Developing Source Generators in C#

Understanding C# Source Generators

Thomas Claudius Huber

Software Developer

@thomasclaudiush | www.thomasclaudiushuber.com

Developing Source Generators in C#

Version Check

Version Check

This course was created by using:

- C# 12
- .NET 8
- Visual Studio 2022

Version Check

This course is 100% applicable to:

- C# 10 to C# 12
- .NET 6 to .NET 8
- Visual Studio 2022

Module Outline

What is a C# source generator?

What you will learn in this course

Explore the starter project

- Look at the ToString method
- Solve the problem with reflection
- Use a C# source generator

What Is a C# Source Generator?

Why Generating Code?

Saves time and ensures consistency

Why Generating Code?

Person.cs

```
public class Person
{
    public string? FirstName { get; set; }
    public string? LastName { get; set; }

    public override string ToString()
    {
        return $"FirstName:{FirstName}; LastName: {LastName}";
    }
}
```

Why Generating Code?

Person.cs

```
public partial class Person
{
    public string? FirstName { get; set; }
    public string? LastName { get; set; }
}
```

Person.g.cs

```
public partial class Person
{
    public override string ToString()
    {
        return $"FirstName:{FirstName}; LastName:{LastName}";
    }
}
```

Approaches to Generate Code in .NET

T4 templates
(.tt files)

3rd party tools

Custom scripts

Workflow to Generate Code

Compile your project to get an assembly

Run the code that generates additional code for your project

Add the generated code to your project

Compile your project again with the generated code

C# Source Generators are a code generation approach introduced with .NET 5

Differences of C# Source Generators

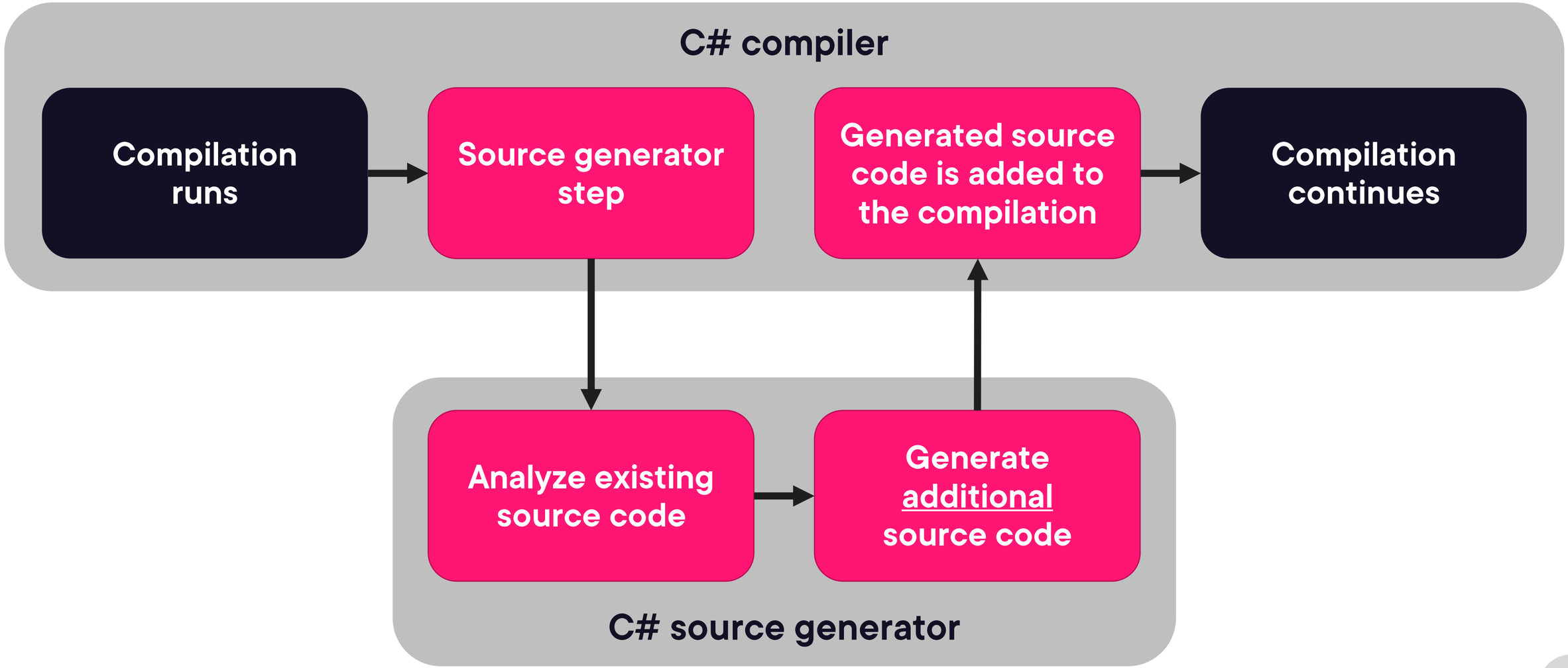
T4 templates, custom scripts etc.	vs	C# source generators
Get the assembly as input		Get the source code as input
Run after the compilation		Run as part of the compilation

