

Productivity gains of using CASE tools for development

Asked 16 years, 4 months ago Modified 6 years, 9 months ago

Viewed 571 times

▲
2
▼

I was using a CASE called [MAGIC](#) for a system I'm developing, I've never used this kind of tool before and at first sight I liked, a month later I had a lot of the application generated, I felt very productive and ... I would say ... satisfied.



In some way a felt uncomfortable, cause, there is no code and everything I was used to, but in the other hand I could speed up my developing. The fact is that eventually I returned to use C# because I find it more flexible to develop, I can make unit testing, use CVS, I have access to more resources and basically I had "all the control". I felt that this tool didn't give me confidence and I thought that later in the project I could not manage it due to its forced established rules of development. And also a lot of things like sending emails, using my own controls, and other things had their complication, it seemed that at some point it was not going to be as easy as initially I thought and as initially the product claims. This reminds me a very nice article called "[No Silver Bullet](#)".

This CASE had its advantages but on the other hand it doesn't have resources you can consult and actually the

license and certification are very expensive. For me another dissapointing thing is that because of its simplistic approach for development I felt scared on first hand cause of my unexperience on these kind of tools and second cause I thought that if I continued using it maybe it would have turned to be a complex monster that I could not manage later in the project.

I think it's good to use these kind of solutions to speed up things but I wonder, why aren't these programs as popular as VS.Net, J2EE, Ruby, Python, etc. if they claim to enhance productivity better than the tools I've pointed?

case-tools

Share

Improve this question

Follow

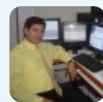
edited Mar 5, 2018 at 22:17



slayernoah

4,472 ● 12 ● 44 ● 76

asked Aug 17, 2008 at 7:08



Nelson Miranda

5,554 ● 5 ● 35 ● 58

4 Answers

Sorted by:

Highest score (default)



1

We use a CASE tool at my current company for code generation and we are trying to move away from it.



The benefits that it brings - a graphical representation of the code making components 'easier' to pick up for new developers - are outweighed by the disadvantages in my opinion.



Those main disadvantages are:

1. We cannot do automatic merges, making it close to impossible for parallel development on one component.
2. Developers get dependant on the tool and 'forget' how to handcode.

Share Improve this answer

answered Aug 17, 2008 at 15:21

Follow



Klelky

404 ● 1 ● 7 ● 15



0



Just a couple questions for you:

How much productivity do you gain compared to the control that you use? How testable and reliant is the code you create? How well can you implement a new pattern into your design?



I can't imagine that there is a CASE out there that I could write a test first and then use a CASE to generate the code I need. I'd rather stick to resharper which can easily do my mundane tasks and retain full control of my code.

Share Improve this answer

answered Aug 17, 2008 at 15:32

Follow



mendicant

27 ● 4



0



The project I'm on originally went w/ the Oracle Development Suite to put together a web application.

Over time (5+ years), customer requirements became more complex than originally anticipated, and the screens were not easily maintainable. So, the team informally decided to start doing custom (hand coded) screens in web PL/SQL, instead of generating them using the Oracle Development Suite CASE tools (Oracle Designer).

The Oracle Report Builder component of the Development Suite is still being used by the team, as it seems to "get the job done" in a timely fashion. In general, the developers using the Report Builder tool are not very comfortable coding.

In this case, it seems that the productivity aspect of such CASE tools is heavily dependent on customer requirements and developer skill sets/training/background.

Share Improve this answer

answered Aug 17, 2008 at 22:17

Follow



user290



Unfortunaly the [Magic](#) tool doesn't generates code and also it can't implement a design pattern. I don't have

0



control over the code cause as i stated before it doesn't have code to modify. Te bottom line is that it can speed up productivity in some way but it has the impossibility to user CVS, patterns also and I can't control all the details.

I agree with gary when he says "it seems that the productivity aspect of such CASE tools is heavily dependent on customer requirements and developer skill sets/training/background" but also I can't agree more with Klelky;

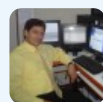
Those main disadvantages are: 1. We cannot do automatic merges, making it close to impossible for parallel development on one component. 2.Developers get dependant on the tool and 'forget' how to handcode.

Thanks

Share Improve this answer

Follow

answered Aug 20, 2008 at 13:25



[Nelson Miranda](#)

5,554 ● 5 ● 35 ● 58