How Fast Can Windows Load a **Lot** of DLLs?

Gene Michael Stover

created Saturday, 2006 June 10
updated Sunday, 2006 August 13

Contents

1 What is this?

I was curious about how quickly Microsloth Winders can load a program that uses a *lot* of DLLs. So I wrote a Lisp program which writes DLL files & C programs that use them.

There probably isn't any practical use. I was just curious.

2 Running it yourself

Here's how to download & run the programs yourself:

1. Download `lotdll.zip` from `http://cybertiggyr.com/gene/lotdll/lotdll.zip` to your Microsloth Windows computer.
2. Unpack it into a directory of its own, such as `C:\lotdll\`.
3. Use a command line or *Windows Explorer* to navigate into the `bin` directory of that `lotdll` directory.

4. There are many programs in that `bin` directory. Most of them use some number of DLLs which are also in the directory. You can tell how many DLLs a program uses by the number in its name. For example, `prog0010.exe` uses ten of the DLLs, but `prog1000.exe` uses 1,000 of the DLLs.

   You can run those programs individually, but if you run `go.exe`, it will run all of them & display the times required for each.

The ZIP file also includes the source code.

3 The source code

"Source code" can be an ill-defined term. I wrote three programs for this test: `lotdll.lisp`, `go.c`, & `build.bat`. The Lisp programs in `lotdll.lisp` generate a lot of C files & a Makefile. C files are commonly considered source code, but the ultimate source code for this project is in those three files I created by hand.

If you want to start from scratch with my three ultimate source code files, you must download them & place them in the appropriate directories. Here's the directory structure, plus links to the files.[1]

- bin
- build.bat
- lib
- lotdll.lisp
- src
  - go.c
- tmp

You must create the directory structure on your computer, then download the three files into it.

Then fire-up your favorite Lisp system. In Lisp, do this:

1. `(load "lotdll.lisp")`
2. `(make-lotdll)`

Back on the Microsloth Winders command line, edit `build.bat`, if necessary. Then run it: `.\build`.

When it's done, you'll have all the programs in the `bin` directory.

number of DLLs	time (seconds)
1	0.00×10^0
10	0.00×10^0
100	2.00×10^0
1,000	1.30×10^1

Figure 1: Results of running `go.exe` on my computer

4 Conclusions

Figure 1 shows the results on my home Winders computer. My computer has Winders XP (home edition, methinks) running on a 1 GHz CPU with $\frac{1}{2}$ a gigabyte of real memory.

It looks like the cost is $O(N)$, maybe a little less, where N is the number of DLLs your program uses. Every 100 DLLs your program uses adds about 1 second to the program's start-up time.

A Other File Formats

- This document is available in multi-file HTML format at http://cybertiggyr.com/gene/lotdll/.
- This document is available in Pointless Document Format (PDF) at http://cybertiggyr.com/gene/lotdll/lotdll.pdf.

References

[1] The links are in the HTML version of this essay, not in the LaTeX version.