

SQL Server 2016

Neuerungen für Programmierer



Thorsten Kansy (tkansy@dotnetconsulting.eu)

Meine Person- Thorsten Kansy

Freier Consultant, Software Architekt,
Entwickler, Trainer & Fachautor



Azure Cosmos DB








Mein Service- Ihr Benefit

- Individuelle Inhouse Trainings
- (Online on-demand) Projektbegleitung
- Beratung
 - Problemanalyse und Lösungen
 - Technologieentscheidungen



Agenda I

- Dynamic Data Masking 
- Row Level Security 
- Always Encrypted 
- JSON Support 
- Temporal Table 



 Auch als Kurzvideo verfügbar

Agenda II

- SESSION_CONTEXT 
- DROP IF EXISTS
- Live Statistics 
- Multiple TempDB Database Files
- Stretch Database

 Auch als Kurzvideo verfügbar

Agenda III

- Query Store 
- In-Memory Verbesserungen
- Clustered Columnstore Index 
- R goes SQL Server
- Polybase

 Auch als Kurzvideo verfügbar



Row Level Security

Zeilenweiser, funktionsbasierter Filter

```
CREATE SECURITY POLICY SecurityFilter
  ADD FILTER PREDICATE dbo.fn_AccessFilter([SecurityDescriptor]) ON
    [dbo].[Mitarbeiter],
  ADD BLOCK PREDICATE dbo.fn_AccessFilter([SecurityDescriptor]) ON
    [dbo].[Mitarbeiter] AFTER INSERT,
  ADD BLOCK PREDICATE dbo.fn_AccessFilter([SecurityDescriptor]) ON
    [dbo].[Mitarbeiter] BEFORE DELETE,
  ADD BLOCK PREDICATE dbo.fn_AccessFilter([SecurityDescriptor]) ON
    [dbo].[Mitarbeiter] BEFORE UPDATE;
```




Dynamic Data Masking

Automatisch Anonymisierung

```
SELECT * FROM dbo.Mitarbeiter;
```



UNMASK?



Results		Messages			
	ID	Name	Gehalt	Telefon	EMail
1	1	T-sy	695.54	xxxx	tXXX@XXXX.com
2	2	J-nd	863.30	xxxx	jXXX@XXXX.com
3	3	D-er	420.50	xxxx	dXXX@XXXX.com

	ID	Name	Gehalt	Telefon	EMail
1	1	Thorsten Kansy	999.90	1234567	tkansy@dotnetconsulting.eu
2	2	James Bond	12000.00	234567	jbond@dotnetconsulting.eu
3	3	Doc Snyder	9.99	3456789	dsnyder@dotnetconsulting...

Filter, Sortierungen, etc. funktionieren!



Always Encrypted

Verschlüsselung von Daten

- Auf den Massenspeichern des Servers
- Im Transit (Netzwerk Server ↔ Client)

- .NET 4.6

```
Data Source=.; Initial Catalog=dotnetconsulting;  
Integrated Security=true; Column Encryption  
Setting=enabled;
```



JSON Support

Integrierter JSON Support

```
SELECT TOP 3
    name, database_id, source_database_id
FROM sys.databases
ORDER BY database_id
FOR JSON PATH, ROOT('Databases'), INCLUDE_NULL_VALUES;
```

- FOR JSON-Klausel
- ISJSON-Funktion
- JSON_VALUE-Funktion
- OPENJSON-Funktion
- JSON_QUERY-Funktion
- JSON_MODIFY-Funktion

JSON Path als Abfragesprache (das XPATH des JSON)



Temporal Table

Automatisch Historisierung von Daten

```
CREATE TABLE dbo.Werte
(
    ID INT IDENTITY(1,1) NOT NULL,
    Wert1 NVARCHAR(10) NULL,
    Wert2 NVARCHAR(10) NULL,
    StartTime datetime2(7) GENERATED ALWAYS AS ROW START HIDDEN NOT NULL,
    EndTime datetime2(7) GENERATED ALWAYS AS ROW END HIDDEN NOT NULL,
    PERIOD FOR SYSTEM_TIME(StartTime, EndTime),
    CONSTRAINT Werte_PK PRIMARY KEY (ID)
)
WITH
(SYSTEM_VERSIONING = ON (HISTORY_TABLE = dbo.Werte_History));

SELECT * FROM dbo.Werte
[FOR SYSTEM_TIME AS OF '2016-01-25 18:21:24.0738473'];
```



