



# C# 10 – What's new and what's next?

Christian Nagel

<https://csharp.christiannagel.com>



Professional  
**C# and .NET**  
**2021 Edition**

Christian Nagel

## Christian Nagel

- Training
  - Coaching
  - Consulting
  - Development
- 
- Microsoft MVP
  - [www.cninnovation.com](http://www.cninnovation.com)
  - [csharp.christiannagel.com](http://csharp.christiannagel.com)
  - [@christiannagel](https://twitter.com/christiannagel)



---

## Topics

- C# 10
- C# 11
- Flashbacks



# Demo: Upgrading an ASP.NET Core 5 Web API to .NET 6

---

# What you've seen

---

- Top-level statements (C# 9)
- File-scoped namespace declaration
- Global using directives
- Implicit using
- Natural types with lambda expressions
- Attributes with lambda expressions
- Minimal APIs (.NET 6)



# Flashback Records (C# 9)

Init-only accessors

Nominal or positional

Primary constructors (positional)

With expressions with clones

Equality implementations

Deconstruction (positional only)

Classes behind the scenes

Inheritance



# Flashback Structural Types

- Value types
- C# tuples (7.0)
- ref struct (7.2)
- readonly struct (7.2)
- readonly instance members (8.0)

# Struct Enhancements (C# 10)



With expressions



parameterless constructors and field  
initializers



record struct (read-write properties)



readonly record struct (init-only  
properties)



# Flashback and Future: Pattern Matching

---

Declaration pattern (7.0)

---

Const pattern (7.0)

---

Var pattern (7.0)

---

Property pattern (8.0)

---

Positional pattern (8.0)

---

Discard pattern (8.0)

---

Logical pattern (8.0)

---

Relational (8.0)

---

Declaration and type pattern (9.0)

---

Extended property pattern (10.0)

---

List and slice pattern (11.0)

---

# Static Abstract Members in Interfaces

Define static abstract members in interfaces

Use operator overloads with generics

Experimental with C# 10 and .NET 6

Interfaces `IAdditionOperators`,  
`INumber`, `IParseable`...



```
void CSharp11(object o!!)
{
    //...
}
```




```
void Dotnet6(object o)
{
    ArgumentNullException.ThrowIfNull(o);
    //...
}
```



```
void Previous(object o)
{
    if (o is null)
    {
        throw new ArgumentNullException(nameof(o));
    }
    //...
}
```

# Flashback and Future: Null checks

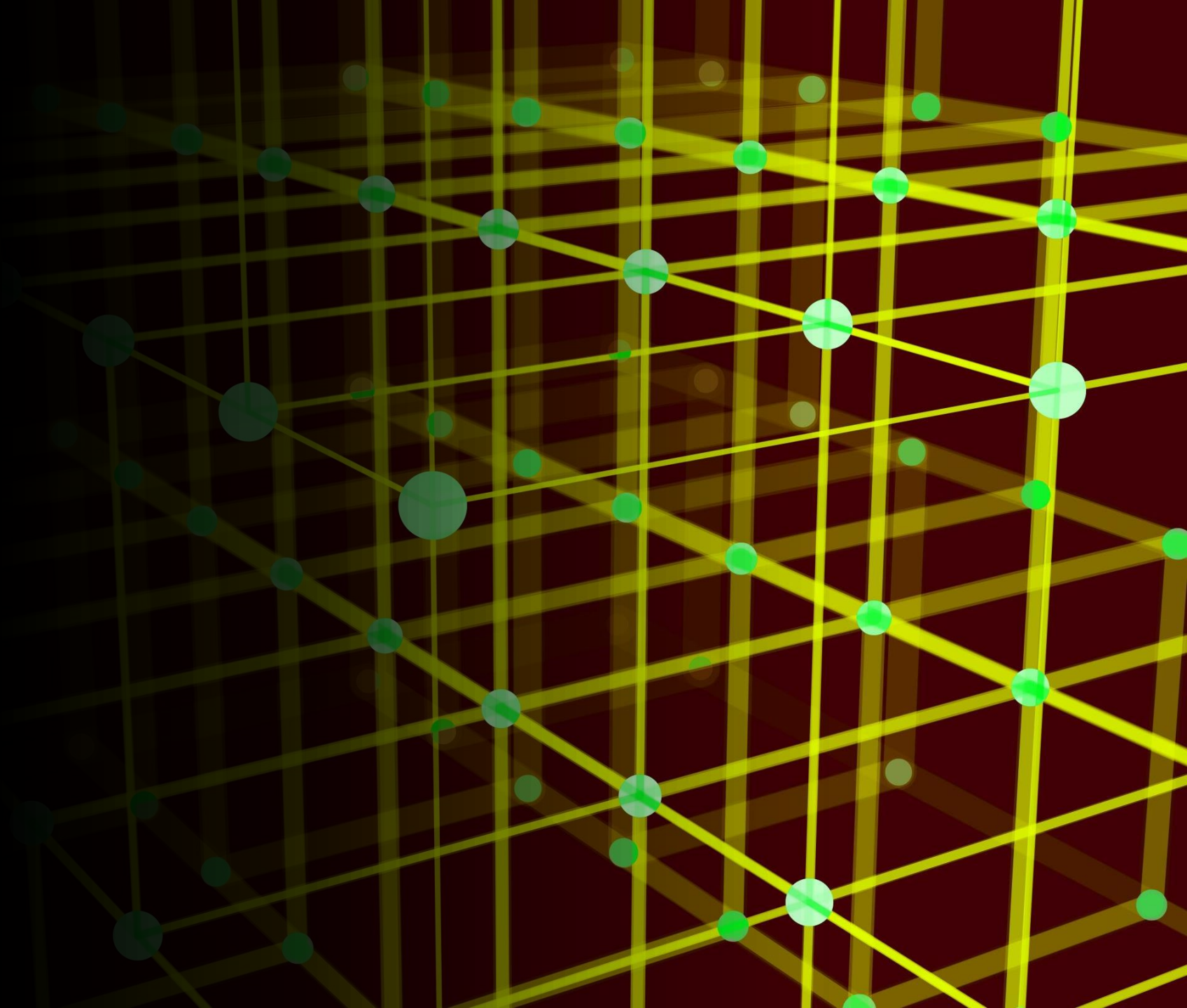
---



## Raw String Literals (C# 11)

---

- Nice indentation in the code
- Supports string interpolation
- Use for JSON, XML, Source Code...







# C# 10 Highlights

- Record structs
- Improvements of structs
  - Parameterless constructor, field initializers
  - With expressions
- Global using
- File-scoped namespace declarations
- Extended property patterns
- Improvement of lambda expressions
- CallerArgumentExpression attribute

# C# 11 Highlights

- Generic attributes (17.0 p4)
- List patterns (17.1 p2)
- Parameter-null checking (17.1 p3)
- Raw string literals (17.2)
- Cache delegates for static method group (17.2)
- Primary constructors
- Required members
- Semi auto-properties, field keyword

# Thank you!

- Questions?
- Enjoy BASTA! Frankfurt!
- Source code:  
<https://github.com/cnilearn>
- More information:  
<https://csharp.christiannagel.com>