Admin-Handbuch

sqmMergeGenerator

T-SQL MERGE statements from a GUI: upsert and SCD type 2

dtcSoftware · Uwe Janke · www.powershelldba.de
Stand: 2026-11-05

sqmMergeGenerator: Handbuch

T-SQL-`MERGE`-Statements per Mausklick: Server waehlen, Quell- und Zieltabelle auswaehlen, Spalten im Stil des SSIS-Mapping-Editors zuordnen, Schluesselspalten anhaken. Das Statement wird bei jeder Aenderung sofort neu erzeugt. Unterstuetzt einfachen Upsert (SCD Typ 1) und Slowly Changing Dimensions Typ 2 mit Aenderungsart pro Spalte.

Zielgruppe: SQL-Server-DBAs und DWH-/ETL-Entwickler, die `MERGE`-Statements fuer Beladungen, Abgleiche oder Dimensionstabellen brauchen. Versionshistorie: CHANGELOG.md, Kurzuebersicht: README.md.

Stand: 2026-11-05, sqmMergeGenerator 0.1.1.0

Inhalt

1. Ueberblick
2. Voraussetzungen und Installation
3. GUI Schritt fuer Schritt
4. Modus Upsert / SCD Typ 1
5. Modus SCD Typ 2
6. Optionen im Detail
7. Skripting ohne GUI
8. Funktionsreferenz
9. Test und Freigabe
10. Troubleshooting und Einschraenkungen
11. Sicherheitshinweise

1. Ueberblick

Ein korrektes `MERGE` von Hand zu schreiben ist fehleranfaellig: Spaltenlisten in vier Klauseln synchron halten, NULL-sichere Aenderungserkennung, `HOLDLOCK` gegen Race Conditions, doppelte Schluessel in der Quelle (Fehler 8672), Identity-Spalten, und bei einer historisierten Dimension zusaetzlich das Schliessen der alten und Einfuegen der neuen Version. sqmMergeGenerator liest die Metadaten beider Tabellen und schreibt das Statement vollstaendig.

Szenario	Modus	Ergebnis
Staging-Tabelle in eine Zieltabelle uebernehmen, Ziel = Abbild der Quelle	Upsert / SCD 1	`MERGE` mit `UPDATE` nur bei echter Aenderung, `INSERT`, optional `DELETE`
Dimension mit Historie (Gueltig ab/bis, Aktuell-Flag)	SCD Typ 2	Alte Version schliessen, neue einfuegen, Typ-1-Spalten in allen Versionen ueberschreiben

Statement im Build- oder Deployment-Skript erzeugen	beide	`New-sqmMergeStatement` ohne GUI und ohne Datenbankverbindung

Das Werkzeug **fuehrt nichts automatisch aus**. Es erzeugt Text. Ausgefuehrt wird nur ueber den Button *Ausfuehren ...* nach Rueckfrage, oder indem Sie das Statement kopieren bzw. speichern.

---

2. Voraussetzungen und Installation

- Windows PowerShell 5.1 oder PowerShell 7 (unter Windows; die GUI benoetigt WinForms).
- **Keine** Modulabhaengigkeit: kein dbatools, kein SqlServer-Modul. Die Verbindung laeuft ueber `System.Data.SqlClient` aus dem .NET Framework. Damit laeuft das Werkzeug auch auf abgeschotteten Servern ohne PSGallery-Zugang.
- SQL Server 2012 oder neuer (`MERGE`, `THROW`). Getestet gegen SQL Server 2022.
- Rechte zum Erzeugen: `VIEW DEFINITION` bzw. Lesezugriff auf die Katalogsichten der beteiligten Datenbanken. Zum Ausfuehren: `SELECT` auf der Quelle, `INSERT`/`UPDATE`/`DELETE` auf dem Ziel, bei Identity-Insert zusaetzlich `ALTER` auf der Zieltabelle.

