
Subject: Re: Spawn and Parallels

Posted by [mankoff](#) on Fri, 05 Sep 2008 13:33:15 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 5, 9:24 am, pgri...@gmail.com wrote:

> I am sorry I don't understand your problem... what's wrong with:

>

> slanze-14: idl

> IDL Version 6.3, Mac OS X (darwin i386 m32). (c) 2006, Research

> Systems, Inc.

> Installation number: XXXXX

> Licensed for use by: YYYYYYYYYYYY

> IDL> spawn,'pwd'

> /a/rom-48/vol/hea1/home/pgrigis

>

> Doesn't this work in parallel?

>

> Ciao,

> Paolo

>

> JMZawo...@gmail.com wrote:

>> I have a Mac running both OS X and Windows XP at the same time via

>> Parallels Desktop for Mac. I hate the way IDL works/looks/requiresX11

>> under OS X so I run IDL in a virtual machine with Windows XP.

>> Unfortunately, my current project requires that I SPAWN a process

>> under OS X (windows version of the software is not available). So, my

>> idea is to use SPAWN to launch terminal.app with the UNIT keyword and

>> use WRITEU to send commands to run the unix program. I can get SPAWN

>> to launch a terminal window under OS X, but I can't write a command

>> via the unit without IDL reporting that the pipe is broken and the

>> process being shut down. The terminal process does not terminate

>> though. I had to use the NOSHELL keyword to get the terminal process

>> to spawn.

>

>> So, I'm looking for either a fix or an alternative way to launch a

>> command line program on the unix side. Since I need to vary the

>> parameters in the command line from call to call, the usual mechanisms

>> that parallels uses to launch Mac apps from the virtual machine

>> probably won't work to directly launch the program I need to run. I'm

>> hoping that I'm just a little rusty with my unix skills and am missing

>> an obvious solution.

>

>> TIA, Joe

>

>

Re Craig & Paolo: He is in Windows, so he cannot interact directly with Unix. But Parallels can associate applications between the

operating systems, so he can have (minimal) control over Aqua applications, but not *nix apps. I use it (rarely) and it is a bit hard to understand what/how the OSES are interacting when it is happening in front of you.

The AppleScript idea could definitely work though.

And I propose a change to my original suggestion: Set up the terminal once, manually. Have it run a shell script and then exit. Then, rather than writing the .terminal file from IDL, just write the shell script unique/custom each time, and then launch the fixed .terminal file.

-k.
