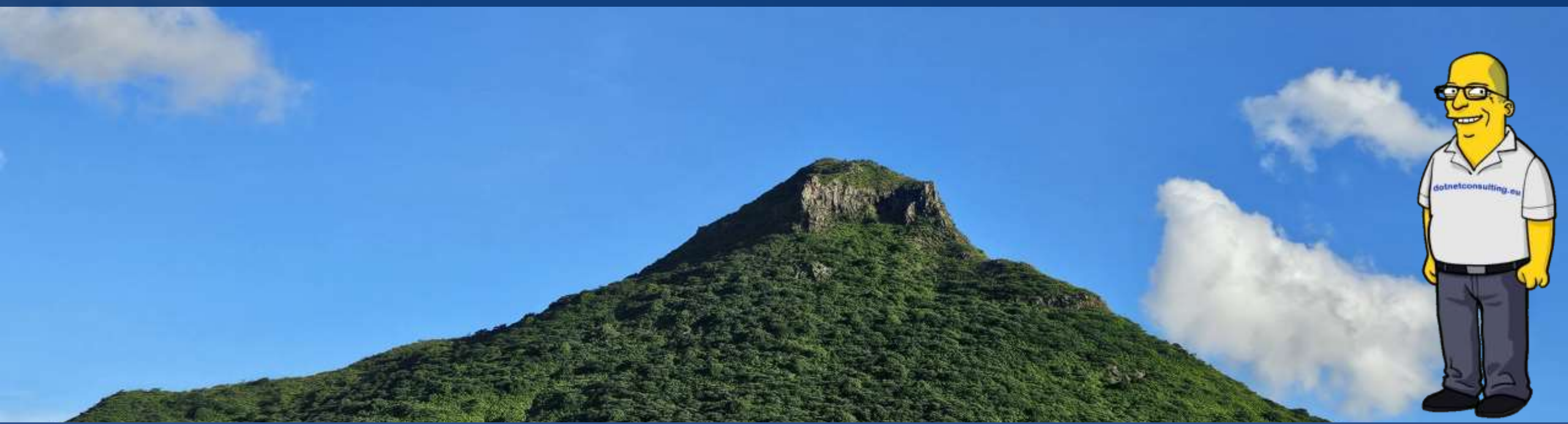


Aus der Praxis EF Core 8.0

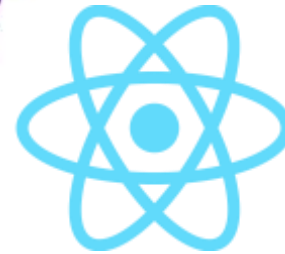
(RDBMS, SQL Server)



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



Agenda

- DbContext
 - Datenbankentwurf
 - Was muss in einem Modell stehen?
 - Temporal Tables
 - Löschen ohne Laden
 - Bulk Updates/ Bulk Deletes
 - AsNoTracking
 - Asynchrones
 - Global Query Filter
 - Datenbankprojekte
-
- EF Power Tools



DbContext



DbContext

Lebenszeit richtig einsetzen!

- ASP.NET Core
 - Per Request
- Desktop-App
 - Nicht pro Anwendungsart
 - Per Dialog?

A close-up photograph of a tortoise's head, showing its eye, nostril, and the texture of its skin and shell. The tortoise is facing right. A semi-transparent blue horizontal band is overlaid across the middle of the image, containing the text 'Datenbankentwurf'.

Datenbankentwurf

Datenbankentwurf

- Sinnvolle Normalisierung
 - Einen PK (numerisch, "Id")
 - Die richtigen Datentypen
 - Datum, Decimal, etc.
 - Sinnvolle Spaltengrößen
 - Relationen, Fremdschlüssel
-
- Je besser die DB aufgebaut ist, desto einfacher ist der Einsatz des EF Core



Was muss in einem Modell stehen?

Was muss in einem Modell stehen?

- Als erstes das, was die Welt macht, damit sie Euch gefällt.
 - sprechende Namen
 - passende Datentypen
 - ...
- Die Tabellen/ Sichten/ Prozeduren/ etc. die wirklich benötigt werden.
- Konventionen
- Ohne Migration:
 - Keine Indizes, DB-Datentypen, etc.



Temporal Tables



Temporal Tables- Automatische Historisierung

```
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'];
```

IsTemporal (Fluent API)

- Temporal table (SQL Server)
- Standard: -

```
modelBuilder.Entity<MyTemporalTable>()  
    .ToTable(t => t.IsTemporal(hist =>  
        {  
            hist.UseHistoryTable("SessionHistory");  
        }  
    );
```

```
dbContext.MyTemporalTable  
    .TemporalAsOf(<Date>).Where(w => ...);
```


Temporal AsOf & Co

TemporalAll()	Alle Daten aus der Tabelle und der historischen Tabelle.
TemporalAsOf(...)	Daten zu einem bestimmten Zeitpunkt.
TemporalFromTo(...)	Daten aus der Tabelle plus der historischen Tabelle exkl. Einschluss der oberen Grenze.
TemporalBetween(...)	Daten aus der Tabelle plus der historischen Tabelle inkl. Einschluss der oberen Grenze.
TemporalContainedIn(...)	Nur Daten aus der historischen Tabelle eines definierten Zeitraums.