Installation

Aus der PowerShell Gallery:

```
Install-Module sqmMergeGenerator -Scope CurrentUser
```

Aus dem Repository:

```
Install.cmd               # AllUsers wenn als Administrator gestartet, sonst CurrentUser
Install.cmd CurrentUser
```

Danach:

```
Import-Module sqmMergeGenerator
Show-sqmMergeGeneratorGui
```

Ohne Installation direkt aus dem Ordner: `Start-sqmMergeGenerator.cmd` (Doppelklick).

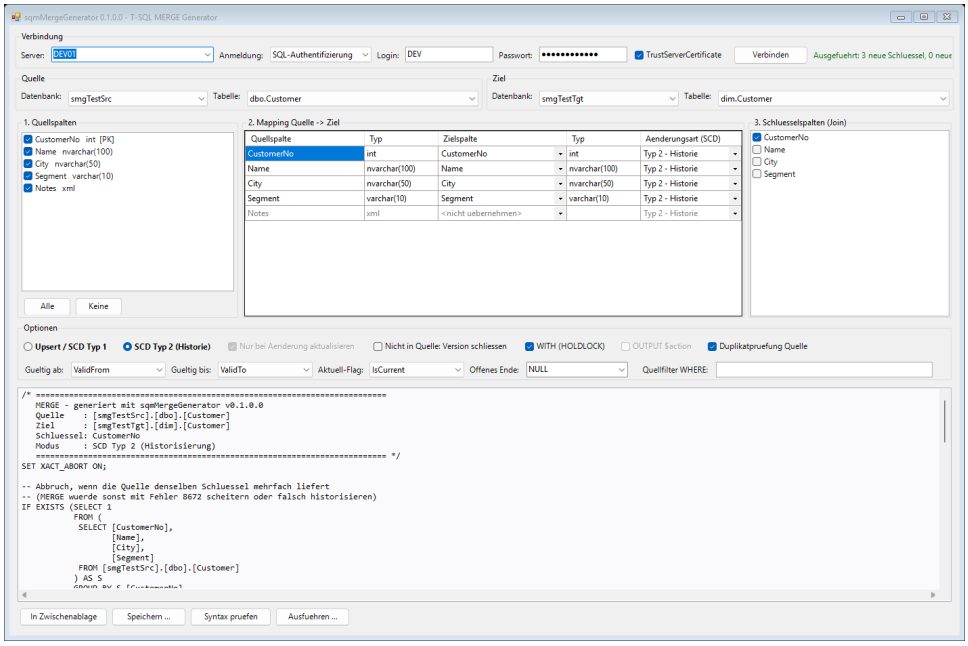
Startmenue-Eintrag: Die Installation legt den Eintrag `sqmMergeGenerator` im Startmenue an (AllUsers: fuer alle Benutzer, CurrentUser: nur fuer den aktuellen Benutzer). Er startet die GUI ohne sichtbares Konsolenfenster. Das Anheften an Start ist unter Windows 10/11 nicht automatisierbar: einmalig per Rechtsklick auf den Eintrag > *An Start anheften*.

Konfiguration

Die zuletzt verwendeten Server und der Login-Name werden in `%APPDATA%\sqmMergeGenerator\config.json` gespeichert. **Das Passwort wird nie gespeichert.**

---

3. GUI Schritt fuer Schritt



3.1 Verbindung

1. **Server** eintragen oder aus der Liste der letzten Server waehlen. Benannte Instanzen und Ports wie gewohnt (`SQL01\INST1`, `SQL01,14330`).
2. **Anmeldung**: Windows-Authentifizierung oder SQL-Authentifizierung (Login + Passwort).
3. **TrustServerCertificate** anhaken, wenn der Server ein selbstsigniertes Zertifikat nutzt.
4. **Verbinden**. Die Datenbanklisten fuer Quelle und Ziel werden gefuellt.

Direktstart mit Verbindung:

```
Show-sqmMergeGeneratorGui -SqlInstance SQL01
Show-sqmMergeGeneratorGui -SqlInstance SQL01 -SqlCredential (Get-Credential)
```

3.2 Quelle und Ziel

Pro Seite **Datenbank** und **Tabelle** per Combobox. Als Quelle werden auch Sichten angeboten. Quelle und Ziel duerfen in unterschiedlichen Datenbanken derselben Instanz liegen; alle Objektnamen werden dreiteilig (`[DB].[Schema].[Tabelle]`) erzeugt.

