

News



Documentation is being moved to github wiki

Rodrigo B. de Oliveira posted on Oct 19, 2012

Thanks to the effort of a few brave contributors the documentation is being moved to [the github wiki](#) where we can have more control over it and also make it easier for other contributors to help.

Be sure to [check it out!](#)

3 Comments



Boo 0.9.4 is here

Rodrigo B. de Oliveira posted on Jan 21, 2011

It's been a long year since the last release but the wait is finally over: Boo 0.9.4 is here!

It's a release packed full of features, improvements and bug fixes:

- **~4x shorter compilation times**
- support for partial interfaces (<http://goo.gl/k5f16>) and enums (<http://goo.gl/LgiLC>)
- support for typed collection initializers (<http://goo.gl/vNH1R> and <http://goo.gl/pfd9x>)
- **much improved Linq extension method support** including type inference for closures (<https://gist.github.com/618168>)
 - System.Core.dll is now referenced by default, import System.Linq.Enumerable and Linq away
- 'new' keyword for explicit shadowing of inherited members (<http://goo.gl/kQKrw>)
- initial support for .Net 4.0
- **loads of metaprogramming improvements**
 - macro application over type members (<http://goo.gl/0GXaE>)
 - support for macros yielding partial types (<http://goo.gl/Dzmq6>)
 - much improved Boo.Lang.PatternMatching
 - much improved quasiquotation and splicing support (<http://goo.gl/WuOVd> - <http://goo.gl/NbOAU> - ...)
 - meta methods can now expand to code containing macros (<http://goo.gl/Flcpc>)
 - better support for block quasiquotation ("block:" trick no longer needed when quasiquoting a complex sequence of statements) (<http://goo.gl/2lbSb>)
- type inference for empty arrays will take their usage into account (<http://goo.gl/FJMST>)
- cast function has been deprecated by the introduction of the new cast operator (e cast type)
- ifdef macro for conditional compilation (<http://goo.gl/RJgZM>)
- simple identifier interpolation: "Hello, \$name!"
- for consistency interpolation now uses () instead of {} for complex expressions ("2+2: \$(2 + 2)" instead of "2+2: \${2 + 2}")
- automatic stubbing of inherited abstract generic methods (<http://goo.gl/gn1WV>)
- proper support for partial classes containing nested types (<http://goo.gl/ixmt0>)
- fixed matrix builtin on .net (broken by a System.Reflection.Emit bug)
- better report of ambiguous generic type reference errors
- fix for shortcircuited expressions involving implicit bool conversion operators
- and many many more...

Kudos to all that contributed to this release throughout the year specially: , Ryan Boggs, Dmitry Malyshev, Maksym Trushyn, Daniel Grunwald, Cedric Vivier, Spruce Weber, socrates877, rektide, George Dernovoy, Benjamin Reed and JB Evain.

As usual the packages can be download from:

<http://dist.codehaus.org/boo/distributions/?C=M;O=D>

A new MonoDevelop addin is also available and can be installed from the Community Add-in Repository:

<http://addins.monodevelop.com/>

Enjoy!



Boo 0.9.3 is here!

Rodrigo B. de Oliveira posted on Dec 28, 2009

Boo 0.9.3 fixes a lot of bugs and extends the match macro with support for regular expressions with variable binding.

Thanks to Martinho Fernandes it's now possible to write code like this::

Kudos also to Andre Van Der Merwe, Cedric Vivier, Gotz Washck and Richard Hubers for contributing to this release!

Complete change log is [here](#) and as usual you can download it from [here](#).

Happy 2010!



Boo 0.9.2 is here!

Cedric Vivier posted on Sep 01, 2009

You read it right and it was about time!

Boo 0.9.2 is mostly a maintenance release, yet it introduces a few new features worth highlighting :

- Unsafe code support including pointer manipulation for richer interoperability with native code [\[BOO-1202\]](#)

- Generic array/matrix constructors [\[BOO-1203\]](#)

- Literal regular expression options [\[BOO-1237\]](#)

- Boo now runs on medium-trust environments [\[BOO-1215\]](#) (such as Mosso ASP.NET platform)

On top of the usual bunch of bug fixes (see the [ChangeLog](#)), this release significantly improves generated IL, resulting in up to 40% leaner assemblies among other performance improvements.

Contributed to this release: [Cedric Vivier](#), Daniel Grunwald, Dmitry Malyshev, Greg Nagel, Joao Braganca, Martinho Fernandes, Paul Lange and [Rodrigo B. De Oliveira](#).

[Download it](#) and have fun!

[1 Comment](#)



Boo 0.9.1 is here!

Cedric Vivier posted on Mar 18, 2009

Just after two months of the huge 0.9 release, here is Boo 0.9.1 - bringing more new features and bug fixes.

Highlights of this release are:

- Macro definition arguments [\[BOO-1146\]](#) - macro definitions can define typed arguments as with any method definition.

- Nested macros extensions [\[BOO-1140\]](#) - nested macros no longer have to be defined within their parent macro block.

- Omitted expression for member references [BOO-1150] - ``.foo'` is now equivalent to ``self.foo'` by default. This behavior can easily be changed by a macro or compiler step.

- Volatile fields [BOO-806]

- `TypeSystem` refactoring - brings cleaner API and faster compilation (-30% time)

Take note that from now on strong versioning is used on Boo releases, this release assemblies are versioned ``2.0.9.1'`.

Contributors to this release: [Cedric Vivier](#), [Daniel Grunwald](#), [JB Evain](#), [Rodrigo B. De Oliveira](#).
Read the [changelog](#) for the complete list of improvements.

[Download](#) it now and have fun!



Boo 0.9 is here!

Cedric Vivier posted on Jan 27, 2009

It's been a long time but the biggest release ever of Boo is right here now!

