Что нового в .NET 7 и C# 11

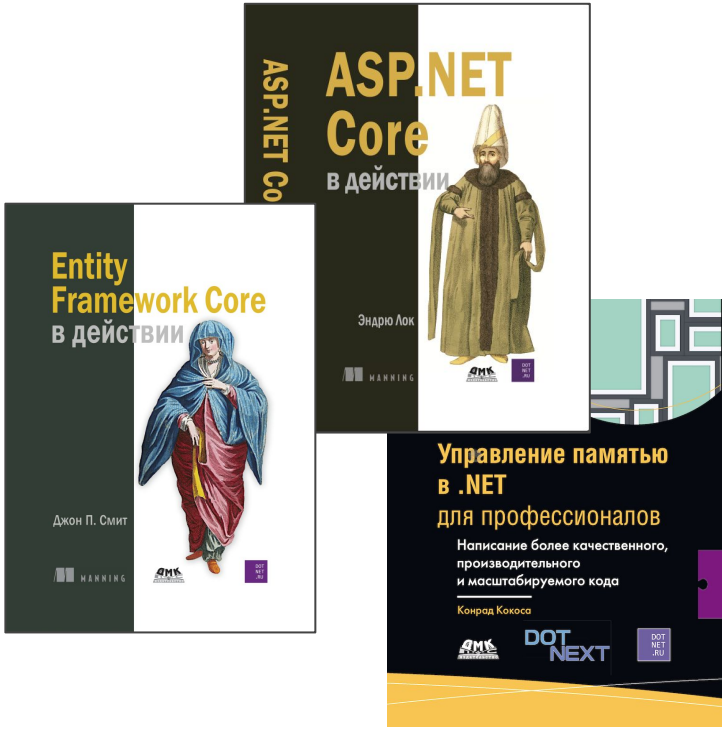
Игорь Лабутин

@ilabutin

Март 2023

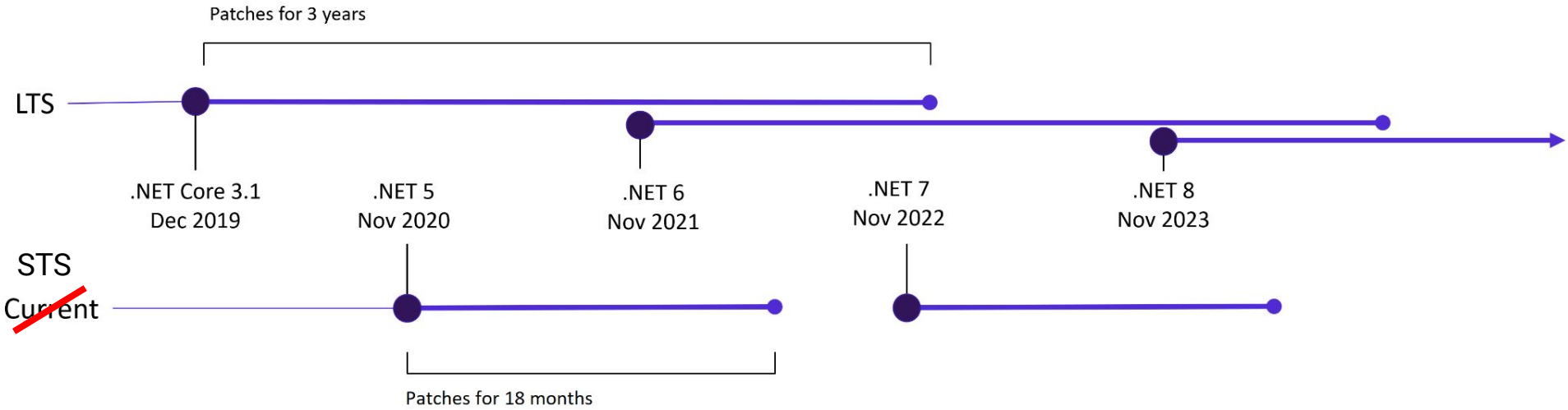
DOT
NET
.RU

RADIO
DOT
NET



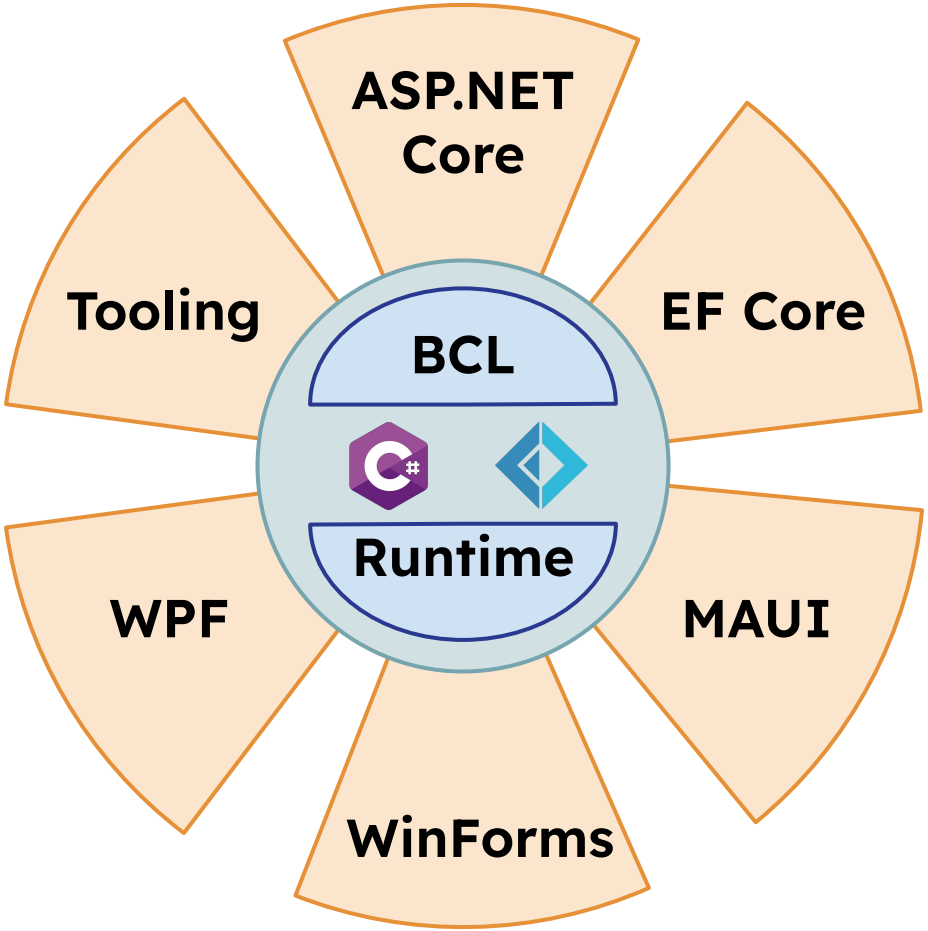
Lanit-Tercom
smart software solutions

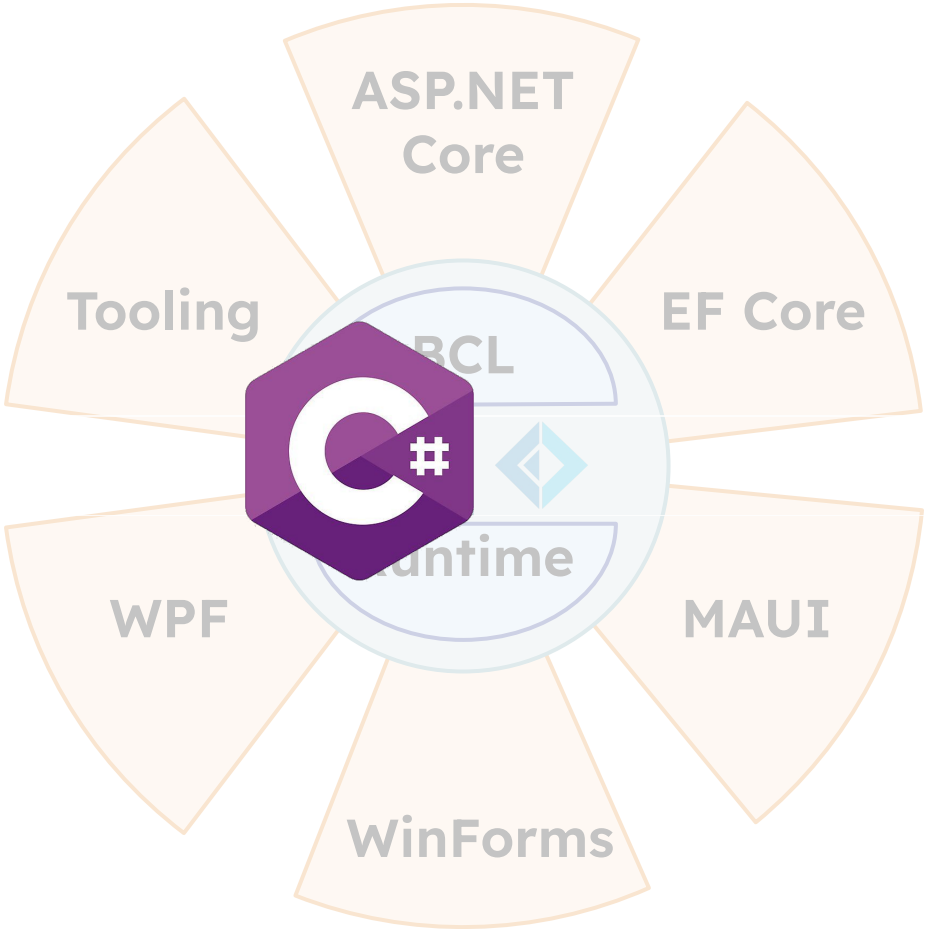
.NET 7



"The quality of all releases are exactly the same"

.NET Official Support Policy


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms


ASP.NET
Core
Tooling
EF Core
BCL
C#
Runtime
WPF
MAUI
WinForms

C# 11

- Required members
- Autodefault structs
- Raw string literals
- Newlines in string interpolation
- Generic attributes
- List patterns
- Static abstract members in interfaces
- Generic math
- Utf-8 string literals
- File-scoped types
- Numeric IntPtr and UIntPtr
- Extended nameof scope
- Unsigned right-shift
- Pattern match on const strings for Span<char>

DEMO TIME

https://devblogs.microsoft.com/dotnet/welcome-to-csharp-11/

```
var pointWithInitializer = new Point() { X = 1, Y = 2 };
var pointWithPartialInit = new Point() { X = 3 };
var pointWithCtor = new Point(x: 5, y: 6);

public class Point
{
  public Point()
  {
  }

  //[SetsRequiredMembers]
  public Point(int x, int y)
  {
    X = x;
    Y = y;
  }

  public required int X { get; init; }
  public required int Y { get; init; }
}
```

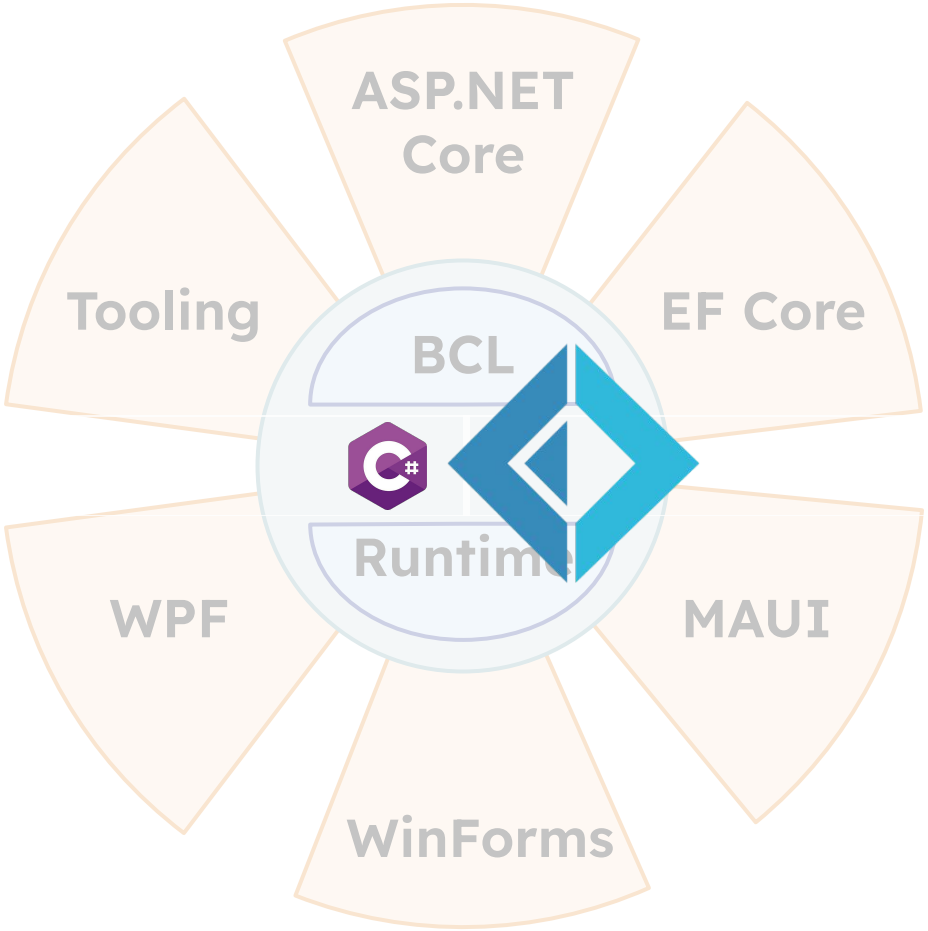
```
string newJson = """
    {
      "key": "author",
      "value": "Igor"
    }
    """;
```

```
string newJsonNoIndent = """
    {
      "key": "author",
      "value": "Igor"
    }
""";
```

```
string newResult = $$"""{ "key": "author", "value": "{{
    (author != null
    ? author.ToUpper()
    : "no author") }}" }""";
```

```
int[] numbers = { 0, 1, 2, 3, 4 };

if (numbers is [0, 1, 2, 3, 4])
  "Numbers is { 0, 1, 2, 3, 4 }".Dump();

