Building Cantera 1.5.4 from Source Code on a Windows PC

D. G. Goodwin
Caltech
June, 2004

This presentation covers...

- How to build the Cantera kernel from the source code using Visual C++ / Visual Fortran
- How to build the Cantera Python interface
- How to build the Cantera MATLAB toolbox

Before we start… are you *sure* you want to build Cantera from source code?

- Do you only want to use Cantera from Python or Matlab?
- If so, you can use the binary installer for Windows. Just download and run Cantera154.msi, and skip everything described here!
- But if you want to use Cantera from Visual C++ or Fortran on a Windows PC, continue to the next slide…

Getting Ready…

Things you will need before you start

- A PC with Windows XP or 2000
- Compilers
  - Microsoft Visual C++ 6.0
  - Compaq Digital Fortran 6.0
  - other compilers may work too but have not been tested
- Free downloads
  - The cygwin unix-like environment for Windows
  - Python
  - Numeric Extensions for Python (NumPy)

Why is cygwin needed?

- Cantera is designed to work on multiple platforms, including linux, unix, and Mac OS X
- On the other platforms, the GNU 'make' utility is used to build Cantera
- Impractical to maintain an entirely separate build procedure for Windows
- By installing cygwin, Windows PCs can use standard unix-like command-line tools, including 'make'

Why is Python needed?

- Python is an easy-to-use, cross-platform scripting language
  - much more powerful than DOS or 'cmd.exe'
  - much easier to use than 'sh' or 'perl'
- The Cantera build procedure writes and runs some Python scripts
- Cantera also uses Python during operation to parse Cantera input files ('CTI Files')
- (If you take an hour or so to learn Python, you'll find lots of uses for it too)

Why is NumPy needed?

- Adds fast MATLAB-like array functions to Python
- Required only if you want to use Cantera from Python
- Not needed if you only plan to use Python to process .cti files

Installing cygwin

- Get it from http://www.cygwin.com by running setup.exe
- When asked whether you want DOS or unix files, choose DOS
- You only need a minimal installation
  - make, sed, and bash are required
  - everything else is optional

Installing Python

- Information about Python is at http://www.python.org
- Run this to install Python: http://www.python.org/ftp/python/2.3.4/Python-2.3.4.exe

Installing NumPy

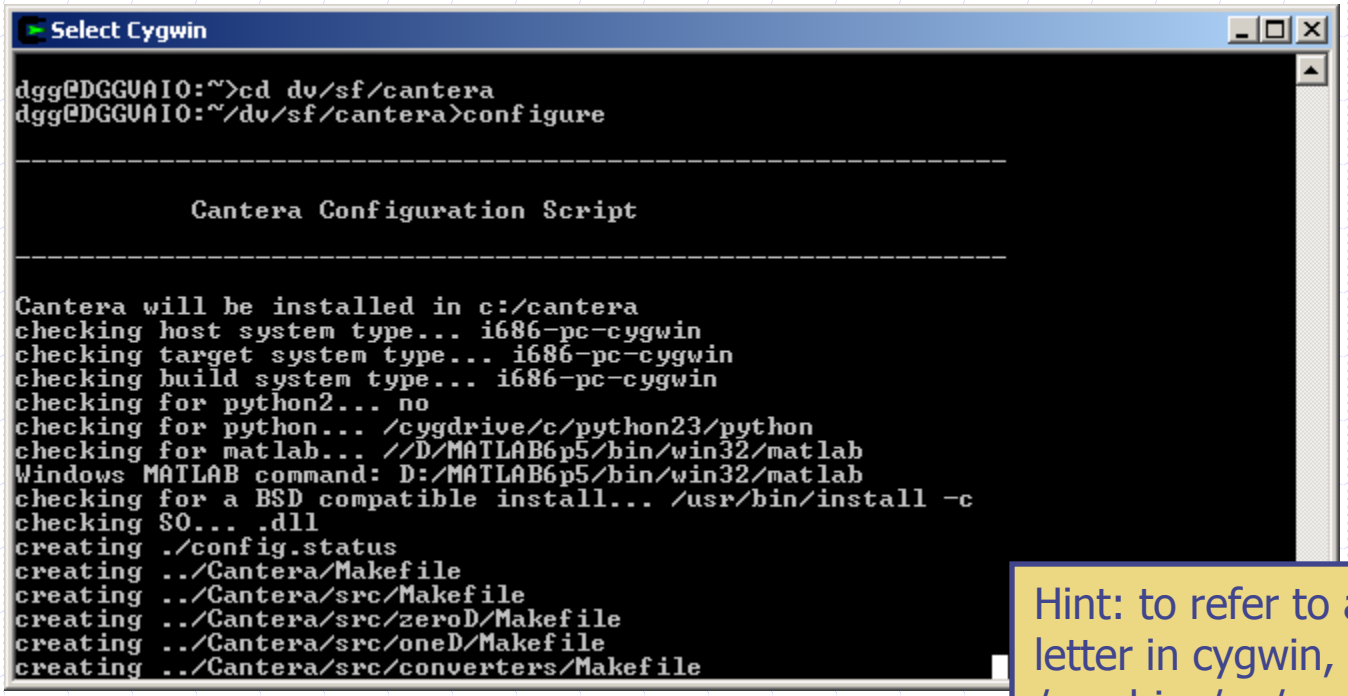
- Go to http://sourceforge.net/projects/numpy
- Under "latest file releases," select **numpy**
- Download and run **Numeric-23.1.win32-py2.3.exe** (or the latest version available)

