



Update für Entwickler

# SQL Server 2012

Constantin „Kostja“ Klein  
Freudenberg Forschungsdienste SE & Co. KG  
[constantin.klein@justcommunity.de](mailto:constantin.klein@justcommunity.de)  
<http://kostjaklein.wordpress.com>  
<http://www.twitter.com/KostjaKlein>



# Neue Books Online

- Books Online sind jetzt Browser basiert.
- Konfigurierbar ob der Inhalt online oder offline angezeigt werden soll.
- Offline Inhalt lässt sich von einem Medium installieren oder online herunterladen.
- Wesentlich besser für Content Update.

# SEQUENCE

- Mit SEQUENCE erhalten wir einen zentralen Identitätsmechanismus (ähnlich zu ORACLE), der getrennt ist von Tabelle und Spalte.
- Ist von der Performanz besser als IDENTITY.
- Entspricht dem ANSI Standard.

```
CREATE SEQUENCE MySequence  
AS INT  
START WITH 1  
MINVALUE 1  
NO MAXVALUE;
```

# SEQUENCE

```
CREATE SEQUENCE [schema_name .] sequence_name
  [ AS [ built_in_integer_type | user-defined_integer_type ] ]
  [ START WITH <constant> ]
  [ INCREMENT BY <constant> ]
  [ { MINVALUE [ <constant> ] } | { NO MINVALUE } ]
  [ { MAXVALUE [ <constant> ] } | { NO MAXVALUE } ]
  [ CYCLE | { NO CYCLE } ]
  [ { CACHE [ <constant> ] } | { NO CACHE } ]
  [ ; ]
```

# Paging mit SQL Server 2000

```
CREATE PROCEDURE ReturnPage(@Select varchar(1000), @OrderBy varchar(1000),
    @StartRow int, @EndRow int)
AS
BEGIN

declare @ColList varchar(2000);
declare @Where varchar(2000);
declare @i int;
declare @i2 int;
declare @tmp varchar(1000);
declare @dec varchar(1000);
declare @f varchar(100);
declare @d varchar(100);
declare @Symbol char(2);
declare @SQL varchar(5000);
declare @Sort varchar(1000);

set @Sort = @OrderBy + ',';
set @dec = ""
set @Where = ""
set @SQL = ""

set @i = charindex(',', @Sort)
while @i != 0
begin
set @tmp = left(@Sort, @i-1)
set @i2 = charindex(',', @tmp)
set @f = ltrim(rtrim(left(@tmp, @i2-1)))
set @d = ltrim(rtrim(substring(@tmp, @i2+1, 100)))
set @Sort = rtrim(ltrim(substring(@Sort, @i+1, 100)))
set @i = charindex(',', @Sort)
set @symbol = case when @d = 'ASC' then '>' else '<' end +
    case when @i=0 then '=' else "" end

set @dec = @dec + 'declare @' + @f + ' sql_variant;'
set @ColList = isnull(replace(replace(@colList, '>','='), '<','=') + ' and ', '') +
    @f + @Symbol + '@' + @f
set @Where = @Where + ' OR (' + @ColList + ')'
set @SQL = @SQL + '@' + @f + '=' + @f
end

set @SQL = @dec + '' +
'SET ROWCOUNT ' + convert(varchar(10), @StartRow) + ';' +
'SELECT ' + substring(@SQL, 3, 7000) + ' from (' + @Select + ') a ORDER BY ' +
@OrderBy + ';' + 'SET ROWCOUNT ' +
convert(varchar(10), 1 + @EndRow - @StartRow) + ';' +
'select * from (' + @Select + ') a WHERE ' +
substring(@Where, 4, 7000) + ' ORDER BY ' + @OrderBy + '; SET ROWCOUNT 0;'

exec(@SQL)
END
```

# Paging seit SQL Server 2005

```
WITH ProductPaging AS  
(  
  SELECT Row_Number() OVER (ORDER BY ProductName) AS  
  Number, ProductID, ProductName FROM Products  
)  
  
SELECT * FROM ProductPaging  
WHERE Number BETWEEN 5 AND 9
```

# Paging mit SQL Server 2012

- Ähnlich zu LIMIT in MySQL, aber ANSI Standard

```
SELECT ...  
ORDER BY ...  
OFFSET <expr> ROWS  
FETCH NEXT <expr> ROWS ONLY;
```

# THROW

- Soll bestehende Probleme mit RAISERROR adressieren.
  1. RAISERROR('Error', 10, 1);  
PRINT 'Hello';
  2. RAISERROR('Error', 16, 1);  
PRINT 'Hello';
  3. RAISERROR('Error', 20, 1);  
PRINT 'Hello';
  4. RAISERROR('Error', 20, 1) WITH LOG;  
PRINT 'Hello';
- Keine einfache Vorhersagbarkeit
- Unterschiedliches Abbruchverhalten



# THROW

- THROW schmeißt immer einen Fehler.
- Beendet den Batch wenn nicht SET XACT\_ABORT ON gesetzt ist. In diesem Fall beendet es die Transaktion.
- Kann ohne oder in Kombination mit sys.messages verwendet werden.

```
THROW <number>, 'Message', <state>
```

```
THROW 51000, 'The record does not exist.', 1
```