C# Source Generators in the C# Compiler

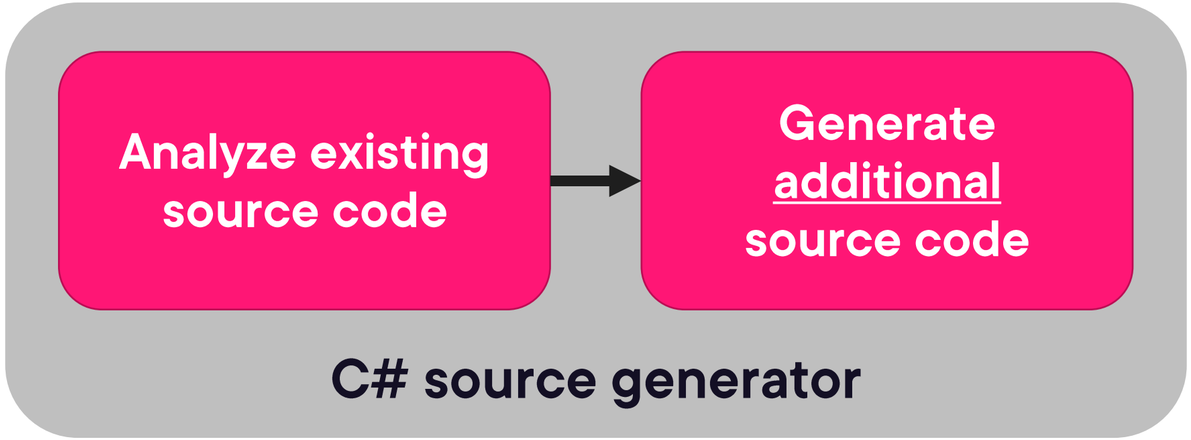
Roslyn - .NET compiler platform

https://github.com/dotnet/roslyn

C# Source Generators in the C# Compiler



C# Source Generators in the C# Compiler



.NET Standard 2.0 class library
that you reference in your project

What You Will Learn in This Course

No prior knowledge about C# source generators is required

Good C# skills are needed

You will develop a C# source generator with me for the Wired Brain Coffee company

How the C# Source Generator Will Work

Person.cs

```
public class Person
{
    public string? FirstName { get; set; }
    public string? LastName { get; set; }
}
```

How the C# Source Generator Will Work

Person.cs

```
[GenerateToString]
public partial class Person
{
    public string? FirstName { get; set; }
    public string? LastName { get; set; }
}
```

Person.g.cs (generated)

```
public partial class Person
{
    public override string ToString()
    {
        return $"FirstName:{FirstName}; LastName:{LastName}";
    }
}
```

Explore the starter project

Solve the problem with reflection

Use a C# source generator

Summary

C# source generators

- Analyze the source code of your project
- Generate additional code
- Run as part of the compilation

Explore the starter project

- Implement the ToString with reflection

Use a C# source generator

Up Next:

Setting up a C# Source Generator