

C# und .NET

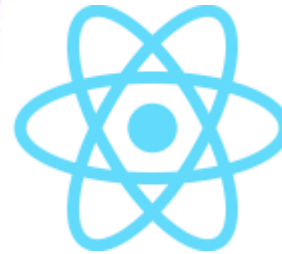
Was bringen die ganzen neuen Funktionen und wer braucht sie?



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



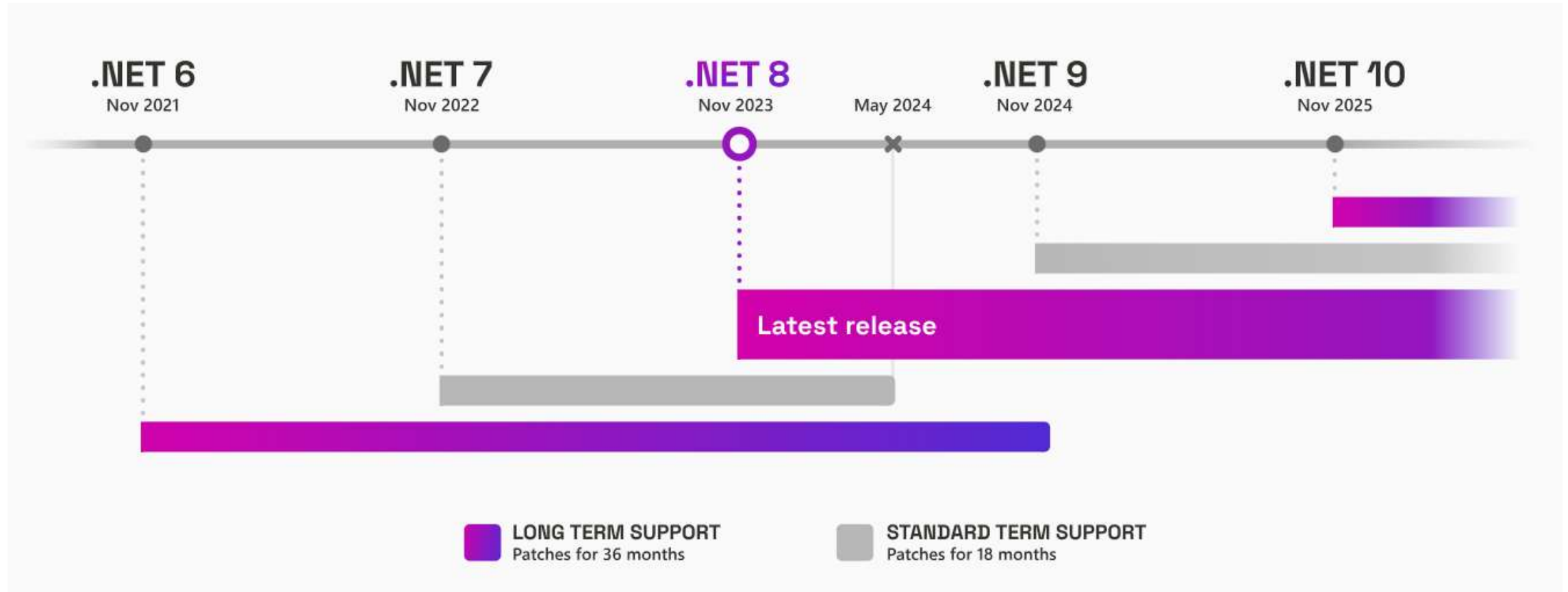
Alle Jahre wieder



Eine neue .NET Version
Eine neue C# Version

Aber auch :
Eine weitere alte .NET Version

Der .NET (Core) Roadmap



<https://dotnet.microsoft.com/en-us/platform/support/policy/dotnet-core>

Es ist eigentlich noch schlimmer



[Download .NET](#)