3.3 Die drei Spalten

Bereich	Inhalt
1. Quellspalten	Alle Spalten der Quelle mit Datentyp und `[PK]`-Markierung. Angehakte Spalten werden verwendet. *Alle* / *Keine* fuer schnelles Umschalten.
2. Mapping Quelle -> Ziel	Pro Quellspalte die Zielspalte (automatisch nach Namen zugeordnet, Gross-/Kleinschreibung egal), beide Datentypen und die **Aenderungsart (SCD)**. `<nicht uebernehmen>` schliesst eine Spalte aus.
3. Schluesselspalten (Join)	Die Spalten fuer die `ON`-Bedingung. Vorbelegt mit dem Primary Key des Ziels. Ist der Ziel-PK nicht gemappt (typische SCD-2-Dimension mit Surrogatschluessel), wird der PK der Quelle genommen.

Berechnete Spalten, `rowversion`/`timestamp` und `GENERATED ALWAYS`-Spalten (System-Versioned Temporal Tables) werden nie als Ziel angeboten, weil sie nicht beschreibbar sind.

3.4 Optionen und Statement

Unter den drei Spalten liegen die Optionen (Kapitel 6), darunter das erzeugte Statement. Jede Aenderung an Auswahl, Mapping, Schluessel oder Option erzeugt das Statement sofort neu. Ist die Kombination ungueltig (z. B. kein Schluessel), steht im Statementfenster der Grund als Kommentar `-- Kein Statement: ...`.

3.5 Buttons

Button	Wirkung
In Zwischenablage	Statement kopieren, z. B. nach SSMS.
Speichern ...	Als `.sql`-Datei speichern.
Syntax pruefen	Kompiliert das Statement mit `SET NOEXEC ON` gegen den Server: Syntax, Tabellen- und Spaltennamen werden aufgeloest, **nichts wird ausgefuehrt**.
Ausfuehren ...	Fuehrt das Statement nach Rueckfrage aus und zeigt die Zeilenzahlen in der Statuszeile.

---

4. Modus Upsert / SCD Typ 1

Das Ziel wird zum Abbild der Quelle. Erzeugtes Statement (Beispiel mit *Nicht in Quelle: loeschen*, `HOLDLOCK` und `OUTPUT $action`):

```
SET XACT_ABORT ON;

MERGE INTO [DWH].[dbo].[Customer] WITH (HOLDLOCK) AS T
USING (
   SELECT [CustomerNo],
          [Name],
          [City]
   FROM [Staging].[dbo].[Customer]
) AS S
   ON T.[CustomerNo] = S.[CustomerNo]
WHEN MATCHED AND EXISTS (SELECT S.[Name], S.[City]
      EXCEPT
      SELECT T.[Name], T.[City]) THEN
   UPDATE SET
      T.[Name] = S.[Name],
      T.[City] = S.[City]
WHEN NOT MATCHED BY TARGET THEN
   INSERT ([CustomerNo], [Name], [City])
   VALUES (S.[CustomerNo], S.[Name], S.[City])
WHEN NOT MATCHED BY SOURCE THEN
   DELETE
OUTPUT $action AS MergeAction, COALESCE(inserted.[CustomerNo], deleted.[CustomerNo]) AS [CustomerNo]
;
```

Wichtige Details:

- `EXISTS (SELECT S... EXCEPT SELECT T...)` erkennt Aenderungen NULL-sicher. Ein Vergleich `S.Name <> T.Name` wuerde Wechsel von bzw. nach `NULL` uebersehen. Unveraenderte Zeilen werden nicht angefasst: kein unnoetiges Log, keine Trigger, kein geaenderter `rowversion`.
- Ein zweiter Lauf mit unveraenderten Daten aendert **0 Zeilen** (idempotent).
- Aenderungsart *Fest* bedeutet auch hier: Spalte wird nur beim `INSERT` gesetzt.
- Ist eine **Identity-Spalte** des Ziels gemappt, wird das Statement automatisch in `SET IDENTITY_INSERT ... ON/OFF` eingeschlossen und die Spalte aus `UPDATE SET` herausgenommen.

