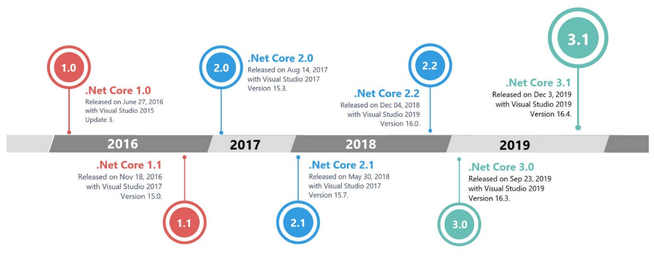
WLAN
DDCDCC
Passwort: Developer23

.NET ist gekommen, um zu bleiben

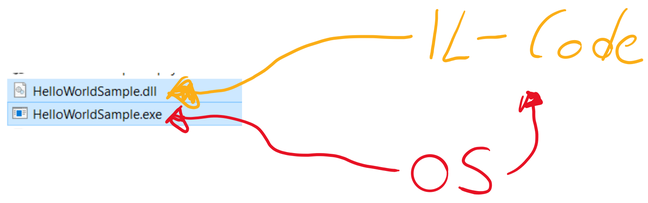


.NET (Core)
* IDE (VS2022, Visual Studio Code, JetBrains Rider)
* .NET (Core) SDK

* Runtime Installation (https://dotnet.microsoft.com/en-us/download/dotnet/8.0)

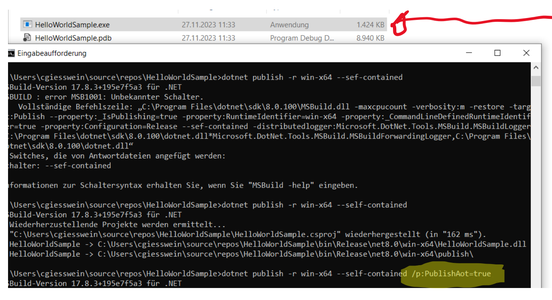
--> .NET CLI (Command Line Interface)
--> CI/CD

dotnet CLI
> dotnet new
> dotnet build
> dotnet run
> dotnet pack
> dotnet test
> dotnet restore
> dotnet publish



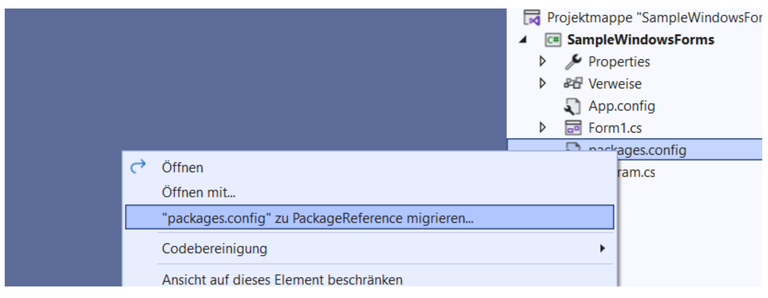

```
dotnet publish -r win-x64 --self-contained
```

> Runtime Installation notwendig
> Self-Contained (inkl. Runtime / GC / JIT ...)
> AOT (Ahead of Time Compilation)



Alles für den Betrieb der Anwendung (CLR, GC)..
> Dynamischer Code wird nicht supported

dotnet publish -r win-x64 --self-contained /p:PublishSingleFile=true



Migration alter Packages.config Dateien

> packages.config wird entfernt
> Packages folder wird entfernt
> *.csproj wird um PackageReference angepasst

```
<ItemGroup>
  <Reference Include="Newtonsoft.Json, Version=13.0.0.0,
    <HintPath>..\packages\Newtonsoft.Json.13.0.3\lib\net4
  </Reference>
  <Reference Include="System" />
```

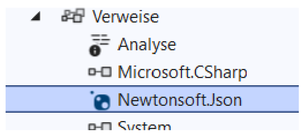
```
<ItemGroup>
  <PackageReference Include="Newtonsoft.Json">
    <Version>13.0.3</Version>
  </PackageReference>
</ItemGroup>
```

Migrationstools

> https://learn.microsoft.com/de-de/dotnet/core/porting/upgrade-assistant-install#install-the-net-global-tool

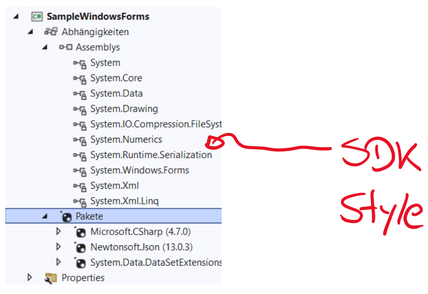
> https://github.com/hvanbakel/CsprojToVs2017

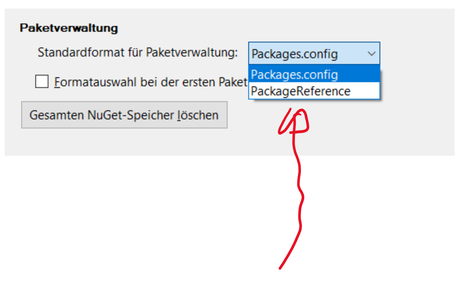
> https://dotnet.microsoft.com/en-us/platform/upgrade-assistant



Probleme bei der Migration
> WCF Hosting ~ --> CoreWCF (https://github.com/CoreWCF/CoreWCF)
---> .NET Core unterstützt WCF Clients (!) --> Per Nuget System.ServiceModel.* nachinstallieren
> WebForms --> Geht nicht!

> AppDomains --> Geht nicht! --> Laufzeitfehler
> .NET Remoting --> Geht nicht!
> BinaryFormatter --> Serialisierung --> Warnung wird kurzfristig noch funktionieren (Achtung: Drittkomponenten) - https://github.com/dotnet/designs/blob/main/accepted/2020/better-obsoletion/binaryformatter-obsoletion.md





Visual Studio Option für neue .NET Framework Projekte

Migrationspfad
---
* PackageReference statt Packages.config (Kompatible mit .NET FX und .NET Core)
* CSPROJ - SDK Style Format (Kompatible mit .NET FX und .NET Core)
-> Upgrade Assistant (Microsoft)

Ohne Code Umbau möglich
---
* Analyse welche Code-"Technologien" werden verwendet (AppDomain, BinaryFormatter, WCF, WebForms...)
---
* Umstellung TargetFramework von net471 auf net6.0, net7.0, .net8.0
---
* Frameworks, Libraries, Nuget Updates (EF6 / EF Core, Newtonsoft.Json, System.Data.SqlClient / Microsoft.Data.SqlClient, Dritt Hersteller Nuget)

```
<TargetFramework>net472</TargetFramework>
```

net47
net471
net472
net48

netcoreapp3.0
netcoreapp3.1
net5.0
net6.0
net7.0
net8.0
....
net7.0-windows
net8.0-windows

.NET4.7 / .NET 4.8 --> C# 7.3
.NET 5 --> C# 9
.NET 6 --> C# 10
.NET 7 --> C# 11
.NET 8 --> C# 12

```
<PropertyGroup>
    <TargetFrameworks>net471;net5.0-windows;net8.0
    <UseWindowsForms>true</UseWindowsForms>
    <LangVersion>12</LangVersion>
</PropertyGroup>

<ItemGroup>
```

Projekteinstellung um auch in .NET Framework Projekte C# 12 zu verwenden

---