

## Re: VB vs C#

**Source:** <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2004-04/1146.html>

---

**From:** Jon Skeet [C# MVP] ([skeet\\_at\\_pobox.com](mailto:skeet_at_pobox.com))

**Date:** 04/13/04

Date: Tue, 13 Apr 2004 12:33:21 +0100

Cor Ligthert <[notfirstname@planet.nl](mailto:notfirstname@planet.nl)> wrote:

- > > A C# developer can use the *Microsoft.VisualBasic* namespace if they
- > > really want to. I don't remember seeing any examples where it's faster
- > > than can be achieved by not using it, however.
- >
- > We did some test now already again a some long time ago in the *language.vb*
- > group (I did although with comments and test from the others as Jay B.).
- >
- > The *Find* from a string is twice as fast as the *indexOf* method with a string.

I'd be interested to see that thread/benchmark. Do you remember what it was called?

It's certainly possible that *String.IndexOf* hasn't been well-optimised, but I suspect the optimisation could be done just as well in a C# method if necessary.

- > I was also suprised because I have no real VB classic background, so I
- > never use that (still not), as I often told you I think that the 1 as start
- > of an index is the best however, I am used to the zero to start an index.

Not sure how that's relevant here...

- > The *indexOf* however outclassed the *Find* when it is with a real defined char
- > at least 10 times in performance (the char *Find* does not exist).
- >
- > Not so long ago I did also test with the *Replace* and was expecting the same
- > result. However that *MB find* had the same speed as the *Regex*. The
- > *string.replace* was amazing enough the fastest with the *stringbuilder.replace*
- > (that is the one I like) on a very good second place (only a slight
- > difference). The test are done by more persons on more places in the world
- > on different computers and the methode is checked also by those persons.

Why is it amazing that *String.Replace* is fast?

- > The *MB replace* and the *Regex* was 10 to 100 times slower dependable on the
- > situation 1000 times a short string or 10 times a long string than the
- > *string.replace*. (It was a very simple *Regex*).

>

> *I hope this gives you some idea's*

I'm not at all surprised that Regex was fairly slow. People always seem to be recommending that one should use a regex when it's just as simple (and more readable) to use a few string operations.

Regular expressions are incredibly powerful, but most of the time that power isn't needed.

--

Jon Skeet - <skeet@pobox.com>

<http://www.pobox.com/~skeet>

If replying to the group, please do not mail me too