

C# 2017: State of the Nation

Oliver Sturm • @olivers • oliver@oliversturm.com



Oliver Sturm

- Training Director at DevExpress
- Consultant, trainer, author, software architect and developer for over 25 years
- Microsoft C# MVP
- Contact: oliver@oliversturm.com

Agenda

- The new old stuff – what we never knew was going to be interesting
- Today's state of things – what we know and what we should know
- The future: where is C# going?

Some thoughts on dynamic

- Available since C# 4.0
- Perhaps the least accepted 'big' feature the language has ever seen?
- ASP.NET MVC is quite 'dynamic'
 - ... and so are other libraries dealing with the dynamic world of online data
- Runtime code generation mechanisms - food for thought

Something we know:

C# 6 Feature Set

- Long teased, finally finalized, stripped bare on short notice
- Favorites:
 - Auto-property assignments
 - Expression bodies
 - String interpolation
 - `using static`

Easily missed: C# REPL

- *C# Interactive* introduced in VS 2015 Update 1
- Useful tool, much loved by devs everywhere, now finally for C#
- Scripting supported (.csx files)

What else is happening with Roslyn?

- In my experience, most initial bugs have now been fixed :)
- Commercial and open source projects use it as a parser and for runtime code generation
- Best case: you don't notice you're using it

The world out there: Platforms

- You don't need Windows anymore to develop for .NET
 - Mono, but also .NET Core
- You don't need Visual Studio anymore to develop C#
 - Visual Studio Code
 - Xamarin
 - Rider

Popular Patterns

- Reactive: Rx.NET
 - Commonly used in many environments, especially JavaScript
 - Centralized, functional and parallel technique to deal with event-driven information flow
- Actors: akka.net – check out my presentation *Parallelization with Actors in C#*
 - Fault-tolerant, automatically parallel message passing service framework

The (close) future: C# 7

- Visual Studio 2017 RC builds available now
 - Out parameters
 - Pattern matching
 - Tuples
 - Local functions
 - Expression bodies for getters, setters, ctors and finalizers
 - Throw expressions
 - Literals
 - Ref returns and locals

The more distant future

- Just published: [The .NET Language Strategy](https://blogs.msdn.microsoft.com/dotnet/2017/02/01/the-net-language-strategy/)
(<https://blogs.msdn.microsoft.com/dotnet/2017/02/01/the-net-language-strategy/>)
- GitHub repo for C# design: <https://github.com/dotnet/csharp-lang>
- Mailing list: <https://lists.dot.net/mailman/listinfo/csharp-lang>

Sources

- Demo source code:
 - <https://github.com/oliversturm/cs6-demos>
 - <https://github.com/oliversturm/cs7-demos>
 - <https://github.com/oliversturm/rx-demos>
- This presentation:
 - <https://oliversturm.github.io/cs-state-of-the-nation>
 - Deprettified content in pdf format: <https://oliversturm.github.io/cs-state-of-the-nation/slidecontent.pdf>

Thank You

Please feel free to contact me about the content anytime.

oliver@oliversturm.com