A wide-angle photograph of a coastal area. On the left, a green hillside is topped with several buildings, including a prominent one with a golden roof. Below the hill is a paved promenade with people walking. In the foreground, a concrete walkway runs along a sandy beach. To the right of the beach is a rocky shoreline with large, light-colored rocks. Several people are visible near the water's edge, some appearing to be fishing. The ocean is a deep blue, and the sky is clear and bright blue.

Löschen ohne Laden

Löschen ohne Laden

- Der Zusatz einer Entity bestimmt das Statement, das EF Core generiert!
- Manchmal macht es Sinn, ohne Laden aus der DB zu löschen.
- Achtung! Entität ***muss*** existieren.

(Bekannter) Zustand einer Entität

Der Zustand einer Entity bestimmt das Statement, das EF Core generiert!

```
_efContext.Entry(s).State = EntityState.Added;
```

Wert	
Detached (*)	Entität dem Kontext unbekannt
Unchanged (*)	Keine Änderung, keine Aktion für die Datenbank
Deleted (*)	Entität gelöscht => DELETE-Anweisung
Modified (*)	Entität verändert => UPD
Added (*)	Entität (neu) hinzugefügt

(*) Microsoft.EntityFrameworkCore

```
Speaker speaker2 = new() { Id = 34 };  
_efContext.Entry(speaker2).State = EntityState.Deleted;  
  
// Speichern  
_efContext.SaveChanges();
```



Demo





Bulk Updates/ Bulk Deletes

Bulk Updates

Updates und Deletes direkt auf der Datenbank

```
// Alle Datensätze ändern
int rowsAffected = _bulkUpdatesDbContext.Speakers
    .Where(w => w.Id <= 500)
    .ExecuteUpdate(w => w.SetProperty(p => p.Name, p => "Sprecher"));

int rowsAffected = _bulkUpdatesDbContext.Speakers
    .Where(w => w.Id <= 500)
    .ExecuteDelete();
```



Demo





AsNoTracking

AsNoTracking()

- Verbesserte Leistung gegenüber regulären LINQ-Abfragen.
- Vollständig materialisierte Objekte.
- Ergebnis wird in den Speicher gestreamt- besonders effizient bei größeren Datenmengen.
- Umgeht den EF Core Cache.
- Umgeht die „unique key per record“-Regel.

QueryTrackingBehavior

Bestimmt, ob die Entität vom Kontext überwacht werden soll

```
// Change Tracker - Alles tracken außer...
_efContext.ChangeTracker.QueryTrackingBehavior = QueryTrackingBehavior.TrackAll;
var queryNoTrack = from q in _efContext.Speakers.AsNoTracking()
                   where q.Name.StartsWith("Speaker")
                   select q;

// Change Tracker - Nichts tracken außer...
_efContext.ChangeTracker.QueryTrackingBehavior = QueryTrackingBehavior.NoTracking;
var queryTrack = from q in _efContext.Speakers.AsTracking()
                  where q.Name.StartsWith("Speaker")
                  select q;
```

A close-up photograph of a tortoise's head, showing its eye, nostril, and the texture of its skin and shell. A semi-transparent blue horizontal band is overlaid across the middle of the image, containing the word "Asynchrone" in white text. The background is blurred, showing more of the tortoise and some outdoor ground.

Asynchrone

Asynchrones

- Wenn möglich, immer asynchrone Methoden bevorzugen.
- App ist nicht automatisch schneller, verbraucht aber weniger Ressourcen.
- Damit mehr Aktionen mit gleichen Ressourcen durchgeführt werden können.
- Ein Task wartet nicht auf das Ende einer Operation, sondern macht an anderer Stelle weiter.



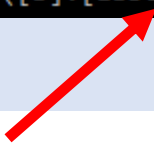
Global Query Filter

HasQueryFilter (Fluent API)

- Globale Einschränkung für alle Abfragen
 - Kann ignoriert werden (`IgnoreQueryFilters()`)
- Standard: –

```
modelBuilder.Entity<Session>()  
    .HasQueryFilter(p => p.IsDeleted == false);
```

```
SELECT TOP(1) [s].[Id], [s].[ContentDescription], [s].[Begin], [s].[Created], [s].[Difficulty], [s].[Duration], [s]  
].[End], [s].[EventId], [s].[IsDeleted], [s].[SpeakerId], [s].[TechEventId], [s].[Title], [s].[Updated]  
FROM [dnc].[Sessions] AS [s]  
WHERE ([s].[IsDeleted] = 0) AND ([s].[Id] = @__get_Item_0)
```