# RAISERROR vs. THROW

| RAISERROR-Anweisung   | THROW-Anweisung   |
|---|---|
| Wenn eine <code>msg_id</code> an RAISERROR übergeben wird, muss die ID in <code>sys.messages</code> definiert werden. | Der <code>error_number</code> -Parameter muss nicht in <code>sys.messages</code> definiert werden.    |
| Der <code>msg_str</code> -Parameter kann printf-Formatvorlagen enthalten.   | Der <code>message</code> -Parameter akzeptiert keine printf-Formatierung.                             |
| Der <code>severity</code> -Parameter gibt den Schweregrad der Ausnahme an.  | Es gibt keinen <code>severity</code> -Parameter. Der Ausnahmeschweregrad ist immer auf 16 festgelegt. |

# Neue Funktionen

---

# Konvertierungsfunktionen

- PARSE
- TRY\_CONVERT
- TRY\_PARSE

# Datums- und Zeitfunktionen

- DATEFROMPARTS
- DATETIME2FROMPARTS
- DATETIMEFROMPARTS
- DATETIMEOFFSETFROMPARTS
- EOMONTH
- SMALLDATETIMEFROMPARTS
- TIMEFROMPARTS

# Logische Funktionen

- CHOOSE
- IIF - syntactic sugar um CASE

# String Funktionen

---

- CONCAT
- FORMAT

# Window functions

---



# Was ist eine window function?

- „The OVER clause defines a window or user-specified set of rows within a query result set. A window function then computes a value for each row in the window.“

--SQL Server 2012, BOL

# Enhanced OVER

- Die OVER Syntax wurde erweitert um window functions besser zu unterstützen

```
OVER (  
    [ <PARTITION BY clause> ]  
    [ <ORDER BY clause> ]  
    [ <ROW or RANGE clause> ]  
)
```

# Analytische Funktionen

- CUME\_DIST
- FIRST\_VALUE
- LAG
- LAST\_VALUE
- LEAD
- PERCENTILE\_CONT
- PERCENTILE\_DISC
- PERCENT\_RANK

# File Handling

---

# (BIG) Data

---

**1.000.000.000.000.000.000.000**  
**= 1 Zettabyte**

# Recap - Filestream

- Attribut für varbinary(max)
- Dateien werden direkt im Dateisystem gespeichert
- Duales Programmiermodell

# Was ist neu mit SQL Server 2012

- Filestream ist jetzt ausgelegt für > 100 Millionen Dokumente
- FILETABLE (Filestream für alle 😊)
- Volltext Verbesserungen
- Statistische semantische Suche

# FILETABLE

- Ist eine Tabelle mit Dateien und Ordnern:
  - Eine Zeile pro Datei oder Ordner
  - Festes Schema
- Die Datei und Ordner Hierarchie ist nach außen über einen Windows Share sichtbar
  - Das heißt, die Speicherung in der Datenbank ist transparent für Windows Anwendungen



# Volltext Verbesserungen

- Unterstützung für neue Sprachen:
  - Czechisch, Dänisch, Griechisch, Polnisch, Türkisch
- Update für word breakers und stemmers
- Property Suche
  - Suche in Dateieigenschaften, wie z.B. Autor und Titel

# Customizable NEAR

- Relevanz ist häufig abhängig von der Nähe der Suchworte
- SQL Server 2008 hatten einen festen NEAR Operator
- NEAR in SQL 2012 erlaubt die Angabe der:
  - Max. Anzahl Non-Search Terms zwischen Treffern
  - Reihenfolge von Suchworten

# Statistische semantische Suche

Erlaubt

- Bedeutung und nicht Worte zu suchen
- tiefe Einblicke in Dokumente

Durch

- Automatisierte Tag-Extraktion
- Das Finden von ähnlichem Content

# SQL Server Data Tools (SSDT) und LocalDB

---

Fragen ???

# Wrap up

- SQL Server 2012 ist ein echtes Major Release mit Neuerungen in fast allen Bereichen.
- Auch für (Datenbank-)Entwickler ist wieder viel Neues dabei.
- Ich hoffe Ihr habt jetzt einen guten Einstieg und einen ersten Überblick.

# SQLServerBlogs.de

- Informationsquelle für deutschsprachige SQL Server Inhalte

SSB <http://SQLServerBlogs.de>

Twitter:

<http://www.twitter.com/SQLServerBlogs>

# Weitere Termine – Community

- Weiteres Community Event im September:
  - SQL Saturday 230 in St. Augustin am 13.07.2013
    - <http://sqlsaturday.com>





**Feedback !!!**

# Speakerinfo: Constantin Klein



Constantin arbeitet als Anwendungsarchitekt und Entwickler bei der Freudenberg Forschungsdienste SE & Co. KG. Dort beschäftigt er sich hauptsächlich mit dem Design und der Entwicklung von Web-Informationssystemen und Datenbanken. Seit seinem Studium der Wirtschaftsinformatik gilt sein besonderes Interesse darüber hinaus allen aktuellen Themen im Microsoft .NET Umfeld, insbesondere aber dem Thema Softwarearchitektur. Er ist MCSD, MCITP Database Developer und MCPD Web + Enterprise Application Developer. 2010, 2011 und 2012 wurde er von Microsoft zum Most Valuable Professional (MVP) für SQL Server ernannt. Er engagiert sich zusätzlich im Vorstand des [Just Community e.V.](#) und als Leiter der [.NET User Group Frankfurt](#).

Sein Blog finden Sie unter

<http://kostjaklein.wordpress.com>

# Ressourcen

- Programmability Enhancements (Database Engine) in BOL  
[http://msdn.microsoft.com/en-us/library/cc645577\(v=SQL.110\).aspx](http://msdn.microsoft.com/en-us/library/cc645577(v=SQL.110).aspx)
- SQL Server 2012 Developer Training Kit  
<http://www.microsoft.com/download/en/details.aspx?id=27721>