

# Re: C# Language Specification – Extension Methods

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2005-09/msg02691.html>

---

- *From:* "Nicholas Paldino [.NET/C# MVP]" <mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - *Date:* Tue, 20 Sep 2005 12:30:57 -0400
- 

Marshal,

Unfortunately, the argument doesn't hold. "global" is a language concept in C#, and you are proposing changing that. Granted, it's not as ubiquitous as "this", but it is still the same thing.

The same thing applies to other keywords as well. Return doesn't always mean return. What happens when it is used in an attribute declaration, or with yield?

It's like anything else, what it means depends on the context it is used in.

The reason they chose "this" (at least as I see it) was twofold. The first was to give some sort of indication that the extension method would be applied to instances. "this" helps convey the idea of an instance pretty well. The second was to help not introduce new keywords into the language so as to reduce conflicts with pre-existing code. Your example in this case helps prevent that as well, so it's applicable here. However, I don't see "global" as a good way of saying "this should be applied to an instance".

--

– Nicholas Paldino [.NET/C# MVP]  
– mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

"Marshal" <Marshal@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message <news:4C3D27F0-FB4C-42E2-BF8D-A204614B20F2@xxxxxxxxxxxxxxxxxxxx>  
>I think one of the major advancements on C# will be the growing support for  
> Extension Methods and related concepts...  
>  
> The current syntax involves a static class with static members using a  
> warping of the "this" concept applied to function parameters. Surely we'll  
> see more twisting and warping of language semantics as this technology  
> evolves...  
>  
> Rather than warp language symantics... It would be more apparently proper

## Re: C# Language Specification – Extension Methods

> to  
> allow something like:  
>  
> (\*\* Proposed syntax:  
> public class global::System.String {  
> public static string Encrypt(object key) { ... }  
> }  
> \*\*)  
>  
> Something generally synonymous with partial types... saying, hey look, you  
> can even partialize an existing type. People will ooh and ahh over it... I  
> can hear you doing it now as you prepare your rebuttle. Initially only  
> static  
> members would be allowed, but you could then provide support for more  
> things  
> without ever warping language concepts.  
>  
> Currently, we have to teach... "this" means "this", unless it's on a  
> parameter in an extension class, in which case it means "that".  
>  
>

- 
- Prev by Date: [\*Custom Thread Pool \(by Mr. Jon Skeet\) Enhancement\*](#)
  - Next by Date: [\*Re: Database Connection questions.\*](#)
  - Previous by thread: [\*Custom Thread Pool \(by Mr. Jon Skeet\) Enhancement\*](#)
  - Next by thread: [\*Re: C# Language Specification – Extension Methods\*](#)
  - Index(es):
    - ◆ [\*Date\*](#)
    - ◆ [\*Thread\*](#)