Updates to RoslynDom 1.0.14 Alpha

# Added Expression parser to RDomCSharp

# Added format methods to RDomCSharp

# Added AddUsingDirective(s) and AddNamespace(s) to base Stem Container

A total of 8 methods added to IRoot and INamespace. These include name and typed parameters.

# Added Default as a literal type, avoiding a crash when used in parameters

# Began adding friendly constructors to RoslynDom classes

# Added support for Events

Not yet support for accessors

# Enums supported in attribute values

# Issue with whitespace in qualified declarations fixed

This may have occurred other places as well.

# Generic field and event types

*Not fixed*

# Implemented Interfaces

*Not fixed*

Updates to RoslynDom 1.0.13 Alpha

# RoslynDom files roundtrip

Files in the RoslynDom assembly can be roundtripped. Additional testing and fixes were in support of this.

# Separate scope on Get/Set accessors

Different scopes are now output correctly

# Whitespace

Continuing work on whitespace.

Among other things, break is no longer left aligned.

# Generics on structs

Fixing this also led to better code reuse across methods, interfaces, classes and structs.

# Zero LanguageElement no longer Comment

Ooops. The LanguageElement with the value zero was Comment, and NotApplicable had a number. This has been fixed. Normally I wouldn’t bother to report a change in an enum value, but if you were comparing with comment to indicate an unused item, please alter your code.

Updates to RoslynDom 1.0.12 Alpha

# Improved Code Coverage

Many small fixes found during testing.

I realize that code coverage means very little if you encounter a bug, please send me the code that did not roundtrip correctly. That’s my current push: lambdas, delegates, events, destructors and async are all known not to be working yet.

Current stats are 609 unit passing unit tests, 18 inconclusive tests marking work that’s understood and not done. There are 10,158 covered blocks, 752 uncovered blocks (93.11%). Part of the remaining is things known to be in place and incomplete (destructors). 3,641 lines of test code/4959 lib LOC. Not that I think any of that matters, except to show that the intent is solid and complete testing – just not there yet.

FxCop has slipped with currently 25 issues.

# Aliasing is preserved

If a type uses an alias (string vs String), then an alias is used when the syntax tree is rebuilt.

# ReferencedType resolves with Using Directives on output

A referenced type (like System.String) is now be shortened using the current Using Directives (String if System is in a using directive).

# Default parameter values lost on output

While the flag stating that optional parameters were optional was set correctly, the default values were not output – fixed.

# Declarations with no initializer produce empty entry in descendants

The descendant list of a method includes a null after any declaration that does not include an initializer. – Fixed

# Return void produces empty entry in descendants

The descendant list of a method includes a null after a void return. – Fixed

# Finally appeared before statements in descendants of try statement

Fixed

# Catch block doesn’t include exception declaration in descendant list

Fixed

# Checked returns statements before resource declaration in children and descendants

FIxed

# Copying types with constructors failed on BuildSyntax

The constructor used the parent to determine its name and the parent wasn’t reestablished on copy. If the constructor is orphaned, the name “unknown\_name” will be used to avoid an error: fixed

Updates to RoslynDom 1.0.11 Alpha

# Fixes to solve FxCop issues

Dropped fro 160 to 11 that I will continue to sort through

# Bug Fixes

This release contained a huge number of bugs. I’ll document how this happened in a blog post, but for now I want to get this better version up as quickly as possible.

## Modifiers on fields and methods

A bunch of mofiers were not being output correctly. The ones I know about: Fixed

## Type parameters on methods, classes and interfaces

In many scenarios, these were not output correctly. The ones I know about: Fixed

## Base classes and interfaces

In many scenarios, these were not output correctly. The ones I know about: Fixed

## Check, Using, Lock statements

In many scenarios, these were not output correctly. The ones I know about: Fixed

## Constructors

In all scenarios, these were not output correctly. The ones I know about: Fixed

# Refactored AttributeTests

These changes are documented in this blog post

# Refactored SameIntent Tests

This additional testing is what uncovered the bugs fixed in this update. These are now broad tests, not just SameIntent – name may change in the future.

# Bug that caused crash on output of typeof attribute values

Fixed