Was bedeutet “Kein Support”?

```
Windows PowerShell
```

```
.NET Runtimes:
```

Name	Version	Status
Microsoft.AspNetCore.App	3.1.32	.NET 3.1 is out of support.
Microsoft.NETCore.App	3.1.32	.NET 3.1 is out of support.
Microsoft.WindowsDesktop.App	3.1.32	.NET 3.1 is out of support.
Microsoft.AspNetCore.App	6.0.25	Patch 6.0.26 is available.
Microsoft.NETCore.App	6.0.25	Patch 6.0.26 is available.
Microsoft.WindowsDesktop.App	6.0.25	Patch 6.0.26 is available.
Microsoft.NETCore.App	6.0.26	Up to date.
Microsoft.WindowsDesktop.App	6.0.26	Up to date.
Microsoft.AspNetCore.App	7.0.7	.NET 7.0 is going out of support soon.
Microsoft.NETCore.App	7.0.7	.NET 7.0 is going out of support soon.
Microsoft.WindowsDesktop.App	7.0.7	.NET 7.0 is going out of support soon.
Microsoft.AspNetCore.App	7.0.15	.NET 7.0 is going out of support soon.
Microsoft.NETCore.App	7.0.15	.NET 7.0 is going out of support soon.
Microsoft.WindowsDesktop.App	7.0.15	.NET 7.0 is going out of support soon.
Microsoft.AspNetCore.App	8.0.0	Patch 8.0.1 is available.
Microsoft.NETCore.App	8.0.0	Patch 8.0.1 is available.
Microsoft.WindowsDesktop.App	8.0.0	Patch 8.0.1 is available.
Microsoft.AspNetCore.App	8.0.1	Up to date.
Microsoft.NETCore.App	8.0.1	Up to date.
Microsoft.WindowsDesktop.App	8.0.1	Up to date.

Software funktioniert noch!

Keine Bug fixes

Keine neuen Features

Keine Security Patches

Keine Garantie der
Lauffähigkeit aus neuen OS-
Versionen

Ein paar Beispiele gefällig?



Top level statements
Global usings
Record classes/structs
Primary constructors
...

Top level statements (C# 9)

Mehrwert?

```
Program.cs x
dotnetconsulting.TimeTracking.GuiMVC
11 using dotnetconsulting.TimeTracking.AppLogic;
12 using dotnetconsulting.TimeTracking.EFCosmosSqlRepository.EntityFramework;
13 using dotnetconsulting.TimeTracking.EFRepository.EntityFramework;
14 using dotnetconsulting.TimeTracking.GuiMVC.Code;
15 using dotnetconsulting.TimeTracking.GuiMVC.Data;
16 using dotnetconsulting.TimeTracking.Interfaces;
17 using Microsoft.AspNetCore.Identity;
18 using Microsoft.EntityFrameworkCore;
19 using efRepo = dotnetconsulting.TimeTracking.EFRepository;
20 using efRepoCosmosSql = dotnetconsulting.TimeTracking.EFCosmosSqlRepository;
21
22 var builder = WebApplication.CreateBuilder(args);
23
24 builder.Services.Configure<CookiePolicyOptions>(options =>
25 {
26     // This lambda determines whether user consent for non-essential cookies is needed for a given request.
27     options.CheckConsentNeeded = context => true;
28     options.MinimumSameSitePolicy = SameSiteMode.None;
29 });
30
31 builder.Services.AddDbContext<ApplicationDbContext>(options =>
32     options.UseSqlServer(builder.Configuration.GetConnectionString("DefaultConnection")!));
33 builder.Services.AddDefaultIdentity<IdentityUser>()
34     .AddEntityFrameworkStores<ApplicationDbContext>();
35
36 builder.Services.AddMvc(o =>
37 {
```

Mehrwert?