Huge improvements all over the board as you can read in the full [changelog](#), its chief weapons are:

- Generator macros [BOO-1077] - macros are no longer limited to returning a single statement or block and instead are able to **yield** an indefinite number of nodes:
- Nestable macros [BOO-1120] - macro definitions can be nested to allow for context sensitive keywords
- Type member macros [BOO-415] - macros can be used in class definition bodies
- Pattern matching [BOO-1106] - simple but powerful object pattern matching with the **match/case/otherwise** macros
- Support for generic extension methods [BOO-937] - LINQ style extension methods
- Generic parameter constraints [BOO-935] and Generic parameter type inference [BOO-1102]
- Strict mode [BOO-1115] - strict mode changes a few compiler rules: default visibility for members is private, method parameter types and return types must be explicitly declared, among other things
- Asymmetric property accessor visibility [BOO-1094]
- String interpolation formatting (eg. `"0x${n:x4}"`) [BOO-1006]
- Support for SilverLight profile [BOO-1117] - and Vladimir Lazunin kicked it off with a Tetris [example](#).

You can read examples on these 0.9 new features on Rodrigo's [blog](#).

This release is brought to you by Avishay Lavie, Cedric Vivier, Daniel Grunwald, Marcus Griep and Rodrigo B. De Oliveira.

[Download](#) it now and have fun!

Join the [mailing-list](#) for questions and latest updates about Boo development.



Boo 0.8.2 is out!

Cedric Vivier posted on May 20, 2008

Yeah, it's that time again!

As usual lots and lots of improvements in this new release including :

- (almost) complete nullable type support
- **shorthands** for nullable types (T?) and enumerables (T* instead of IEnumerable[of T])
- improved booish behavior with nicer colors (and it should work inside emacs now 😊)
- 'else' block for 'for' and 'while' loops
- fixes and improvement related to generic methods ([overloads](#) and [interface declarations](#))
- and lots of other fixes here and there

Complete list of changes available [here](#).

Contributors to this release :

Avishay Lavie, Cedric Vivier, Marcus Griep, [Rodrigo B. De Oliveira](#)

Have fun!

1 Comment



Boo 0.8.1 released !

[Cedric Vivier](#) posted on Feb 08, 2008

This release includes lots of improvements including :

- a simpler way for writing macros
- support for [nested functions](#)
- generic methods overloading works
- support for CLR 3.5 extension methods (moreover boo extension methods)
- compile-time conditionals through [ConditionalAttribute](#) and the new -define SYMBOL booc option
- [AttributeUsageAttribute](#) is now supported and enforced
- a better interactive interpreter (previously known as [booish2](#))
- warnings about unused private members, unused namespaces, unreachable code
- new error messages, including suggestions for misspelled members or types
- exception filters, exception fault handlers
- for loop IDisposable.Dispose integration

Contributors to this release :

Avishay Lavie, Bill Pierce, Cédric Vivier, Daniel Grunwald, Marcus Griep, and last but not least [Rodrigo B. De Oliveira](#).



Specter 0.8 released

[Cedric Vivier](#) posted on Jan 08, 2008

[Specter](#) is back. It is a behavior-driven development framework for the CLI, powered by Boo meta-programming capabilities.

This release makes it work with recent Boo releases and introduces new features including standalone spec runners and a new 'subject' macro.

Full changelog and links [here](#).



Ayende on Boo

[Cedric Vivier](#) posted on Dec 18, 2007

There is a nice [post](#) from Ayende Rahien about why he loves Boo on his [blog](#).



Boo at Mono Summit 2007

[Cedric Vivier](#) posted on Dec 07, 2007

Rodrigo made a talk on Boo at Mono Summit 2007 in Madrid which got [very good feedback](#). He shared his impressions on his [blog](#) along with the [slides](#) and some [examples](#) of the new meta-programming features newly available on Boo 0.8

Also, it is worthy to note Rodrigo made reference to the new [Specter](#) object-behaviour specification framework, another great project powered by Boo.



Boo 0.80

Bryan Kelly posted on Oct 27, 2007

This release includes bug fixes, performance improvements and better meta-programming capabilities (1).

Special thanks to Marcus Griep, Nick Fortune and Matt McElheny!

What? - <http://boo.codehaus.org/>

Download - <http://boo.codehaus.org/Download>

Official irc channel - <irc://irc.codehaus.org/boo>

Full change log [here](#) (2).

Have fun!

(1) see the 'match' and 'data' macros in the boo-extensions (3) project for examples

(2) [http://jira.codehaus.org/browse/BOO?report=com.atlassian.jira.plugin](http://jira.codehaus.org/browse/BOO?report=com.atlassian.jira.plugin....)

(3) <http://code.google.com/p/boo-extensions/>



Boo 0.7.9

Bryan Kelly posted on Sep 01, 2007

This release includes bug fixes, improves on generic support and introduces a few metaprogramming facilities (still on early stage).

Many thanks to the growing boo community!

Check out the [Release Notes](#) and grab it from [here](#)!



Boo 0.7.8 is Here!

Bryan Kelly posted on Jun 23, 2007

With many thanks to the people who contributed for this release: Andrew Davey, Avishay Lavie, Cedric Vivier, Chris Prinos, Doug Holton, Jim Lewis and Max Bolingbroke.

What? - <http://boo.codehaus.org/>

Download - <http://boo.codehaus.org/Download>

Official irc channel - <irc://irc.codehaus.org/boo>

Highlights for this release include [dramatic improvements to dynamic dispatching performance](#), a friendlier DSL syntax and of course bug fixes. This is also the **last release to support .net 1.1**.

Full change log [here](#).

Yeah!



Test News

Bryan Kelly posted on Jun 23, 2007

Why have news items ceased to post?

Syndicate this site via [RSS](#).