

4 reviews of: C# and .NET Introduction – History, Current

---

### Opposition 1:

Review of:  
C# and .NET Introduction – History, Current

I think you have done a real good job with your report, I think it was easy to read and I really like the way you add your own code examples. It gives the reader an insight to what the code may look like. I also feel you have a nice layout on your report. What I mean about layout is the way you describe something in text and then follow up with the code bit to show what you mean. I also feel that the information you supply us as readers with is typed in such a manner that it is easy to understand and informative.

What I don't feel like you need to tell us all the time is from where you have gotten the desired paragraph or section. You tend to start each paragraph or section with "in this section", which I feel you do not need to do as you show that by using your references. I like the addition of the pictures you have included, but they are too small in my opinion, it is near impossible to read what is typed on them. Either you should make them bigger in the text or add them as appendix. I have found some small errors in the report. I have also seen that you have put some personal reference in the report and that should have been left until the conclusion.

### Questions

1. LINQ, is what is that exactly?
2. Who claims that C# works faster than Java?
3. In the section Generics, what do you mean?

---

### Opposition 2:

#### **Review of Thomas Bragesjö's Rapport**

By Niklas Norman

This report is generally very good and explains everything thoroughly. It almost makes me want to learn C#.

The main things that I think you should change is how you mention your references with according to.... on page 5 is an example

*VB.NET does not have static classes but Modules can be used instead according to Wikipedias comparison of the two languages [10].*

I think that you should instead reference in this manner.

*VB.NET does not have static classes but Modules can be used instead [10]*

Another thing you do in your report is mentioning yourself a lot. Instead of saying "sample code made by myself", you should write something in the line of "here is a sample code to show how this and that works".

Below follows a list of what I would consider reviewing.

Page 6. 1.4.3. keep personal opinion until the conclusion.

Page 8. 1.6 than, not then.

Page 9. Section 2. in my experience, you should not to mention yourself in a paper, 2.0.6, just use the [3] instead of “according to ...”

Page 10 height not hight. Also page 11.

Page 13. according to...

Page 14, according to...  
where I removed, try not to mention yourself.

Page 16, created by myself...

Page 17, “Wikipedia writes [8] that Microsoft are currently working on adding things from functional languages like Lambda expressions”. Should be something like “Microsoft is currently working on adding things from functional languages like Lambda expressions [8]”.

Page 18. There are many people who claim that C# programs executes faster then Java programs (especially graphical) but I do not know if it is true or not.

Many people? Who? Anyone I know?

See [http://en.wikipedia.org/wiki/Wikipedia:Avoid\\_weasel\\_words](http://en.wikipedia.org/wiki/Wikipedia:Avoid_weasel_words) to see what I mean with this. You also mention yourself.

Page 19, mentioning yourself a lot.

Page 24, according to...

---

### Opposition 3:

REVIEW: C# and .NET Introduction - History, Current and Trends  
Mikael Asberg

#### Review

The 'Abstract' section gives a well explanation of the reports topic, indeed it is a overview of the .NET platform and C#. Interesting part where the author explains investigations of the creation of .NET and the birth of C#. However, the author does not explain much about the .NET platform and framework. I am not familiar with .NET and would have liked if the author could have explained what .NET is, more detailed. But of course, many aspects of it are well presented.

But in general, it seems like the subject is to large considering the size that this paper should be (15-20 pages). I think the report get's to thin becuase of the big topic. If the topic had been smaller, perhaps only .NET, then the author could have gone deeper

into the subject, making it more interesting. Or perhaps only focusing on C#, or some aspect of it.

But of course, I recognize that C# and .NET are tightly connected, but it's a bit boring having so much about C#, and so little explaining of .NET. The part about C# is sometimes explained at a too low level, I think the reader skips the part about how loops look like in C#. The C# part is very much about simple constructs, which is unnecessary. The part about OO considering C# is more like it, a better abstraction. Section 3 about design patterns is a bit of an outsider, it does not belong to the topic really. Showing what you can do with C# is out of the scope, I think section 3 makes it's own report. So skipping this design pattern section would make the report more uniform. A remark though is that it is very nice written considering his own examples. But there is too much complex code snippets, and too little explaining of them. As a reader, it gets to exhausting. I think the report has a lot of focus on C#, maybe leaving the .NET part out of the report name could be good consideration perhaps. And let .NET be a part of or a subsection of C#, perhaps the opposite. But in the end, the topic is too wide. Most importantly, skip section 3, or make it a separate paper. I must give the author a medal for doing such a big paper all by himself, most of it is very interesting and nice written. But the next time, narrow your subject. It often happens that one picks a too large topic thinking, will it be enough? And it ends up with a far too big paper! But as with many other things, training gives skills. The author is very precise with naming his sources for each chapter, which is of course very good.