---

5. Modus SCD Typ 2

5.1 Voraussetzungen an die Zieltabelle

Eine Dimension mit Historie braucht mindestens **eine** dieser Verwaltungsspalten:

Spalte	Typischer Name	Inhalt
Aktuell-Flag	`IsCurrent`, `IstAktuell`	`1` fuer die aktuelle Version, `0` fuer historische
Gueltig bis	`ValidTo`, `GueltigBis`, `EndDate`	Ende der Version; offenes Ende `NULL` oder ein festes Datum
Gueltig ab (optional)	`ValidFrom`, `GueltigAb`, `StartDate`	Beginn der Version

Der GUI erkennt diese Spalten an verbreiteten deutschen und englischen Namensmustern, belegt die Comboboxen vor und schaltet **automatisch in den SCD-2-Modus**. Die Verwaltungsspalten duerfen nicht gleichzeitig gemappt sein.

Der **Schluessel** ist im SCD-2-Modus der fachliche Schluessel (Business Key, z. B. `CustomerNo`), nicht der Surrogatschluessel der Dimension. Der Surrogatschluessel (meist `IDENTITY`) wird von der Tabelle vergeben und darf nicht gemappt werden.

5.2 Aenderungsart pro Spalte

Aenderungsart	Wirkung im SCD-2-Modus
Typ 2 - Historie	Aenderung schliesst die aktuelle Version (`ValidTo = jetzt`, `IsCurrent = 0`) und fuegt eine neue Version ein.
Typ 1 - Ueberschreiben	Aenderung wird in **allen** Versionen des Schluessels ueberschrieben, es entsteht keine neue Version (z. B. Korrektur eines Tippfehlers im Namen).
Fest	Wird nie aktualisiert, nur beim ersten Einfuegen gesetzt.

5.3 Ablauf des erzeugten Statements

Alles laeuft in **einer Transaktion** mit `XACT_ABORT ON`:

1. Optional: **Duplikatpruefung** der Quelle, Abbruch mit `THROW 50001`.
2. `UPDATE` der **Typ-1-Spalten** in allen Versionen.
3. `MERGE` gegen die **aktuelle** Version (`AND T.IsCurrent = 1` bzw. `ValidTo` offen): neuer Schluessel wird als aktuelle Version eingefuegt, bei einer Typ-2-Aenderung wird die aktuelle Version geschlossen, optional (*Nicht in Quelle*) auch die Version fehlender Schluessel. Die betroffenen Zeilen gehen per `OUTPUT ... INTO #smgMergeOutput` in eine Temp-Tabelle.
4. `INSERT` der neuen Versionen fuer alle geschlossenen Schluessel, die noch in der Quelle sind.
5. Zusammenfassung als eine Ergebniszeile:

Spalte	Bedeutung
`NeueSchluessel`	Schluessel, die zum ersten Mal geladen wurden
`NeueVersionen`	Schluessel mit Typ-2-Aenderung (alte Version geschlossen, neue eingefuegt)
`GeschlosseneSchluessel`	In der Quelle fehlende Schluessel, deren Version geschlossen wurde
`Typ1Aktualisiert`	Zeilen, die durch Typ-1-Spalten ueberschrieben wurden

5.4 Warum kein Composable DML?

Das klassische Lehrbuchmuster `INSERT INTO dim SELECT ... FROM (MERGE ... OUTPUT ...) AS X` ist von SQL Server verboten, sobald die Zieltabelle **Trigger** hat oder von **Fremdschluesseln** (z. B. einer Faktentabelle) referenziert wird, und es meldet die eingefuegten Zeilen nicht sauber. sqmMergeGenerator nutzt deshalb `OUTPUT INTO #Temp` plus ein normales `INSERT`. Das funktioniert auch bei Dimensionen mit Triggern und FK-Beziehungen; beides ist im Live-Test abgedeckt.

5.5 Offenes Ende

Offenes Ende legt den Wert von `ValidTo` fuer die aktuelle Version fest: `NULL` (Standard) oder ein festes Datum wie `'9999-12-31'`. Eintraege sind frei editierbar; der Wert wird so wie eingegeben ins Statement uebernommen (Datumsliterale also in Hochkommas).