if (numbers is [0, ..])
  "Numbers starts with 0".Dump();

if (numbers is [0, _, _, _, _])
  "Numbers starts with 0, and has 5 elements".Dump();

if (numbers is [_, _, _, _, var last])
  last.Dump ("Last number in the series of 5");

if (numbers is [.., var last2])
  last2.Dump("Last number in the series");

if (numbers is [0, .. var rest])
  rest.Dump ("Numbers after the zero");

if (numbers is [0, .. var middle, 4])       // "slice"
  middle.Dump("Numbers in the middle");

if (numbers is [_, _, >2, ..])
  "Third number above 2".Dump();
```

```
// Define interface with Parse method
interface IParseable<T> where T : IParseable<T>
{
  static abstract T Parse(string s);
}

// Define record which implements the interface
record Point(int X, int Y) : IParseable<Point>
{
  public static Point Parse(string s)
  {
    var match = Regex.Match(s, @"Point { X = (\d+), Y = (\d+) }");
    if (!match.Success) throw new ArgumentException("Cannot parse point - invalid string");
    return new Point(int.Parse(match.Groups[1].Value), int.Parse(match.Groups[2].Value));
  }
}

// Define helper function for easier calling
T ParseAny<T>(string s) where T : IParseable<T> => T.Parse(s);
```


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms

F#

- Static abstract members в интерфейсах

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/

F#

- Static abstract members в интерфейсах
- Поддержка required properties

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/

F#

- Static abstract members в интерфейсах
- Поддержка required properties
- Упрощение работы с SRTP

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/

F#

- Static abstract members в интерфейсах
- Поддержка required properties
- Упрощение работы с SRTP
- Поддержка reference assemblies