SESSION_CONTEXT

Verbindungsweites Dictionary

```
EXEC sp_set_session_context  
    @key = 'Department',  
    @value='Sales';
```

```
SELECT SESSION_CONTEXT(N'Department');
```

„Schalter“ für Trigger, Views, etc.

STRING_SPLIT()

Zeichenkette mit Separator aufteilen

```
SELECT [Value]
      FROM STRING_SPLIT('1;2;3',';');
```

Gegenstück (STRING_AGG()
ab SQL Server 2017)

DROP IF EXISTS

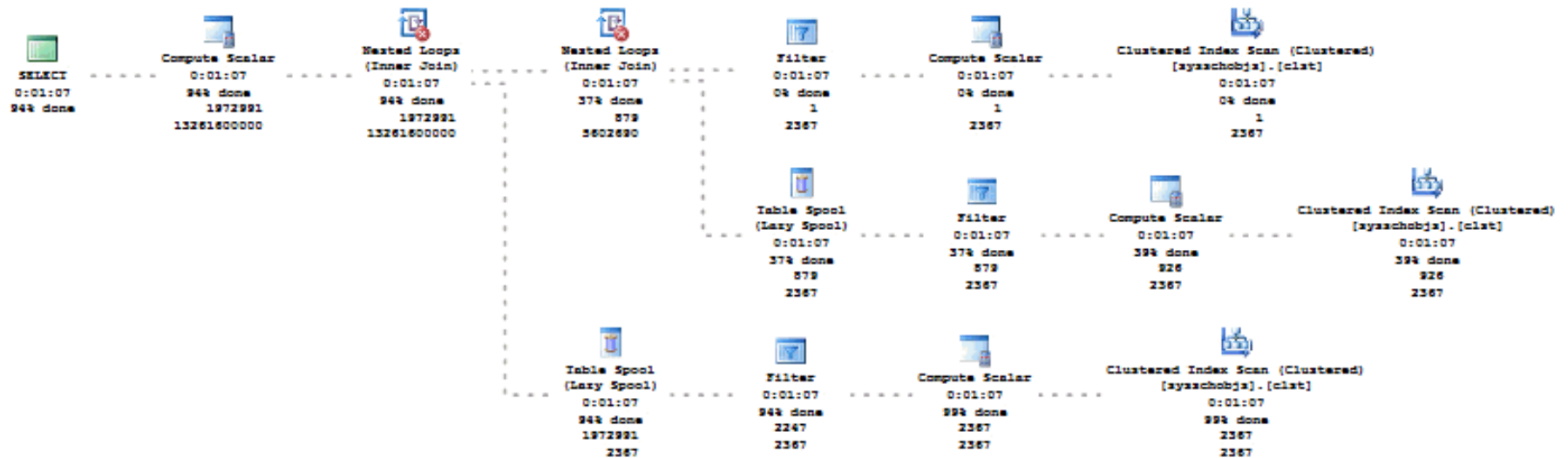
Sorgloses Droppen von Objekten

```
DROP PROCEDURE IF EXISTS dbo.uspProc1;  
DROP TABLE IF EXISTS dbo.table1;  
DROP VIEW IF EXISTS dbo.view1;  
DROP ... IF EXISTS ...;
```



Live Statistics

Echtzeit Auswertung von Abfragen



Multiple TempDB Files

TempDB Setup bei Installation

SQL Server 2016 CTP3.3 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators, data directories and TempDB settings.