---

6. Optionen im Detail

Option	Upsert / SCD 1	SCD Typ 2	Standard
Nur bei Aenderung aktualisieren	`WHEN MATCHED AND EXISTS (... EXCEPT ...)`	immer aktiv	an
Nicht in Quelle	`WHEN NOT MATCHED BY SOURCE THEN DELETE`	aktuelle Version schliessen	aus
WITH (HOLDLOCK)	verhindert Race Conditions bei parallelen Laeufen	ebenso	an
OUTPUT $action	Aktion + Schluessel pro Zeile	Zusammenfassung statt Einzelzeilen	aus
Duplikatpruefung Quelle	`THROW 50001`, wenn die Quelle einen Schluessel doppelt liefert	ebenso	an
Quellfilter WHERE	`WHERE` in der `USING`-Unterabfrage	ebenso	leer

Hinweise:

- **Nicht in Quelle: loeschen** loescht alles im Ziel, was die Quelle nicht liefert. Bei einer gefilterten Quelle (*Quellfilter*) oder einer Teillieferung loeschen Sie damit auch Zeilen ausserhalb des Filters. Nur bei Vollabzuegen einschalten.
- **HOLDLOCK** sollte an bleiben. Ohne `HOLDLOCK` koennen zwei gleichzeitige `MERGE` denselben neuen Schluessel beide als *nicht vorhanden* sehen und doppelt einfuegen bzw. an einem Unique-Index scheitern.
- **Duplikatpruefung**: Ohne sie bricht der `MERGE` bei doppelten Schluesseln mit Fehler 8672 ab, oder im SCD-2-Modus entstehen falsche Versionen. Die Pruefung liefert eine klare Meldung, bevor irgendetwas geaendert wird.

Datentypen, die nicht vergleichbar sind

`xml`, `text`, `ntext`, `image` und CLR-Typen (`geography`, `geometry`, `hierarchyid`) koennen in `EXCEPT` nicht direkt verglichen werden. Fuer die Aenderungserkennung werden sie automatisch gecastet (`xml`/`ntext` nach `nvarchar(max)`, `text` nach `varchar(max)`, `image` und CLR-Typen nach `varbinary(max)`); zugewiesen wird der Originalwert.

---

7. Skripting ohne GUI

`New-sqmMergeStatement` ist ein reiner Textgenerator und braucht keine Datenbankverbindung. Geeignet fuer Build-Pipelines, Deployment-Skripte oder Agent-Jobs, die das Statement zur Laufzeit erzeugen.

Upsert

```
New-sqmMergeStatement -SourceDatabase Staging -SourceTable Customer `
   -TargetDatabase DWH -TargetTable Customer `
   -ColumnMapping @{ CustomerNo = 'CustomerNo'; Name = 'Name'; City = 'City' } `
   -KeyColumns CustomerNo -DeleteNotMatchedBySource -HoldLock
```

SCD Typ 2

```
$map = @(
   @{ SourceColumn = 'CustomerNo'; TargetColumn = 'CustomerNo' }
   @{ SourceColumn = 'Name';       TargetColumn = 'Name';    ChangeType = 'Type1' }
   @{ SourceColumn = 'City';       TargetColumn = 'City';    ChangeType = 'Type2' }
   @{ SourceColumn = 'Segment';    TargetColumn = 'Segment'; ChangeType = 'Fixed' }
)
New-sqmMergeStatement -SourceDatabase Staging -SourceTable Customer `
   -TargetDatabase DWH -TargetSchema dim -TargetTable Customer `
   -ColumnMapping $map -KeyColumns CustomerNo -Mode Scd2 -DeleteNotMatchedBySource `
   -HoldLock -DuplicateKeyCheck `
   -ValidFromColumn ValidFrom -ValidToColumn ValidTo -CurrentFlagColumn IsCurrent
```

