Public Annotations

Code is data that communicates your intent. If you have no special relationship with your compiler, you don’t need any special data to communicate additional intent.

Once you’re in an open compiler world, you may need to communicate with your compiler. This features has been called “design time attributes” and “annotations.” I’m adding this feature to RoslynDom and calling it “PublicAnnotations”

# Why not just use attributes?

Attributes do not work very well to communicate with the compiler for at least these reasons:

* Can’t tell what’s available at runtime
* If design attributes are visible at runtime, they become a contract
* Can result in build dependency
  + If one player removes attributes it’s done with to avoid runtime contracts
* Must follow attribute syntax
* Only constants allowed
  + No lambda expressions
  + No generic types
  + No expressions
* Can’t be placed in all desired locations
  + Not on namespaces, files, or in random locations inside or outside methods

I think the first is actually the biggest issue. I think it’s important to differentiate communications with the compiler pipeline, including design time with the Visual Studio/Roslyn linkage, and runtime attributes. But even if you disagree with that, attributes simply don’t work because of limitations in attribute content and attribute location.

# The Syntax

Eventually there will be enough examples of public annotations that an obvious syntax can be included in the languages. I’m not willing to wait as I need public annotations right now, like today.

The syntax has to reside inside a comment. That’s the only way to solve the content and location limitations of attributes without changing the compiler.

The syntax should be clearly differentiated from all other lines of code to allow easy recognition by human, parser/loader and later IDE colorization. The syntax should also be easily found via RegEx to allow updates when language support arrives.

RoslynDom supports the following, with any desired whitespace within the line.

//[[ NameOfAnnotation( stuff) ]]

This requires the annotation appear on a single line. I will support multi-line comments only if I can make colorization work. Preliminary work has convinced me that colorization is absolutely essential.

# Implementation

RoslynDom currently holds the public annotation internally. It’s available via the PublicAnnotation property on all IDom classes.

When RoslynDom supports tree changes via output of a new tree, public annotations will appear on the SyntaxNode as annotations in a property bag with the key “Public.”

# The Naming

[Skip this part if you just care about the feature]

The phrase “design time attributes” implies an implementation as a special case of an attribute.

The phrase “annotations” is used in Java for a similar type of design time code marking, but the .NET Compiler Platform/Roslyn already has a feature called annotations and it’s intended as a private way for components to temporarily drop a piece of information onto a SyntaxNode.