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/

F#

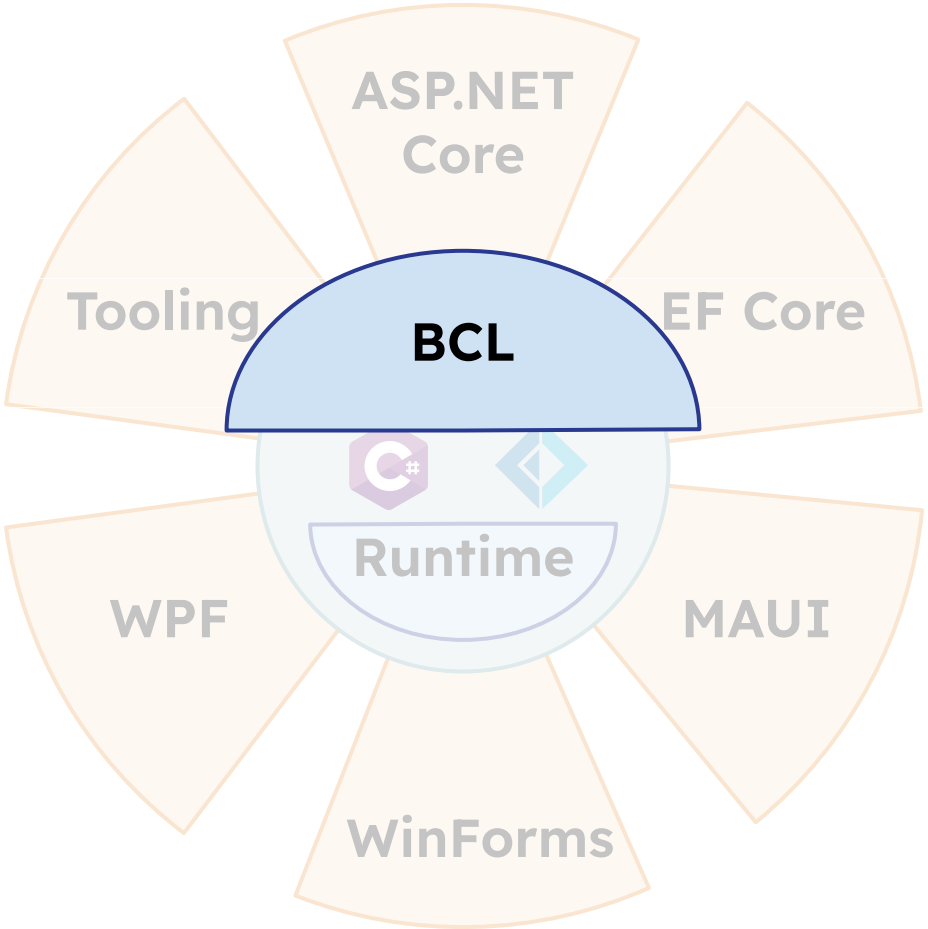
- Static abstract members в интерфейсах
- Поддержка required properties
- Упрощение работы с SRTP
- Поддержка reference assemblies
- Сборка self-contained и native AOT

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/

F#

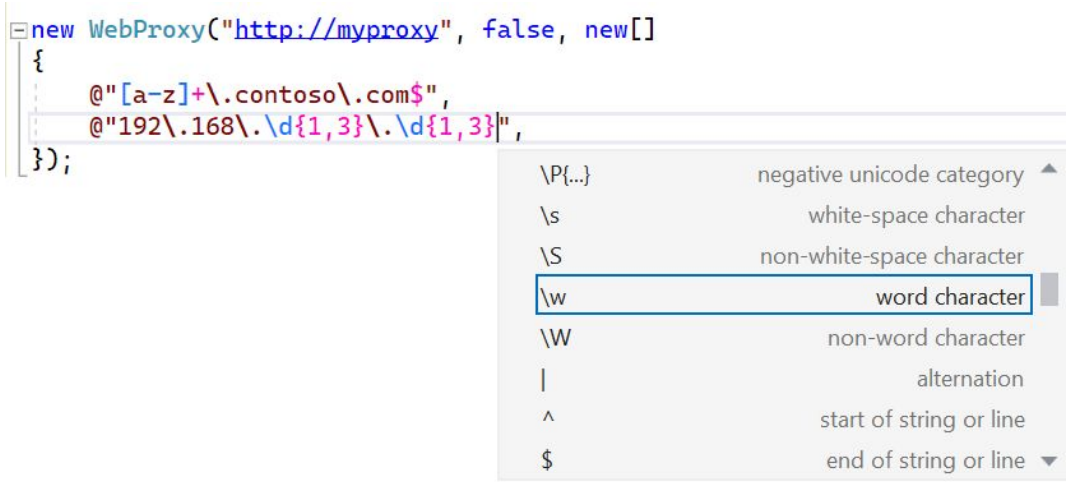
- Static abstract members в интерфейсах
- Поддержка required properties
- Упрощение работы с SRTP
- Поддержка reference assemblies
- Сборка self-contained и native AOT
- Поддержка ARM64

https://devblogs.microsoft.com/dotnet/announcing-fsharp-7/


ASP.NET Core
Tooling
BCL
EF Core
Runtime
WPF
MAUI
WinForms

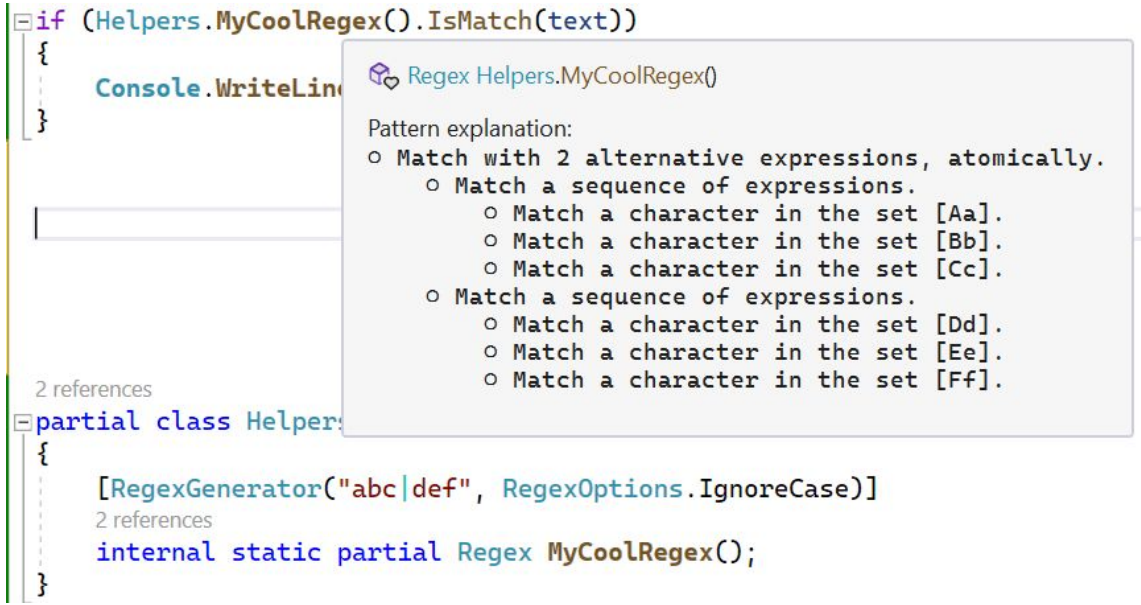
StringSyntaxAttribute

- StringSyntaxAttribute
  - CompositeFormat
  - DateOnlyFormat
  - DateTimeFormat
  - EnumFormat
  - GuidFormat
  - Json
  - NumericFormat
  - Regex
  - TimeOnlyFormat
  - TimeSpanFormat
  - Uri
  - Xml



Регулярные выражения - генератор кода

- Скорость
- Отладка



https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd
```

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd

Шаг 1: пробуем .+ с aaabc -> следующий d не подходит
```

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd

Шаг 1: пробуем .+ с aaabc -> следующий d не подходит
Шаг 2: пробуем .+ с aaab -> следующий c не подходит
```

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd

Шаг 1: пробуем .+ с aaabc -> следующий d не подходит
Шаг 2: пробуем .+ с aaab -> следующий c не подходит
Шаг 3: пробуем .+ с aaa -> следующий b подходит
```

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd

Шаг 1: пробуем .+ с aaabc -> следующий d не подходит
Шаг 2: пробуем .+ с aaab -> следующий c не подходит
Шаг 3: пробуем .+ с aaa -> следующий b подходит
```

- `RegexOptions.NonBacktracking`

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - новый режим

