntaxprüfung und Existenzprüfung der SQL Server-Objekte
  - Erstellen und Speichern des Ausführungsplans im Arbeitsspeicher
  - Ausführen der Gespeicherten Prozedur anhand des Ausführungsplans
  
- Ausführungsplan
  - »Routenplaner« zur Datenermittlung
  - Beinhaltet Reihenfolge der Zugriffe auf Tabellen und Indizes
  - Liegt im Arbeitsspeicher – im »Plancache«
  
- Ab der zweiten Ausführung einer Gespeicherten Prozedur
  - SQL-Syntaxprüfung und Existenzprüfung der SQL Server-Objekte
  - Ermitteln vom Ausführungsplan im Arbeitsspeicher
  - Ausführen der Gespeicherten Prozedur anhand des Ausführungsplans

### ■ Datenverarbeitung mit Gespeicherten Prozeduren / Vorteile

#### ■ Sicherheit

- Hohe Sicherheit beim Datenzugriff
- Ermitteln und Verarbeiten von Daten nur per Gespeicherte Prozeduren
- An Tabellen weder Lese- noch Schreibrechte
- Lediglich EXECUTE-Rechte an Gespeicherten Prozeduren

#### ■ Geschäftslogik

- Zentrale Stelle der Geschäftslogik
- Änderungen an Geschäftslogik direkt verfügbar
- Ausliefern vom Frontend nur bei Strukturänderungen notwendig

#### ■ Flexibilität

- Unterschiedliche Techniken als Frontend denkbar
- Beinhalten lediglich Aufrufe von Gespeicherten Prozeduren

- **Datenverarbeitung mit Gespeicherten Prozeduren**

- **Demo**

- Erstellen einer Gespeicherten Prozedur

## ■ Agenda

- Migration einer Access-Applikation
  - Logik einer Access-Applikation
  - Migration einer Access-Applikation
  - SQL Server Migrations-Assistent für Access
  
- Analyse der Access/SQL Server-Applikation
  - Eingebundene Tabellen
  - Access-Abfragen
  - Analyse mit Extended Events
  
- Migration der Logik von Access nach SQL Server
  - Gründe zur Migration der Logik
  - Datenverarbeitung mit Gespeicherten Prozeduren
  - Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Gespeicherte Prozeduren und Pass Through-Abfragen

#### ■ Access und Gespeicherte Prozeduren

- Aufruf per Pass Through-Abfrage
- ODBC-Verbindung erforderlich

#### ■ Pass Through-Abfrage

- Direkter Durchgriff auf das Backend – »Pass Through«
- Keine Beeinflussung der SQL-Anweisung durch den Treiber
- Enthält SQL-Anweisung in der Syntax vom Backend – »T-SQL«
- Aufruf von Gespeicherten Prozeduren empfehlenswert

#### ■ Pass Through-Abfragen und Sperren

- Daten nur schreibgeschützt verfügbar
- Kein automatisches Aktualisieren der Daten
- Einmaliges Aktivieren von Lesesperren

- **Gespeicherte Prozeduren und Pass Through-Abfragen**

- **Demo**

- Erstellen einer Pass Through-Abfrage zum Aufruf einer Gespeicherten Prozedur

### ■ Gespeicherte Prozeduren und Pass Through-Abfragen / Pass Through-Abfragen

- **Pass Through-Abfrage als Auswahlabfrage**
  - Datenherkunft für Comboboxen, Listboxen und Berichte
  - Datenherkunft für Formulare
  
- **Pass Through-Abfrage als Aktionsabfrage**
  - Direkte Datenverarbeitung auf dem Backend
  - Keine Datenverarbeitung über eingebundene Tabellen
  - Bessere Performance und weniger Netzwerklast
  - Aufruf von Gespeicherten Prozeduren empfehlenswert
  
- **Pass Through-Abfrage und Parameter**
  - Keine direkte Parameterübergabe möglich
  - Parameterübergabe nur in VBA realisierbar
  - Änderung des Quellcodes der Pass Through-Abfrage

### ■ Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Demo

- Erstellen einer Gespeicherten Prozedur mit Parameter
- Erstellen einer Pass Through-Abfrage
- Anpassen des Quellcodes zur Übergabe des Parameterwerts