Global Usings/ Implicit Namespace Imports

Global Namespace Importe

```
<PropertyGroup>  
  <OutputType>Exe</OutputType>  
  <TargetFramework>netcoreapp6.0</TargetFramework>  
  <!-- Standard für .NET 6.0 -->  
  <LangVersion>preview</LangVersion>  
  <ImplicitUsings>>true</ImplicitUsings>  
</PropertyGroup>  
  
<ItemGroup>  
  <Using Include="System.Web" />  
  <Using Remove="System.Threading.Tasks" />  
</ItemGroup>
```

Alternative im Code

```
10  
11 global using System.Text.Json;
```

Console/Library

- System
- System.Collections.Generic
- System.IO
- System.Linq
- System.Net.Http
- System.Threading
- System.Threading.Tasks

Web

In addition to the console/library namespaces :

- System.Net.Http.Json
- Microsoft.AspNetCore.Builder
- Microsoft.AspNetCore.Hosting
- Microsoft.AspNetCore.Http
- Microsoft.AspNetCore.Routing
- Microsoft.Extensions.Configuration
- Microsoft.Extensions.DependencyInjection
- Microsoft.Extensions.Hosting
- Microsoft.Extensions.Logging

Worker

In addition to the console/library namespaces :

- Microsoft.Extensions.Configuration
- Microsoft.Extensions.DependencyInjection
- Microsoft.Extensions.Hosting
- Microsoft.Extensions.Logging

Mehrwert?

Records class (C# 9)/ structs (C# 10)

```
// C# 9.0
```

```
0 references | - changes | -authors, -changes
```

```
record class RecordClassA(string Name, string GiveName, DateOnly Geb);
```

```
// C# 10.0
```

```
0 references | - changes | -authors, -changes
```

```
record struct RecordStruct(string Name, string GiveName, DateOnly Geb);
```

```
0 references | - changes | -authors, -changes
```

```
record RecordClassB(string Name, string GiveName, DateOnly Geb);
```


Mehrwert?

Primary constructors (C# 12)

5 references | Thorsten Kansy, 48 days ago | 1 author, 2 changes

```
public class AppLogicV1(IAppRepository AppRepository) : IAppLogic  
{  
    private readonly IAppRepository AppRepository = AppRepository;
```

2 references | Thorsten Kansy, 48 days ago | 1 author, 2 changes

```
public async Task<IList<CustomerDto>> GetCustomersAsync(CancelLa
```

Noch ein paar gefällig?



Deconstructor (Mixed
declarations)

Interpolated Verbatim Strings

...

Mixed declarations and expressions (C# 10)

Mehrwert?

```
Rect r = new Rect(10, 20);
```

```
// C# 9.0 So...
```

```
int w1, h1;
```

```
(w1, h1) = r;
```

```
// ...oder so
```

```
(int w2, int h2) = r;
```

```
// C# 10.0
```

```
int w3;
```

```
(w3, int h3) = r;
```


Interpolated Verbatim Strings

- Statt `$@"` ist nun auch `@$"` erlaubt

```
WriteLine($"File size of C:\temp\config.ini {size:N0} at {date:d}");
```

```
WriteLine(@$"File size of C:\temp\config.ini {size:N0} at {date:d}");
```

Nichts wirklich gutes?



Target type new() expression

LINQ-Operators

Keyed Service

AOT Native code

...

Target-typed new expressions (C# 10)

Vereinfachte Erstellung von Instanzen mittels new

```
// Erstellung einer neuen Instanz  
Customer customer = new Customer();  
  
// Keine Wiederholung der Klasse  
Customer customer = new();  
  
// Auch mit Parameter  
Customer customer = new("Kansy", 4711);
```


LINQ-Operators (.NET 6)

- `DistinctBy`, `UnionBy`, `IntersectBy`, `ExceptBy`
- `MaxBy` **und** `MinBy`
- `Chunk`
- `FirstOrDefault`, `LastOrDefault` **und** `SingleOrDefault`
 - Standardwert kann nun bestimmt werden (bei leeren Listen)
- Verbesserungen
 - `Zip`, `ElementAt`, `Take`
- `TryGetNonEnumeratedCount`

Keyed Service (.NET 8)

Services nicht über Typ, sondern zusätzlich über eine Schlüssel (Zeichenkette) identifizieren

```
IServiceCollection serviceCollection = new ServiceCollection();  
  
serviceCollection.AddKeyedTransient<IDemo, DemoA>("A");  
serviceCollection.AddKeyedTransient<IDemo, DemoB>("B");  
  
var services = serviceCollection.BuildServiceProvider();
```

Visual Studio und Code Analyzer



Manchmal keine echte Hilfe?
Habt Ihr eine Zero-Warning
Policy?