```
Regex: .+b
Input: aaabcd

Шаг 1: пробуем .+ с aaabc -> следующий d не подходит
Шаг 2: пробуем .+ с aaab -> следующий c не подходит
Шаг 3: пробуем .+ с aaa -> следующий b подходит
```

- `RegexOptions.NonBacktracking`
- `NonBacktracking` не совместимо с `Regex.Compiled` и кодогенератором

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Регулярные выражения - оптимизации

- `RegexOptions.IgnoreCase`
- `Span<T>`
- Векторизация

Method	Runtime	Mean
ManualSet1	.NET 6	85.75 us
IgnoreCase2	.NET 6	235.40 us
ManualSet3	.NET 6	4.312 us
IgnoreCase4	.NET 6	222.387 us
ManualSet1	.NET 7	47.167 us
IgnoreCase2	.NET 7	47.130 us
ManualSet3	.NET 7	4.147 us
IgnoreCase4	.NET 7	4.135 us

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

Приятные мелочи

- Nullable аннотации для Microsoft.Extensions

Приятные мелочи

- Nullable аннотации для Microsoft.Extensions
- Поддержка TAR

Приятные мелочи

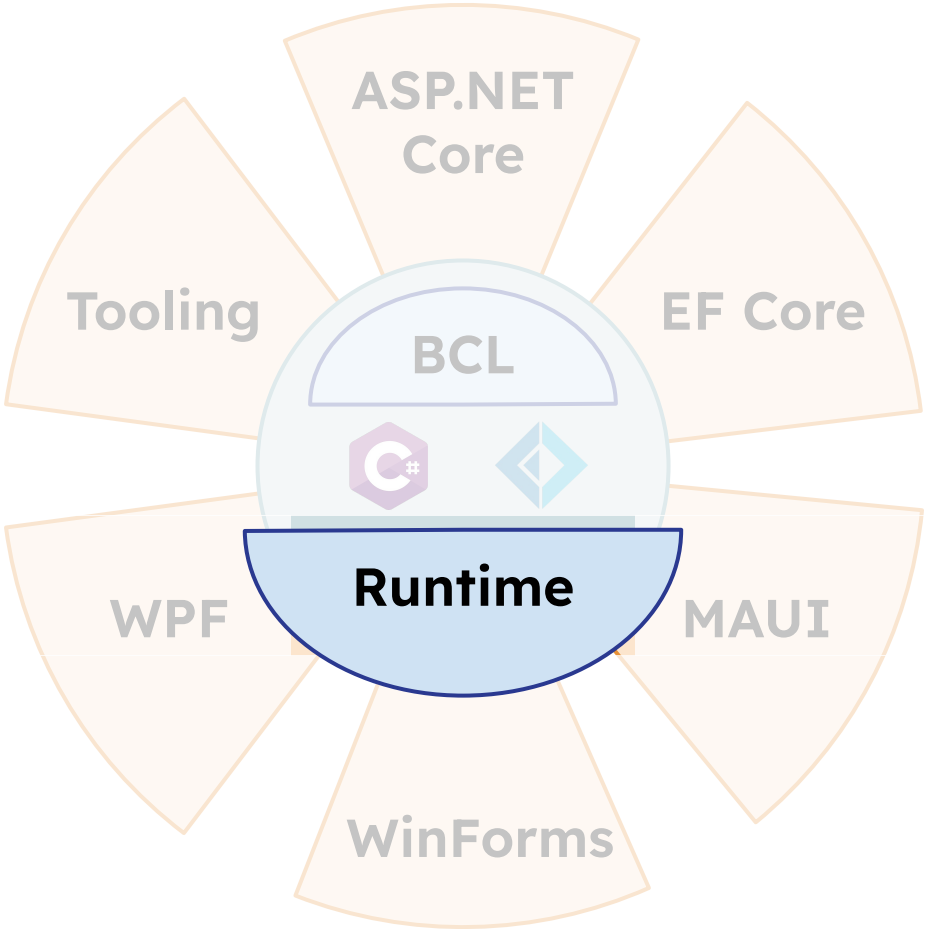
- Nullable аннотации для Microsoft.Extensions
- Поддержка TAR
- `DllImport` -> `LibraryImport`

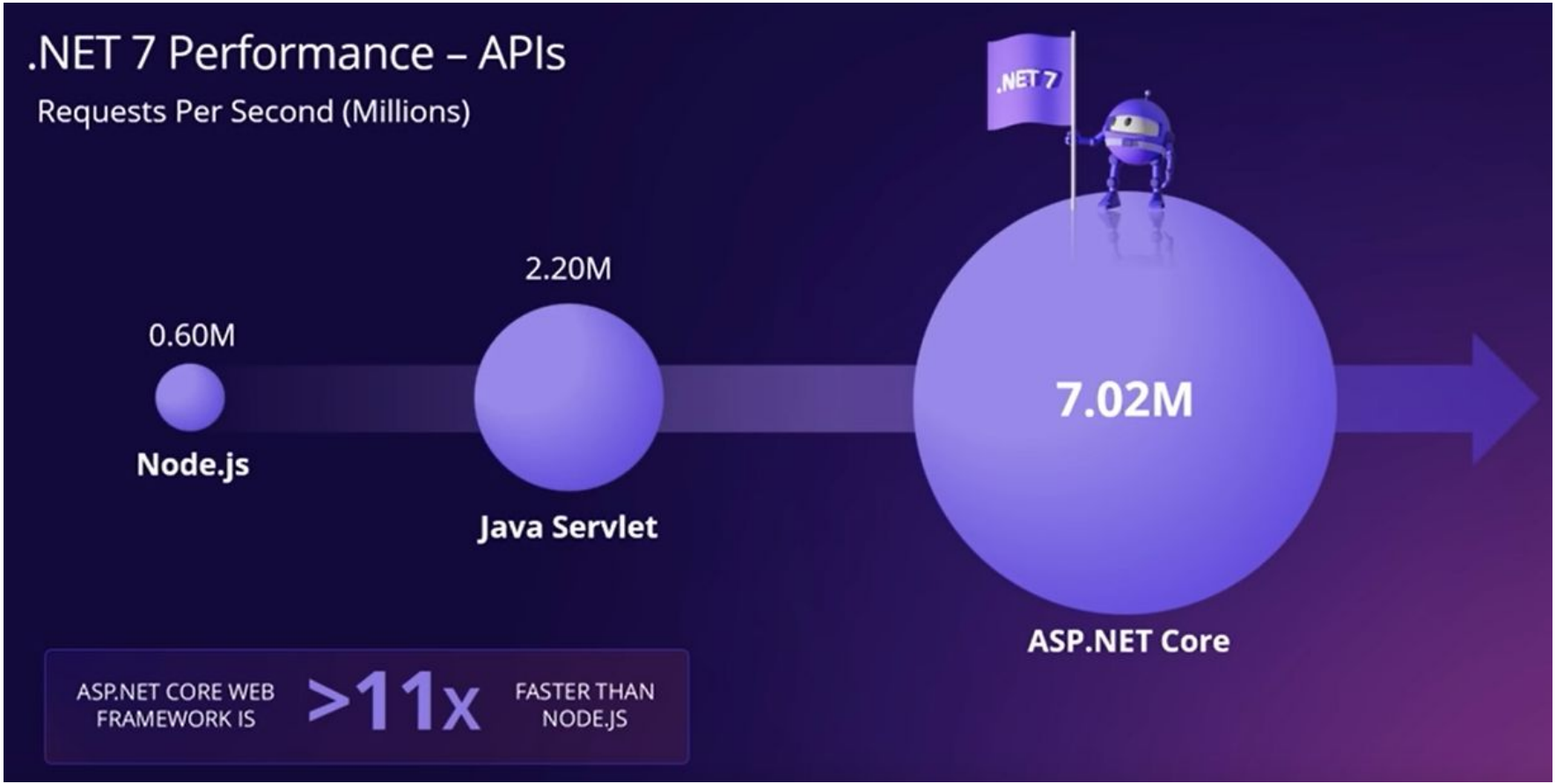
Приятные мелочи

- Nullable аннотации для Microsoft.Extensions
- Поддержка TAR
- `DllImport` -> `LibraryImport`
- Микро- и наносекунды

Приятные мелочи

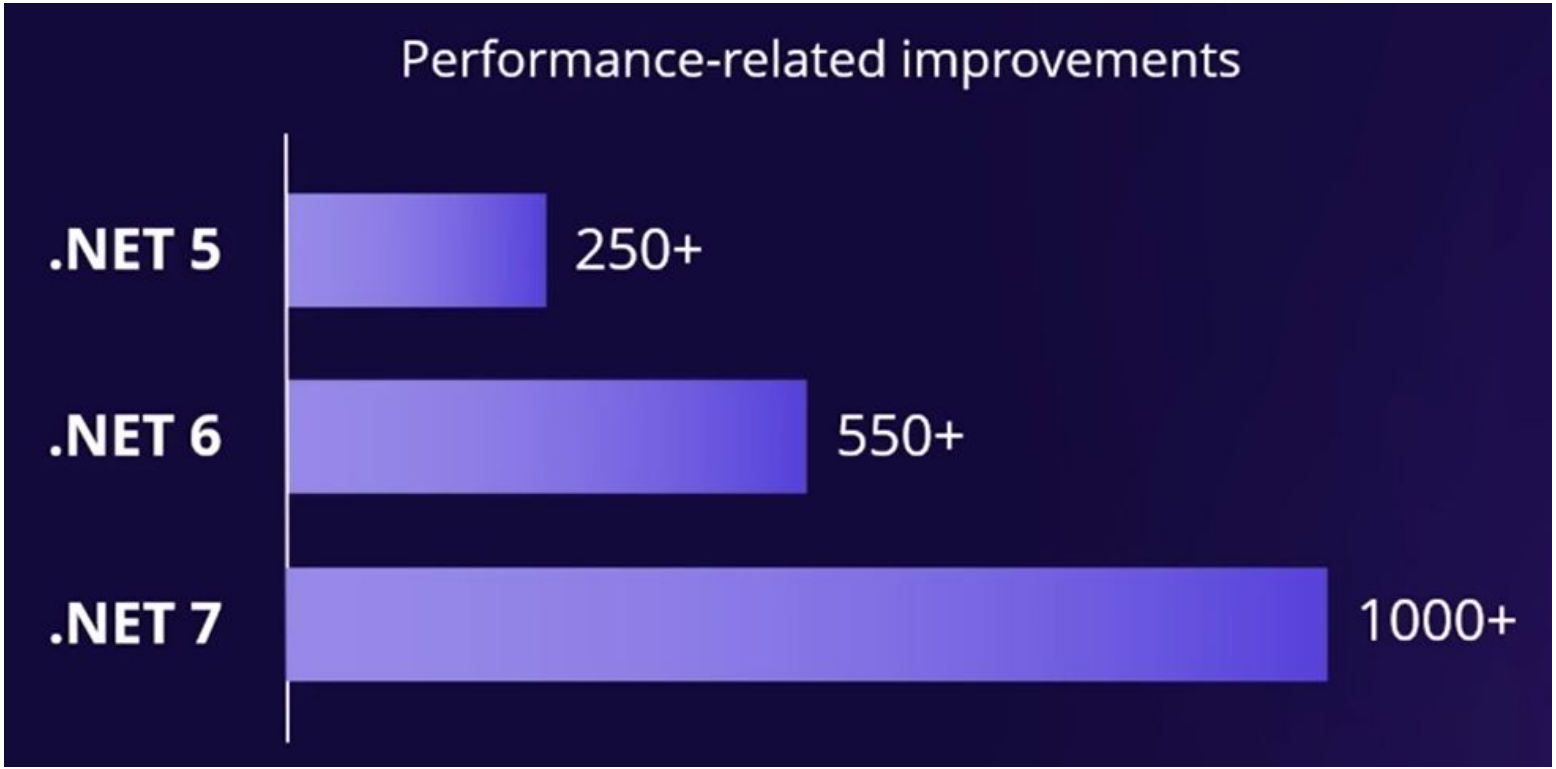
- Nullable аннотации для Microsoft.Extensions
- Поддержка TAR
- `DllImport` -> `LibraryImport`
- Микро- и наносекунды
- `System.IO.Stream`: `ReadAtLeast`/`ReadExactly`


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms



https://www.youtube.com/watch?v=8V_BUGFKdaI

Ускоряемся!


Performance-related improvements
.NET 5
250+
.NET 6
550+
.NET 7
1000+

Ускоряемся!

- Method group conversion для делегатов

Ускоряемся!

- Method group conversion для делегатов
- GC: Регионы
  - Включены по умолчанию
  - Ускоряют фазу разметки

Ускоряемся!

- Method group conversion для делегатов
- GC: Регионы
  - Включены по умолчанию
  - Ускоряют фазу разметки
- Reflection

Ускоряемся!

- Method group conversion для делегатов
- GC: Регионы
  - Включены по умолчанию
  - Ускоряют фазу разметки
- Reflection
- JIT
  - Dynamic PGO

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT
- Добавляем `<PublishAot>true</PublishAot>` в .csproj

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT
- Добавляем `<PublishAot>true</PublishAot>` в .csproj
- Собираем `dotnet publish -r linux-arm64 -c Release`

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT
