

Re: Checking for Modification to a Set of Files

Source: <http://coding.derkeiler.com/Archive/General/comp.programming/2006-03/msg00793.html>

- *From:* Randy Howard <randyhoward@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 27 Mar 2006 07:29:22 GMT
-

Ben Bacarisse wrote
(in article <pan.2006.03.26.04.35.14.277042@xxxxxxxxxx>):

On Sat, 25 Mar 2006 19:47:05 +0000, Randy Howard wrote:

Ben Bacarisse wrote
(in article <pan.2006.03.25.18.36.08.510189@xxxxxxxxxx>):

I was not suggesting you were partisan about platforms. You seemed to be, like me, partisan about camelCasedAPIsFromRedmond and this was obscuring the fact that .NET applications are not Windows-only.

I've yet to encounter or hear anyone claim that they actually use .NET on anything other than Windows for production use.

Hmm... I am not sure what answer you are reserving by using the word "production". The mono team lists various uses that look, to me, like production applications, but I may have missed your point.

What I mean is I have not heard of a single application running in the wild for anyone other than the developers using the .NET platform for anything but a Windows system. When there is a non-zero collection of applications for mono that run on multiple platforms, then it will be more than a pipe dream.

<snip>

I've seen zero effort out of MS to push .NET as a cross-platform standard so far.

Re: Checking for Modification to a Set of Files

No, I am sure they have not. They merely want it (technically CLI and C#) to be "a standard" and have picked the easiest route (ECMA). I did not say they would push it as cross-platform or that they would not try to subvert it to their own ends. I wanted to move the argument away from "it is not standard" to /why/ it is not suitable for cross-platform development (if, indeed, it is not).

Being a pretend-standard doesn't make it any less of an issue. It might in fact be suitable for cross-platform development, however nobody seems to be rushing towards it as such, unless there is a lot of poorly advertised development work going on for mono apps. In fact, Microsoft themselves could do something *VERY* interesting by writing a .NET Office suite that would run (and be sold) on a boatload of platforms. That might make it a lot more believable. I won't hold my breath.

To paraphrase Wilde: "Education is all very well, but it is worth remembering that not everything worth knowing can be taught". Universities teach what can be taught and should not be criticised for that. In other fields (medicine, law, engineering...) students learn the "broader issues" later by other means, and CS should not be any different.

You make a good point. Unfortunately, most tech companies don't make any effort at all to mentor new grads under the wing of experienced programmers. Instead, especially of late, they tend to be focusing on bringing them in to lower their average salary numbers rather than to train them up to speed. That is unfortunate, and it used to be different in some of the better managed technology companies.

When I have been on syllabus committees, employers always want the students to know more about "real-world" programming/applications/systems/design/research (it varies depending on the employer) but they can not tell you how you might teach this.

:~)

That isn't surprising. Most of them can't even tell their own programmers what they want them to write. When is the last time anyone saw a really well written product specification that was completed before the coding (other than prototyping) started?

Looking at the "homework questions" of recent years, that trend is getting continually worse, or entrance criteria for CS programs is

Re: Checking for Modification to a Set of Files

bottoming out, or both.

This is called getting older. Every generation thinks this, in every subject area. Of course, in some generations and in some subjects it might actually be true, but the perception of it follows inevitably from gaining experience: "how could anyone not understand that?".

I hear you, but I'm talking about some pretty extreme cases here. I'm not wondering why entering freshmen don't immediately get pointer indirections. I see questions like "what is the purpose of the 'if' conditional?" and "what is a for loop?".

It's roughly the same amount of surprise as a medical school student asking "what is the purpose of an artery?" in a medical newsgroup. Once upon a time, that was common knowledge of all students long before they got out of High School.

and "it is not available for OS X" is probably false (depending on what you want to be included in the term "available").

"differencies" notwithstanding, apparently.

Indeed. You get to decide if it suits your needs based on some facts about it. Is that not a better situation to be in than assuming it is Windows only?

Not really. If there are no applications for Technology X outside of its original "native" platform, yet it is technically possible to produce them (due to claims of portability) after several years of availability as a development platform, then pretending like those other platforms are a viable tool is probably fooling yourself. Sun put Java out there, and it did not take years for Java apps to show up outside of a Solaris box. Of course, Sun actually *was* interested in fostering it off-platform, the exact opposite situation that Microsoft has been producing. With results that are similarly inverted.

--

Randy Howard (2reply remove FOOBAR)
"The power of accurate observation is called cynicism by those who have not got it." – George Bernard Shaw

Re: Checking for Modification to a Set of Files