Wie sieht es anderswo aus?



SQL Server (T-SQL)

Verbesserungen für weniger
Abfrage-Umfang

CHOOSE()

LEAD()/ LAG()

TRIM()

NULL-Handling

...

CHOOSE() (SQL Server 2012)

Indexierter Zugriff auf eine Werteliste

```
SELECT CHOOSE(3, 'I', 'II', 'III', 'IV', 'V');
```

LAG()/ LEAD() (SQL Server 2012)

Werte aus vorherigen/ folgenden Zeilen verwenden

```
SELECT Name, LAG(Name, 1, '-') OVER (PARTITION BY  
Ort ORDER BY Geburtsdatum ASC) FROM dbo.Personen;
```

```
SELECT Name, LEAD(Name, 1, '-') OVER (PARTITION BY  
Ort ORDER BY Geburtsdatum ASC) FROM dbo.Personen;
```


TRIM() (SQL Server 2022)

Angaben, was wo entfernt werden soll

```
SELECT LTRIM(@caption, '-= ') AS [LTRIM],  
       RTRIM(@caption, '-= ') AS [RTRIM];  
  
SELECT TRIM(LEADING '-= ' FROM @caption) AS [LEADING],  
       TRIM(TRAILING '-= ' FROM @caption) AS [TRAILING],  
       TRIM(BOTH '-= ' FROM @caption) AS [BOTH];
```

NULL-Handling (SQL Server 2022)