- Добавляем `<PublishAot>true</PublishAot>` в .csproj
- Собираем `dotnet publish -r linux-arm64 -c Release`
- Можно собирать нативные .DLL/.SO

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT
- Добавляем `<PublishAot>true</PublishAot>` в .csproj
- Собираем `dotnet publish -r linux-arm64 -c Release`
- Можно собирать нативные .DLL/.SO
- Быстрее запуск

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

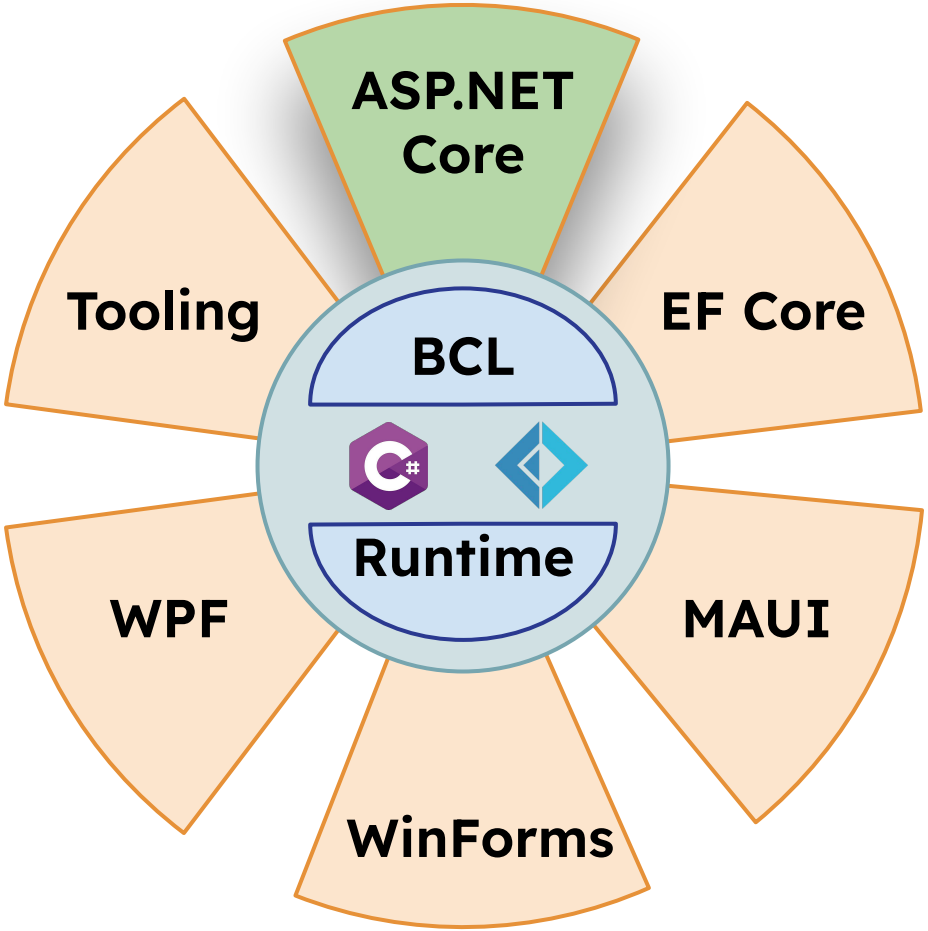
Native AOT

- .NET давно поддерживает (ngen, crossgen)
  - Все еще требует JIT
- Native AOT это другое!
- Полностью нативное приложение, без зависимостей, без JIT
- Добавляем `<PublishAot>true</PublishAot>` в .csproj
- Собираем `dotnet publish -r linux-arm64 -c Release`
- Можно собирать нативные .DLL/.SO
- Быстрее запуск

https://learn.microsoft.com/en-gb/dotnet/core/deploying/native-aot/
https://github.com/dotnet/runtime/tree/main/src/coreclr/nativeaot/docs

Native AOT - Ограничения

- Нет динамической загрузки сборок
- Нельзя генерировать код в рантайме
- Нет поддержки C++/CLI
- Нет поддержки COM
- Требует использования assembly trimming
- Сборка в один файл
- Все зависимости включены в итоговый файл


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms

Что такое ASP.NET Core

- Сервера и рантайм
- MVC
- Minimal APIs
- gRPC
- SignalR
- Blazor

Начнем с простого

- Параметры
  - Атрибут `FromServices`
  - `IParsable<T>.TryParse` для привязки параметров

Начнем с простого

- Параметры
  - Атрибут `FromServices`
  - `IParsable<T>.TryParse` для привязки параметров
- Результаты
  - `Result.Stream`
  - Пространство имен `HttpResults` - все реализации `IResult`

Minimal API

- Типизированные результаты
- Поддержка OpenAPI
- Группировка маршрутов
- Фильтры обработчиков маршрутов
- Привязка параметров (массивы, streams)

DEMO TIME

System.Text.Json

- Полиморфизм
  - [JsonDerivedType] на базовый тип
  - [JsonPolymorphic] для кастомизации поведения

https://devblogs.microsoft.com/dotnet/system-text-json-in-dotnet-7/

System.Text.Json

- Полиморфизм
  - [JsonDerivedType] на базовый тип
  - [JsonPolymorphic] для кастомизации поведения
- Кастомизация контрактов JSON
  - Через `JsonSerializerOptions.TypeInfoResolver`
  - Меняем `JsonPropertyInfo` в резолвере

https://devblogs.microsoft.com/dotnet/system-text-json-in-dotnet-7/

Middlewares

- Rate limiting
- Output caching
- Request decompression

gRPC

- gRPC как JSON API

gRPC

- gRPC как JSON API
  - Изменяем .proto

