A new scripting language doesn't solve everything

devblogs.microsoft.com/oldnewthing/20060427-21

April 27, 2006

Raymond Chen

Yes, there are plenty of scripting languages that are much better than boring old batch. Batch files were definitely a huge improvement over `SUBMIT` back in 1981, but they've been showing their age for quite some time. The advanced age of boring old batch, on the other hand, means that you have millions of batch files out there that you had better not break if you know what's good for you. (Sure, in retrospect, you might decide to call the batch language a design mistake, but remember that it had to run in 64KB of memory on a 4.77MHz machine while still remaining compatible in spirit with CP/M.)

Shipping a new command shell doesn't solve everything either. For one thing, you have to decide if you are going to support classic batch files or not. Maybe you decide that you won't and prefer to force people to rewrite all their batch files into your new language. Good luck on that.

On the other hand, if you decide that you will support batch files after all, then presumably your new command shell will not execute old batch files natively, but rather will defer to `CMD.EXE`. And there's your problem: You see, batch files have the ability to modify environment variables and have the changes persist beyond the end of the batch file. Try it:

```
C> copy con marco.cmd
@set MARCO=polo
^Z
        1 file(s) copied.
C> echo %MARCO%
%MARCO%
C> marco
C> echo %MARCO%
polo
```

If your new command shell defers to `CMD.EXE`, these environment changes won't propagate back to your command shell since the batch file modifies the environment variables of `CMD.EXE`, not your shell. Many organizations have a system of batch files that rely on the ability to pass parameters between scripts by stashing them into environment variables. The DDK's own `razzle` does this, for example, in order to establish a consistent build

environment and pass information to `build.exe` about what kind of build you're making. And I bet you have a batch file or two that sets your `PROMPT` or `PATH` environment variable or changes your current directory.

So good luck with your replacement command shell. I hope you figure out how to run batch files.

Raymond Chen

Follow