- **Gespeicherte Prozeduren und Pass Through-Abfragen / Ausführungspläne**

- **Demo**

- Vergleich der Ausführungspläne von Gespeicherten Prozeduren und SQL-Anweisungen

### ■ Gespeicherte Prozeduren und Pass Through-Abfragen / Pass Through-Abfragen in VBA

#### ■ Pass Through-Abfragen in VBA

- Per »OpenRecordset« als Datenherkunft
- Per »Execute« zur Datenverarbeitung
- Mit in Access gespeicherten Pass Through-Abfragen
- Mit temporären Pass Through-Abfragen

#### ■ Temporäre Pass Through-Abfragen

- Prädestiniert für Pass Through-Abfragen mit Parametern
- Reduziert die Anzahl der Pass Through-Abfragen

#### ■ Datenherkunft für Recordsets

- Zum Verarbeiten der Daten des Recordsets
- Als Datenherkunft von Formularen, Listboxen und Comboboxen
- Nicht möglich als Datenherkunft für Berichte

### ■ Gespeicherte Prozeduren und Pass Through-Abfragen

### ■ Demo

- Temporäre Pass Through-Abfrage als Datenherkunft
- Temporäre Pass Through-Abfrage zur Datenverarbeitung
- Datenherkunft für Berichte
- Migration eines Formulars mit Unterformular

### ■ Zusammenfassung und Fazit

### ■ Migration der Logik

- Logik der Access-Applikation in Gespeicherte Prozeduren
- Gespeicherte Prozeduren per Pass Through-Abfragen in Access

### ■ Performance

- Ausführung der SQL-Anweisungen direkt im SQL Server
- Keine Beeinflussung der SQL-Anweisungen durch den Treiber
- Weniger Netzwerklast
- Wiederverwendung von Ausführungsplänen

### ■ Vorteile

- Zentrale Stelle der Geschäftslogik
- Frontends unterschiedlicher Technologien möglich
- Hohe Sicherheit beim Datenzugriff

## ■ Hinweise

### ■ ADO und OLE DB

- Abkündigung vom Treiber SQLOLEDB zum Sommer 2017
- SQLOLEDB – »Microsoft OLE DB Provider for SQL Server«
- MSDASQL als Alternative möglich
- MSDASQL – »Microsoft OLE DB Provider for ODBC Drivers«

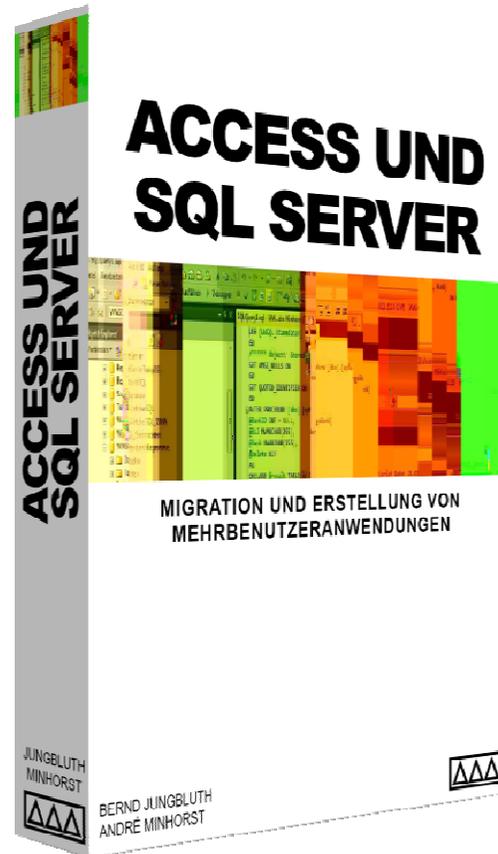
### ■ Extended Events

- Mehr Informationen im Vortrag zur SNEK2
- Download: <http://www.donkarl.com/Downloads/SNEK/index.htm>

### ■ Hinweise

- Seminar »Migration Access nach SQL Server«
- Buch »Access und SQL Server«
- Infos: [www.berndjungbluth.de](http://www.berndjungbluth.de)

■ Danke



**Noch Fragen?**