```
service Greeter {
  rpc SayHello (HelloRequest) returns (HelloReply) {
    option (google.api.http) = {
      get: "/v1/greeter/{name}"
    };
  }
}
```

gRPC

- gRPC как JSON API
  - Изменяем .proto

```
service Greeter {
  rpc SayHello (HelloRequest) returns (HelloReply) {
    option (google.api.http) = {
      get: "/v1/greeter/{name}"
    };
  }
}
```

  - Добавляем сервис: `services.AddGrpc().AddJsonTranscoding()`

gRPC

- gRPC как JSON API
  - Изменяем .proto

```
service Greeter {
  rpc SayHello (HelloRequest) returns (HelloReply) {
    option (google.api.http) = {
      get: "/v1/greeter/{name}"
    };
  }
}
```

  - Добавляем сервис: `services.AddGrpc().AddJsonTranscoding()`
- Только server-side стриминг

gRPC

- gRPC как JSON API
  - Изменяем .proto

```
service Greeter {
  rpc SayHello (HelloRequest) returns (HelloReply) {
    option (google.api.http) = {
      get: "/v1/greeter/{name}"
    };
  }
}
```

  - Добавляем сервис: `services.AddGrpc().AddJsonTranscoding()`
- Только server-side стриминг
- Экспериментальная поддержка OpenAPI: `services.AddGrpcSwagger();`

Observability

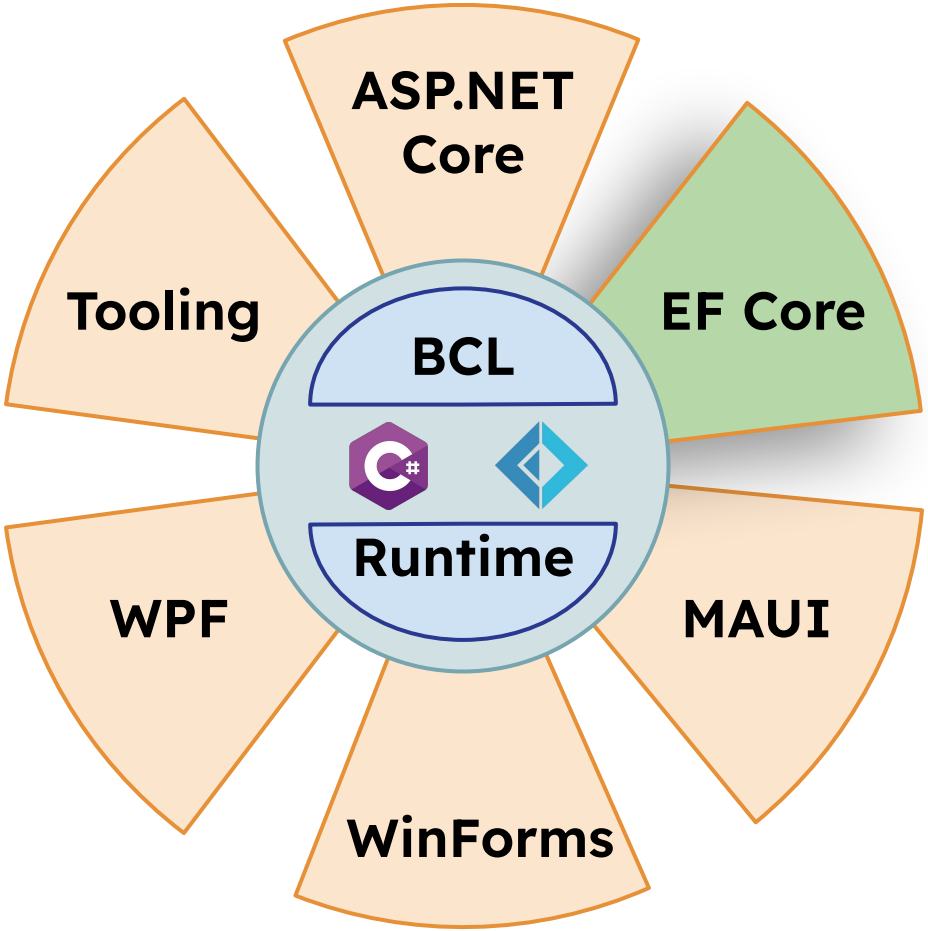
- Поддержка OpenTelemetry
- `System.Diagnostics.Activity`
  - Событие `CurrentChanged`
  - Performant enumerators
- Метрики для `IMemoryCache`

Blazor WASM

- Интероп с JavaScript: `[JSImport]`/`[JSExport]`
- AOT + SIMD
- Исключения в WASM
- Кастомизация запросов авторизации в рантайме
- Экспериментальная поддержка многопоточности

Сервера

- WebSockets поверх HTTP/2
- Официально поддержан HTTP/3
- Экспериментальная поддержка WebTransport поверх HTTP/3

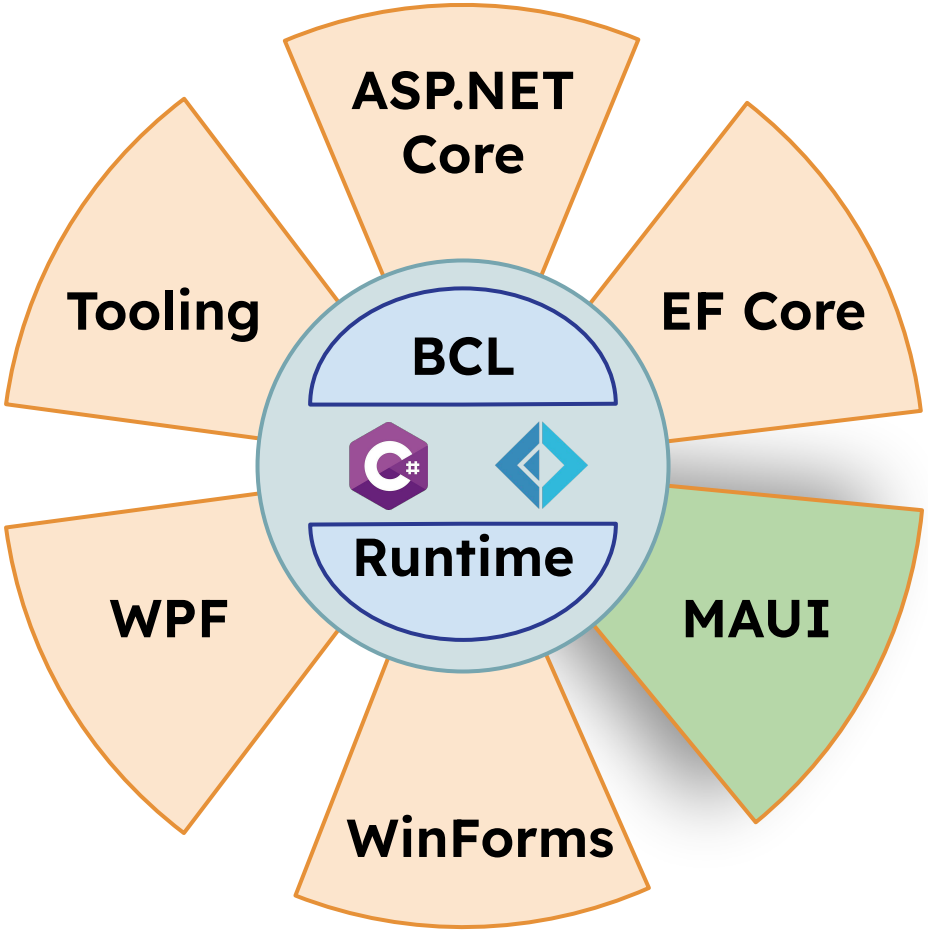

ASP.NET Core
EF Core
MAUI
WinForms
WPF
Tooling
BCL
Runtime

EF Core - новые фичи

- Поддержка Table-per-concrete-type
- Поддержка JSON колонок
- Работа с хранимыми процедурами для insert/update/delete
- Шаблоны T4 для DB-first

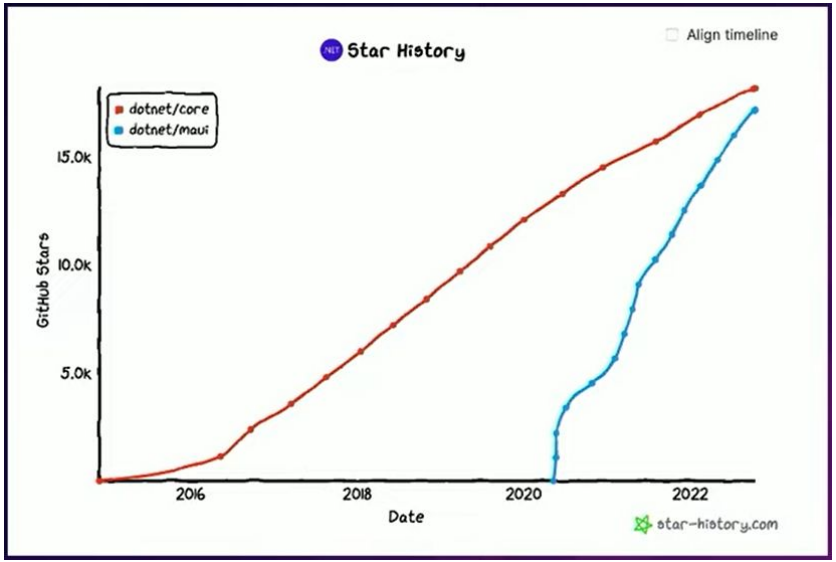
EF Core - улучшение производительности

- EF Core 6 - про чтение
- EF Core 7 - про запись