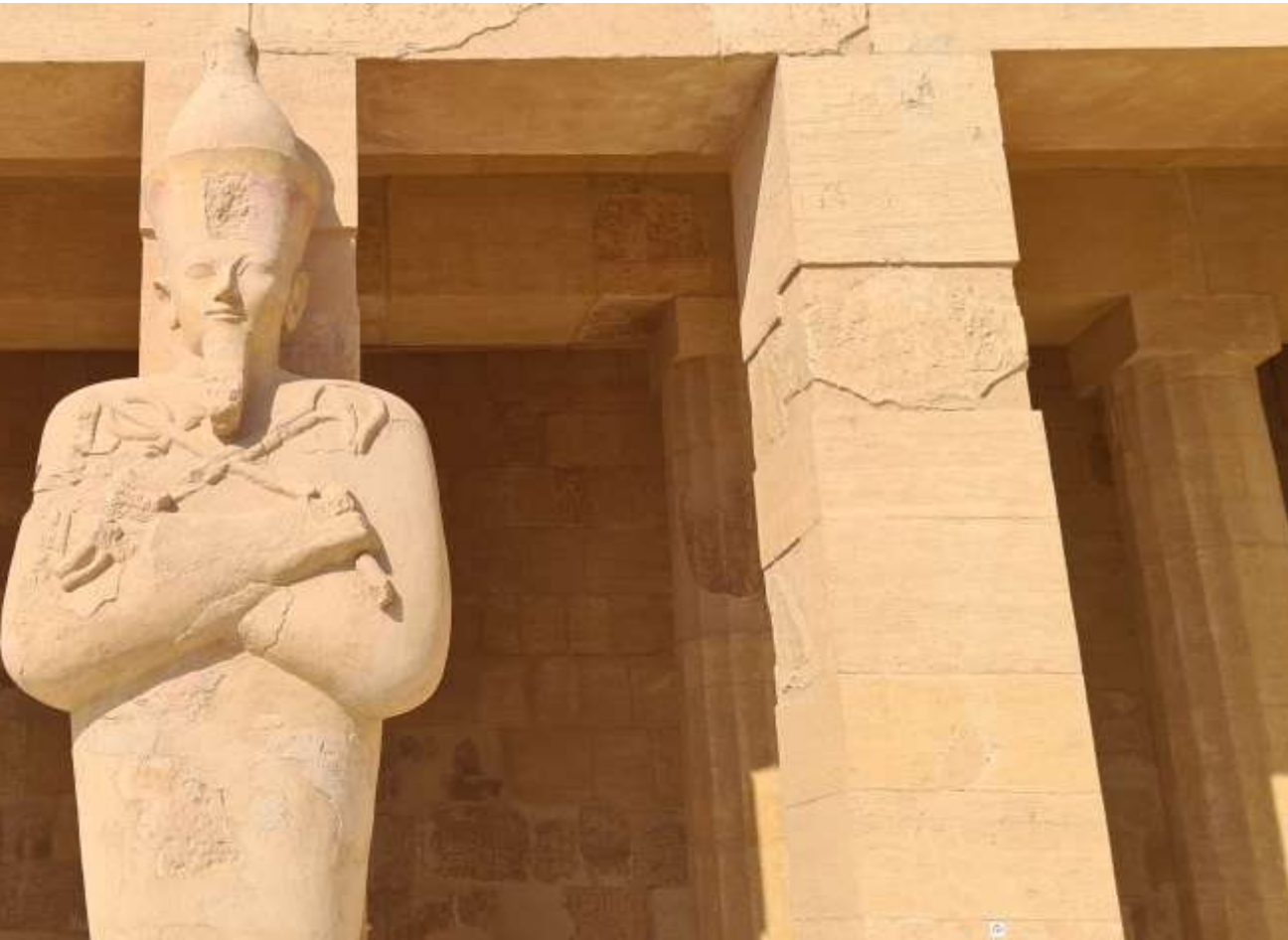
“Gleich”/ “Ungleich” als “nicht unterscheidbar”/ “unterscheidbar”

```
SELECT * FROM sys.Databases WHERE 0 IS NOT DISTINCT FROM 0; -- =
SELECT * FROM sys.Databases WHERE 0 IS NOT DISTINCT FROM 1; -- !=
SELECT * FROM sys.Databases WHERE 0 IS NOT DISTINCT FROM NULL;
SELECT * FROM sys.Databases WHERE NULL IS NOT DISTINCT FROM NULL;
```

A IS DISTINCT FROM B will decode to: ((A <> B OR A IS NULL OR B IS NULL) AND NOT (A IS NULL AND B IS NULL))

A IS NOT DISTINCT FROM B will decode to: (NOT (A <> B OR A IS NULL OR B IS NULL) OR (A IS NULL AND B IS NULL))

Für wenn sind C# & .NET?