Demo



Datenbankprojekte



Datenbankprojekte (SQL Server)

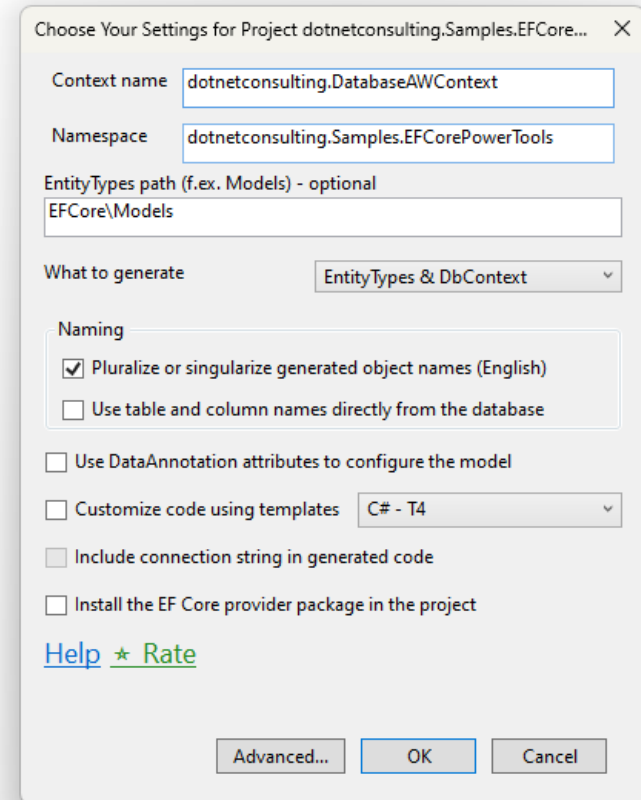
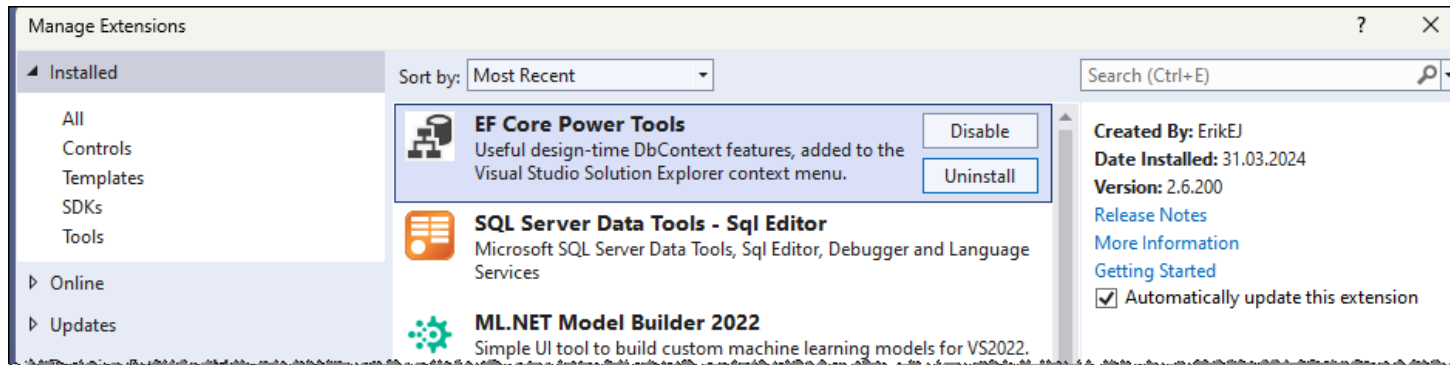
- “Konserviert” die Datenbankstruktur in Solution
 - Eine DB ist mehr als EF Core abdeckt
- Alle Features der Datenbank werden unterstützt
- Inkonsistenzen aufgedeckt
- Refactoring
- Quellcodeverwaltung
- DACPAC

The image shows the interior of a large, modern building with a complex, organic structure. The space is filled with tall, slender columns and beams, some of which are illuminated with warm orange and yellow lights. The ceiling is a series of white, star-shaped or floral patterns, and the walls are also decorated with colorful, abstract designs. The overall atmosphere is one of a futuristic, high-tech environment.

EF Core Power Tools

EF Core Power Tools

UI Basierte Reverse Engineering



A person is silhouetted against a vibrant sunset over the ocean. The person is standing in shallow water, holding a glowing light above their head. The sky is a mix of orange, red, and blue, and the water reflects the colors. A dark rock formation is visible on the horizon to the left.

Sonstiges

Sonstiges

- An Konventionen halten
- Context & Modelle sollten nur das enthalten was wirklich benötigt wird
- Lebenszeit des Contexts berücksichtigen

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.