

---

Subject: Re: idlde: Linux vs. Windows

Posted by [JD Smith](#) on Mon, 15 Sep 2003 18:57:11 GMT

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

---

On Mon, 15 Sep 2003 08:45:44 -0700, Chris ODell wrote:

- > So, after failing to get IDLWAVE going, I switched back to windows 2000.
- > (i know i know). It is working so far, but I'm intrigued by the idea of
- > nedit plus a command line...I may try that if i get forced back to
- > linux.
- >
- > Personally, i enjoy the feature of seeing all the in-memory variables in
- > their own little window in the IDLDE. the above would lose that nice
- > aspect.
- >
- > Geez, though, you'd think RSI would get on the ball a little and write a
- > decent de. I know there have been many other threads on this, but just
- > thought I'd let you all know that I wrote them and told them my
- > feelings, and the corresponder guy said he would "pass my comments along
- > to the relevant people". I remain skeptical :)
- >
- >

Not sure you gave IDLWAVE a fair shot. If you describe the problems you were having in detail, we could offer a solution. Then again, maybe you were looking for a reason to switch back... ;).

Regarding variable examination, IDLWAVE has a host of much more powerful examine commands than the DE (though, admittedly, none as obvious as the variables watch window). You can shift-middle-click on anything (variable, expression, function evaluation, or even random regions you drag out) to print it, and C-S-middle click it to popup a configurable list of examine commands (e.g. structure help, widget info, etc.). You're not limited to variables in the present scope, but can examine function returns, intermediate calculation products, etc. You can even examine variables futher up the calling stack! A few examples of types of custom examine commands you might see: print widget uvalues, invoke the `Print' method of objects, print mean/median/stddev statistics of arrays & vectors... really any inspection commands you might commonly find yourself typing at the command line.

JD

---