Anfänger

Quereinsteiger

Profis

Die Hoffnung, das alles einfacher wird

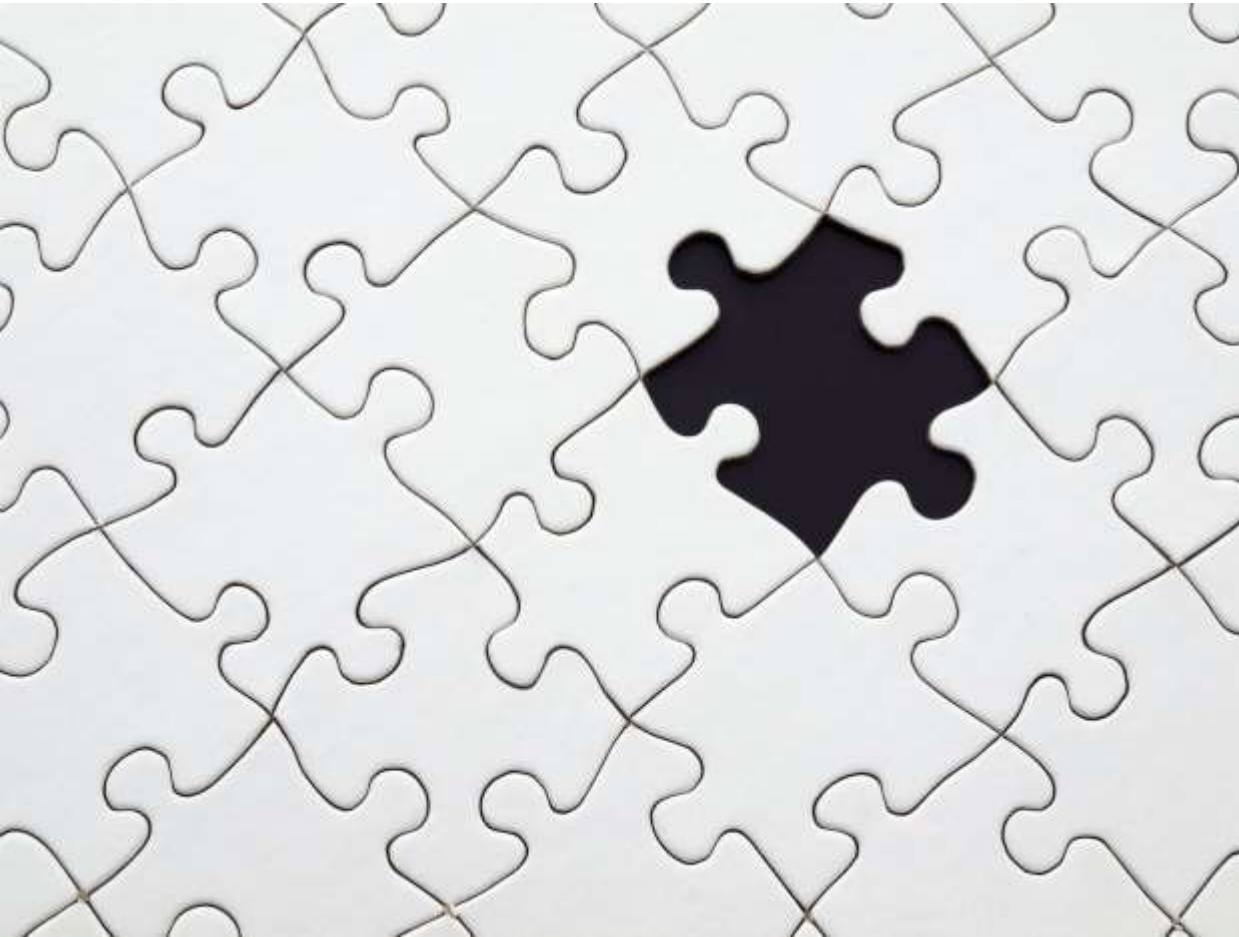


Ohne Anstrengung bessere Programme schreiben,
mehr schaffen, etc.

Mitarbeiter schaffen mehr in weniger Zeit

KI schlägt in die gleiche Kerbe

Und nun? Was ist eine/ die Lösung?



Auf den LTS-Zug aufspringen

Kein Upgrade Druck für neue Funktionen

Pfadfinder-Prinzip bei Anpassungen

Fazit

C# ist für alle, von Anfänger bis Profi

Es wird immer mehr, und man muss genau überlegen, welche Änderungen passen sind

Es wird auch nicht einfacher, da es immer mehr Möglichkeiten gibt ein Problem zu lösen

Aber: viele erreichen mit C# & .NET das angepeilte Ziel

Fragen? Jetzt oder später!

Kontakt

 **E-Mail**
tkansy@dotnetconsulting.eu

 **Telefon**
[+49 \(0\) 6187 / 2009090](tel:+49(0)61872009090)

 **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.