Product Key
License Terms
Global Rules
Microsoft Update
Install Setup Files
Install Rules
Setup Role
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Reporting Services Configuration
Feature Configuration Rules
Ready to Install
Installation Progress
Complete

Server Configuration | Data Directories | **TempDB** | FILESTREAM

TempDB data files: tempdb.mdf, tempdb_mssql_#.ndf

Number of files: 4
Initial size (MB): 8
Autogrowth (MB): 64

Total initial size (MB): 32
Total autogrowth (MB): 256

Data directories: C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER

Add...
Remove

TempDB log file: templog.ldf

Initial size (MB): 8
Autogrowth (MB): 64

Log directory: C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER

< Back | Next > | Cancel | Help

Stretch Database

Historische Daten in Azure SQL
Datenbank auslagern

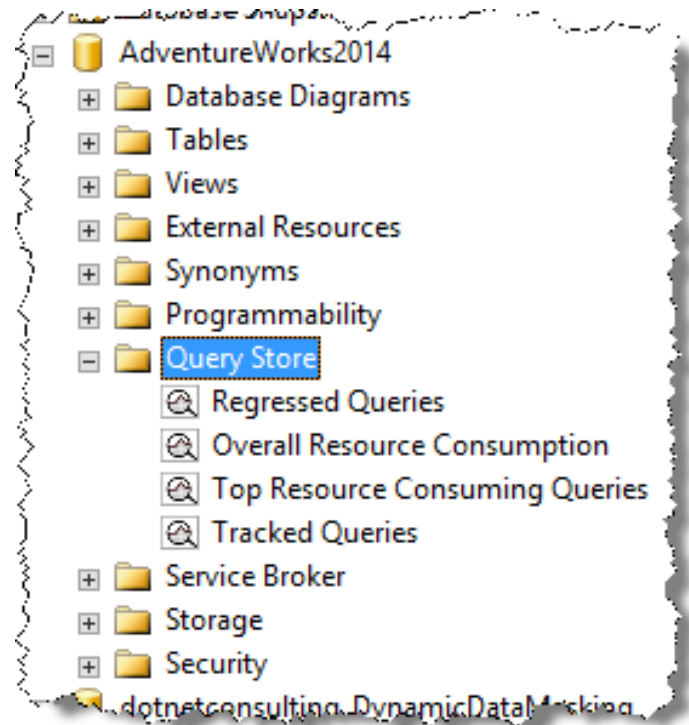
- Warm Storage (kein Cold Storage)
=> stehen für Abfragen bereit

On premise => Clustered Columnstore Index



Query Store

Einfache Analyse von Ausführungspläne
& Performancedaten



```
ALTER DATABASE {DbName} SET QUERY_STORE = ON;
```

In-Memory Verbesserungen

- ALTER TABLE
- Persistierte Memory Optimized Tables > 2 TB
- LOBs (`varbinary(max)`,
`[n]varchar(max)`) werden unterstützt
- DML Trigger
- Fremdschlüssel
- Check/Unique Constraints
- Transparent Data Encryption (TDE)



Clustered Columnstore Index

- Secondary Indexes
- Tuple Mover
 - Merged nun Thread übergreifen

R goes SQL Server

Statistische Analyse Sprache

```
countdown <- function(from)
{
  print(from)
  while(from!=0)
  {
    Sys.sleep(1)
    from <- from - 1
    print(from)
  }
}
countdown(5)
```

Polybase

Zugriff auf semistrukturierte Daten

- Hadoop
- Azure Blob Storage

Fragen? Jetzt oder später!

Kontakt

E-Mail

tkansy@dotnetconsulting.eu

Telefon

+49 (0) 6187 / 2009090

Microsoft Teams

[Meet now](#)

LinkedIn

[Link me](#)

XING

[Xing me](#)

X (Twitter)

[@tkansy](#)



www.dotnetconsulting.eu

SQL Server meets .NET (Core)- professionally!



Ich berate, coache und trainiere im Bereich Entwicklung von .NET (Core) Anwendungen mit Microsoft SQL Server- mit Allem, was dazu gehört- und was man vielleicht weglassen sollte.