Mapping aus den Metadaten ableiten

```
$cn   = @{ SqlInstance = 'SQL01' }
$cols = Get-sqmMergeColumn @cn -Database DWH -Schema dim -Table Customer |
       Where-Object { $_.IsWritable -and -not $_.IsIdentity -and
                      $_.ColumnName -notin 'ValidFrom','ValidTo','IsCurrent' }
$map  = $cols | ForEach-Object {
   @{ SourceColumn = $_.ColumnName; TargetColumn = $_.ColumnName
      TargetDataType = $_.DataType; TargetIsClrType = $_.IsClrType }
}
```

Die Angabe von `TargetDataType` und `TargetIsClrType` sorgt dafuer, dass `xml` und CLR-Spalten fuer den Vergleich korrekt gecastet werden. Ohne diese Angaben wird ohne Cast verglichen.

Statement ausfuehren

Das Modul fuehrt bewusst nichts aus. Zum Ausfuehren eignet sich z. B. `Invoke-Sqlcmd`, `Invoke-DbaQuery` oder ein Agent-Job-Step vom Typ T-SQL mit dem gespeicherten Statement.

---

8. Funktionsreferenz

Funktion	Zweck
`Show-sqmMergeGeneratorGui`	Startet die GUI. Parameter `-SqlInstance`, `-SqlCredential`; `-PassThru` baut das Formular nicht-modal auf (fuer automatisierte Tests).
`New-sqmMergeStatement`	Erzeugt das Statement aus Quelle, Ziel, Mapping und Schluessel. Keine Verbindung noetig.
`Get-sqmMergeDatabase`	Listet die Datenbanken einer Instanz (online, lesbar).
`Get-sqmMergeTable`	Listet Tabellen (und optional Sichten) einer Datenbank.
`Get-sqmMergeColumn`	Spalten mit Datentyp, PK-Position, Identity, berechnet, `rowversion`, `GENERATED ALWAYS`, CLR-Typ und `IsWritable`.

Parameter von New-sqmMergeStatement

Parameter	Bedeutung
`-SourceDatabase`, `-SourceSchema`, `-SourceTable`	Quelle (Schema Standard `dbo`)
`-TargetDatabase`, `-TargetSchema`, `-TargetTable`	Ziel (Schema Standard `dbo`)
`-ColumnMapping`	Liste von Hashtables/Objekten mit `SourceColumn`, `TargetColumn`, optional `ChangeType` (`Type1`, `Type2`, `Fixed`), `TargetDataType`, `TargetIsClrType`, `TargetIsIdentity`. Alternativ eine einzelne Hashtable `Quelle = Ziel`.
`-KeyColumns`	Join-Spalten (Ziel- oder Quellname). Im SCD-2-Modus der Business Key.
`-Mode`	`Upsert` (Standard) oder `Scd2`
`-UpdateOnlyChanged`	`$true` (Standard): Update nur bei echter Aenderung
`-DeleteNotMatchedBySource`	Upsert: loeschen. Scd2: aktuelle Version schliessen
`-HoldLock`, `-OutputAction`, `-DuplicateKeyCheck`	siehe Kapitel 6
`-IdentityInsert`	`IDENTITY_INSERT` erzwingen (wird bei gemappter Identity-Spalte automatisch gesetzt)
`-SourceFilter`	`WHERE`-Bedingung ohne das Wort `WHERE`
`-ValidFromColumn`, `-ValidToColumn`, `-CurrentFlagColumn`	SCD-2-Verwaltungsspalten
`-OpenEndValue`	Wert fuer das offene Ende, Standard `NULL`

---

9. Test und Freigabe

Vor dem ersten produktiven Einsatz eines erzeugten Statements:

1. **Syntax pruefen** in der GUI (oder das Statement in SSMS mit `SET NOEXEC ON` kompilieren).
2. In einer Testdatenbank ausfuehren und die Zusammenfassung bzw. `OUTPUT $action` kontrollieren.
3. **Zweiten Lauf** mit unveraenderten Quelldaten pruefen: es duerfen keine Zeilen geaendert werden (Upsert) bzw. `NeueVersionen = 0` (SCD 2).
4. Erst dann das gespeicherte `.sql` in Job oder Pipeline uebernehmen.