- Bulk updates/deletes
- Ускорение SaveChanges


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms

MAUI

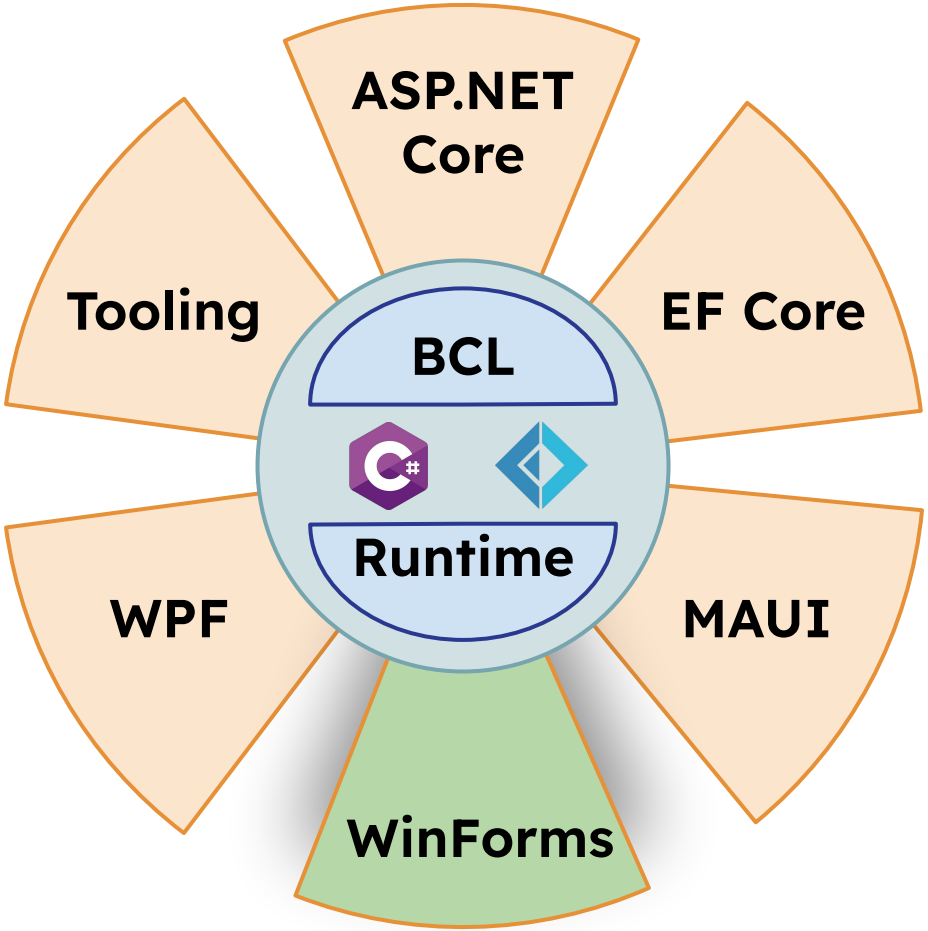
- Поддержка Android, iOS, MacOS, Windows (+Tizen by Samsung)
  - Поддержка Linux - только силами сообщества
- Ускорился рендеринг
- Новые контролы
- Поддержка desktop приложений



MAUI - примеры

- Новое “модельное” приложение - Подкасты
- ASP.NET Core (Minimal APIs, Razor)
- Blazor WASM
- MAUI app
- Hybrid app

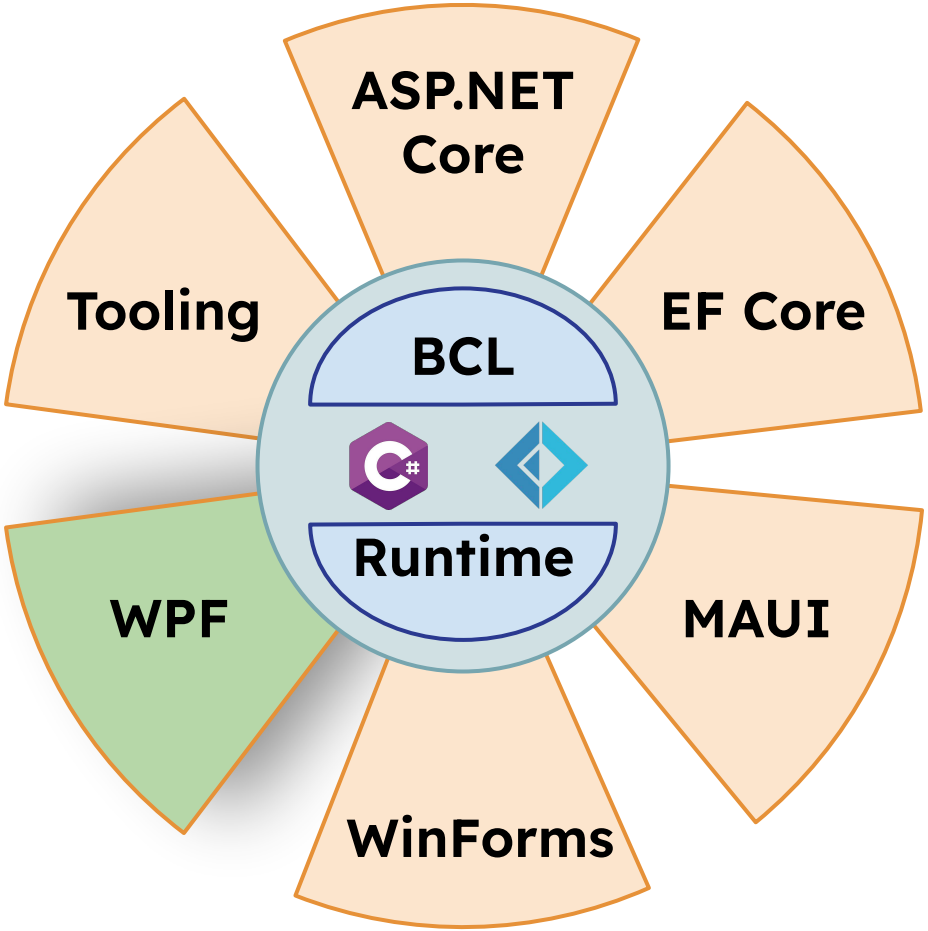
https://github.com/microsoft/dotnet-podcasts


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms

WinForms

- Доступность (accessibility)
- Поддержка HighDPI
- Экспериментальная поддержка ComWrappers и Native AOT
- Экспериментальная превью поддержка новых data bindings

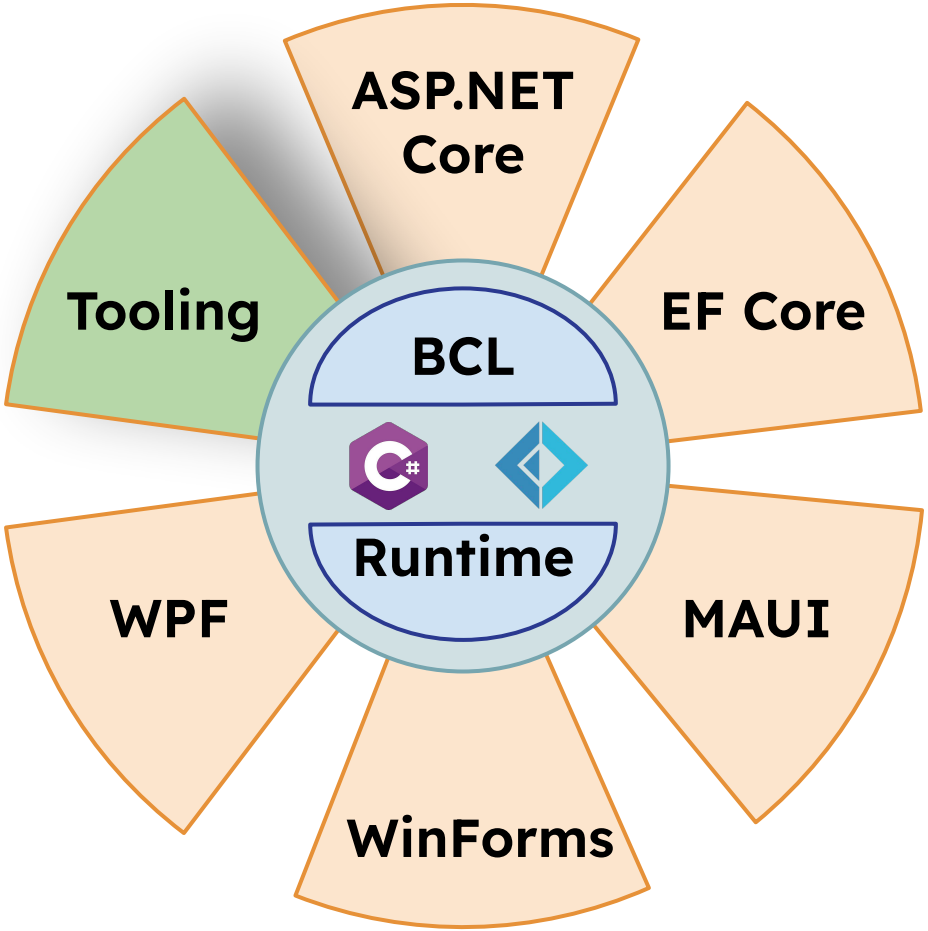
https://devblogs.microsoft.com/dotnet/winforms-enhancements-in-dotnet-7


ASP.NET Core
Tooling
EF Core
BCL
Runtime
WPF
MAUI
WinForms

WPF

- Доступность
- Оптимизация памяти
- Упрощение сторонних контрибьютов
  - Инфраструктура тестов
  - Запуск тестов на PR

https://devblogs.microsoft.com/dotnet/wpf-on-dotnet-7/


ASP.NET Core
EF Core
MAUI
WinForms
WPF
Tooling
BCL
Runtime

Build/tooling

- Visual Studio 2022 17.4
- Rider 2022.3 EAP
- .NET SDK
  - Дополнительные компоненты через “dotnet workload”

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

- Настройка через свойства в .csproj

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

- Настройка через свойства в .csproj
  - Но не всё еще реализовано

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

- Настройка через свойства в .csproj
  - Но не всё еще реализовано
- Сейчас только для Linux-x64

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

- Настройка через свойства в .csproj
  - Но не всё еще реализовано
- Сейчас только для Linux-x64
- Нет авторизации для докера, поэтому push не сработает

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Поддержка контейнеров

- Сборка контейнера без Dockerfile