It is easy to 'check' the report against it's references.

The conclusion (Section 5) contains a too detailed overall description of .NET and C#. This section lacks the authors own thoughts and conclusions regarding the subject. The author should express his thoughts and have a small discussion, I want to read what he has to say about .NET and C#. For example, what are his experiences of using .NET and C#? Another important thing to note, is that many sections are far too small, and many of them are unnecessary. It feels like the author puts too many chapters in this paper, skip these little sections or develop the discussion in them further. But do not have small sections with two lines (chapter 2.0.13).

Misspelling on the header for chapter 2.0.6: 'Strucs', maybe he means 'Structs' Many code examples should have more explaining, for example the inheritance example on page 8. And have titles on the pictures and the code snippets. Many (perhaps all) occurrences of 'inherits' or 'inheritance' are misspelled, considering the context. There are a lot of sentences that are too long, many of them should be separated with a comma.

#### SUGGESTED IMPROVEMENTS:

- Delete or develop a further discussion in the tiny chapters that are scattered throughout the paper.
- Express the conclusion (Section 5) with your own thoughts and ideas.
- Many sentences need a comma (or perhaps need to be divided into several sentences), for example chapter 1.3, the first sentence:  
'The advantage of using the Common Language Runtime (CLR) rather than machine code is that there is a clear framework and no incompatible .dll file trouble and

thereby avoiding 'DL Hell' troubles as they call it in 'Avancerad Pocket Visual Basic NET'[23].'

Questions: Tomas Bragesjo

- In chapter 1.3, what exactly is CLR? A interpretator perhaps. Why does CLR and JIT allow platform optimization and runtime type safety?
- In chapter 1.4.1: 'Visual Basic 6 and previous versions were object based. This meant that code was copied and pasted automatically. VB.NET is instead object oriented.' Explain this part please.
- In chapter 2.0.4, by the way, the header 'Inherits' should be Inheritance. Class Test inherits function somefunction(): int from two classes, could you explain this and base (y).
- In chapter 2.0.10, Integers treated like they are objects?
- In chapter 2.4: 'Wikipedia writes that C also has support for concurrent computing meaning that several tasks can be executed on at the exactly same time.[9]'. So C# does not support threading?

---

#### Opposition 4:

Review of

“C# and .NET Introduction - History, Current and Trends”

by Amir Shariat

Review

Nice report, I have used C# and the .NET framework quite often so I know how big topic it is and it may be hard to try and explain everything, but you have done a great job.

It is very good that you show code examples, it makes it easier to understand and follow, but try to keep the same indentation style and try explain your code more either with comments or with line numbers that you can refer to. It is also nice to see that you are very accurate with your references, even with your code examples.

When I read the report I see that you have covered all sections pretty good, except for a few, example 2.0.13 “Call back and delegation”, where you just refer to a book. I think that you should either remove that section or try to explain it more, perhaps with a code example just like some other sections. Your topic says “C# and .Net introduction ... ” so chapter three (design patterns) feels a bit overkill, it has nothing to do with your topic and it just makes the report too big. My suggestion is that you remove chapter three and instead try write a little more about the small sections that you have, for example 2.0.13. You never mention what you think about C# and .NET, what your thoughts about it is, perhaps you can mention it in the conclusion. I can't come up with anything more to say other than I think that this was a really nice report, and I can see that you have put alot of work into it considering that you wrote it all by yourself. Good job!

Questions

1. Can you explain a little more about callbacks and delegations in section 2.0.13? How are they used in C#?
2. You say that VB 6 was object based and VB.NET is object oriented, can you explain what the difference is?
3. You use namespaces, can you please explain what that is?
4. Section 2.3 “According to Wikipedias comparison of the languages [11], in Java one must implement the observer pattern manually.” - can you explain that further?
5. Section 1.4.2 “VB.NET does not have static classes but Modules can be used instead” - can you explain what Modules is?