Mitgelieferter Live-Test

`Tests\Test-sqmMergeGenerator.ps1` legt zwei Wegwerf-Datenbanken (`smgTestSrc`, `smgTestTgt`) an, fuehrt die erzeugten Statements aus, prueft die Ergebnisse und loescht die Datenbanken wieder. Abgedeckt sind u. a. Upsert mit Delete, idempotenter zweiter Lauf, xml-Aenderungserkennung, Identity-Insert, SCD 2 mit Typ-1/Typ-2/Fest-Spalten, Schliessen fehlender Schluessel, zurueckkehrende Schluessel, Trigger und Fremdschluessel an der Dimension sowie die Duplikatpruefung (36 Pruefungen).

```
.\Tests\Test-sqmMergeGenerator.ps1 -SqlInstance SQL01
.\Tests\Test-sqmMergeGenerator.ps1 -SqlInstance SQL01 -SqlCredential (Get-Credential)
```

Nur auf einer Test-Instanz ausfuehren: der Test benoetigt `CREATE DATABASE`.

---

10. Troubleshooting und Einschraenkungen

Meldung / Verhalten	Ursache und Abhilfe
`-- Kein Statement: Mindestens eine Schluesselspalte ist erforderlich.`	In Bereich 3 keinen Schluessel angehakt.
`-- Kein Statement: Schluesselspalte '...' ist nicht im Mapping enthalten.`	Schluesselspalte in Bereich 1 abgewaehlt oder auf `<nicht uebernehmen>` gesetzt.
`Zielspalte(n) mehrfach gemappt`	Zwei Quellspalten zeigen auf dieselbe Zielspalte.
`SCD Typ 2 benoetigt eine Aktuell-Kennzeichen-Spalte und/oder eine Gueltig-bis-Spalte.`	In den Optionen *Gueltig bis* oder *Aktuell-Flag* waehlen.
`SCD-Verwaltungsspalte '...' darf nicht gleichzeitig gemappt sein.`	`ValidFrom`/`ValidTo`/`IsCurrent` im Mapping auf `<nicht uebernehmen>` setzen.
Fehler 50001 beim Ausfuehren	Die Quelle liefert einen Schluessel mehrfach. Quelle bereinigen oder Schluessel erweitern.
Fehler 8672	Wie 50001, aber ohne aktivierte Duplikatpruefung.
Identity-Fehler im SCD-2-Modus	Der Surrogatschluessel darf nicht gemappt sein.
Verbindung schlaegt mit Zertifikatsfehler fehl	*TrustServerCertificate* anhaken.

Bekannte Einschraenkungen:

• Quelle und Ziel muessen auf **derselben Instanz** liegen (dreiteilige Namen). Fuer instanzuebergreifende Abgleiche zuerst in eine Staging-Tabelle auf der Zielinstanz kopieren, z. B. mit sqmDataTransfer.

- `MERGE` feuert `AFTER`-Trigger einmal pro Aktionstyp, auch wenn 0 Zeilen betroffen sind. Trigger, die auf Aufrufanzahl reagieren, muessen das beruecksichtigen.
- Sehr grosse Quellen (zweistellige Millionen Zeilen) in einem einzigen `MERGE` belasten das Transaktionsprotokoll entsprechend. Bei Bedarf per *Quellfilter* in Teilmengen aufteilen (dann **ohne** *Nicht in Quelle*).
- Die GUI ist deutschsprachig, die erzeugten Kommentare im Statement ebenfalls.

---

11. Sicherheitshinweise

- Das Passwort einer SQL-Anmeldung wird nur im Speicher gehalten und nie in die Konfigurationsdatei geschrieben.
- Objekt- und Spaltennamen werden immer in eckige Klammern gesetzt und `]` korrekt maskiert.
- Der **Quellfilter** und das **Offene Ende** werden unveraendert ins Statement uebernommen. Sie sind fuer den Bediener gedacht, nicht fuer ungepruefte Eingaben Dritter.
- *Ausfuehren ...* verlangt immer eine Bestaetigung. Fuer produktive Ziele empfiehlt sich der Weg ueber *Speichern ...* und eine regulaere Freigabe.