```
# add a reference to a (temporary) package that creates the container
dotnet add package Microsoft.NET.Build.Containers

# publish your project for linux-x64
dotnet publish --os linux --arch x64 -c Release -p:PublishProfile=DefaultContainer
```

- Настройка через свойства в .csproj
  - Но не всё еще реализовано
- Сейчас только для Linux-x64
- Нет авторизации для докера, поэтому push не сработает

https://devblogs.microsoft.com/dotnet/announcing-builtin-container-support-for-the-dotnet-sdk/

Централизованное управление пакетами

- Directory.Packages.props

```
<Project>
  <PropertyGroup>
    <ManagePackageVersionsCentrally>true</ManagePackageVersionsCentrally>
  </PropertyGroup>
  <ItemGroup>
    <PackageVersion Include="Newtonsoft.Json" Version="13.0.1" />
  </ItemGroup>
</Project>
```

Централизованное управление пакетами

- Directory.Packages.props

```
<Project>
  <PropertyGroup>
    <ManagePackageVersionsCentrally>true</ManagePackageVersionsCentrally>
  </PropertyGroup>
  <ItemGroup>
    <PackageVersion Include="Newtonsoft.Json" Version="13.0.1" />
  </ItemGroup>
</Project>
```

- .csproj

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

Централизованное управление пакетами

- Directory.Packages.props

```
<Project>
  <PropertyGroup>
    <ManagePackageVersionsCentrally>true</ManagePackageVersionsCentrally>
  </PropertyGroup>
  <ItemGroup>
    <PackageVersion Include="Newtonsoft.Json" Version="13.0.1" />
  </ItemGroup>
</Project>
```

- .csproj

```
<ItemGroup>
  <PackageReference Include="Newtonsoft.Json" VersionOverride="12.0.0"/>
</ItemGroup>
```

Централизованное управление пакетами

- Directory.Packages.props

```
<Project>
  <PropertyGroup>
    <ManagePackageVersionsCentrally>true</ManagePackageVersionsCentrally>
  </PropertyGroup>
  <ItemGroup>
    <PackageVersion Include="Newtonsoft.Json" Version="13.0.1" />
  </ItemGroup>
</Project>
```

- .csproj

```
<ItemGroup>
  <PackageReference Include="Newtonsoft.Json" VersionOverride="12.0.0"/>
</ItemGroup>
```

- Транзитивное закрепление версий

Централизованное управление пакетами

- Directory.Packages.props

```
<Project>
  <PropertyGroup>
    <ManagePackageVersionsCentrally>true</ManagePackageVersionsCentrally>
  </PropertyGroup>
  <ItemGroup>
    <PackageVersion Include="Newtonsoft.Json" Version="13.0.1" />
  </ItemGroup>
</Project>
```

- .csproj

```
<ItemGroup>
  <PackageReference Include="Newtonsoft.Json" VersionOverride="12.0.0"/>
</ItemGroup>
```

- Транзитивное закрепление версий
- Глобальные ссылки на пакеты

Итоги

- Скорость
- Удобство
- Эксперименты

https://devblogs.microsoft.com/dotnet/announcing-dotnet-7/

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

https://devblogs.microsoft.com/dotnet/performance_improvements_in_net_7/

https://devblogs.microsoft.com/dotnet/regular-expression-improvements-in-dotnet-7/

https://devblogs.microsoft.com/nuget/announcing-nuget-6-4-signed-central-delivered/

https://www.dotnetconf.net/

https://www.youtube.com/playlist?list=PLbxr_aGL4q3QsWawTIJPOf6sMWpP54OoH

https://github.com/ilabutin/dotnext_2022_net7whatsnew/

https://devblogs.microsoft.com/dotnet/announcing-dotnet-7/