Get the Cantera source code

- Go to http://sourceforge.net/projects/cantera
- Under "latest file releases," select package **cantera**
- Get file cantera-1.5.4-src.zip
- Extract its contents somewhere *other* than where you plan to install Cantera (e.g., not in c:\cantera)

Configuring the Installation

- Double-click the cygwin icon on the desktop to open a cygwin shell window
- 'cd' to the directory where you extracted the Cantera source code, and type 'configure' to run the configuration script



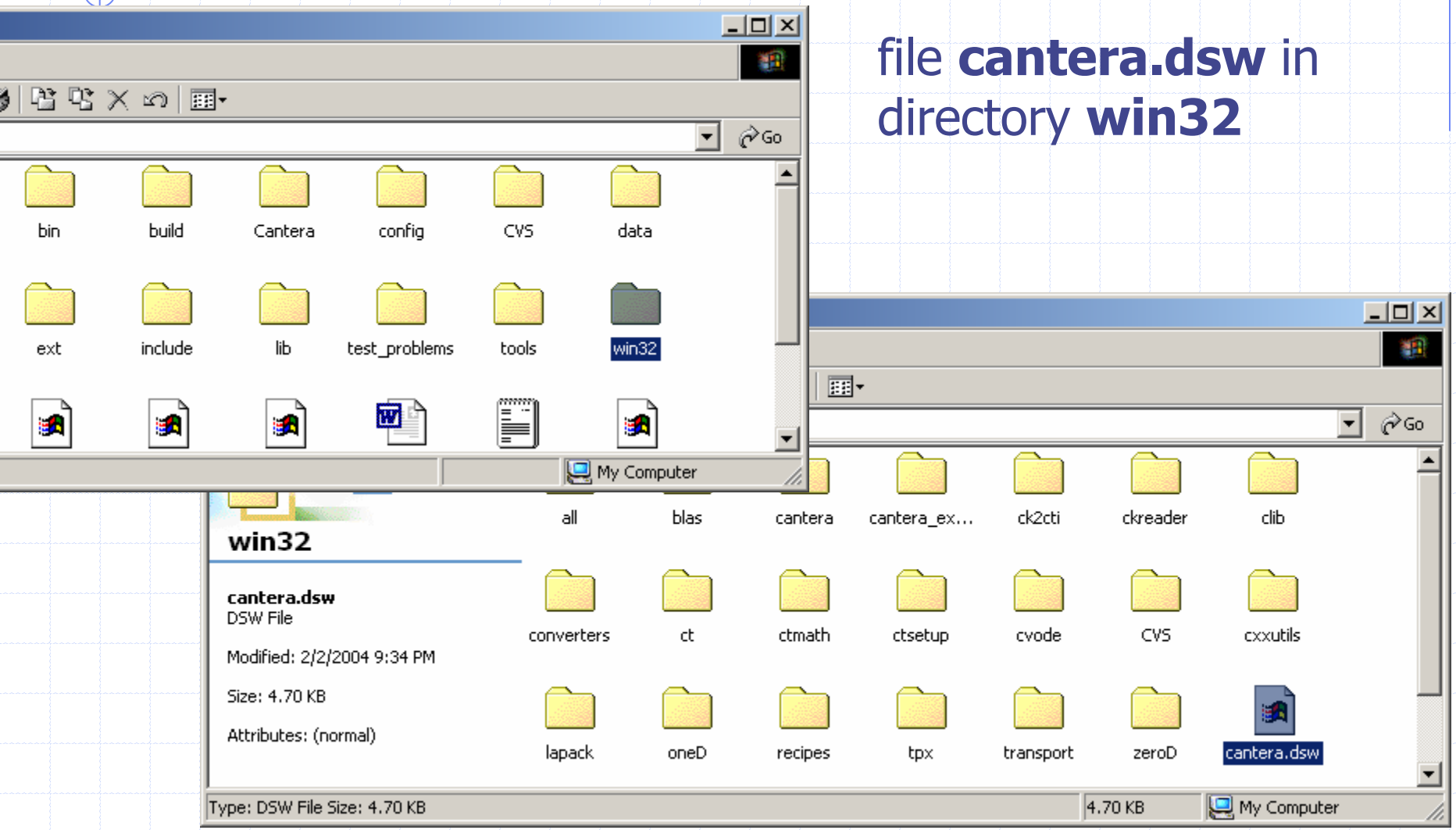
Hint: to refer to a DOS drive letter in cygwin, do it like this: /cygdrive/c, /cygdrive/d etc.

Choosing the Installation Directory

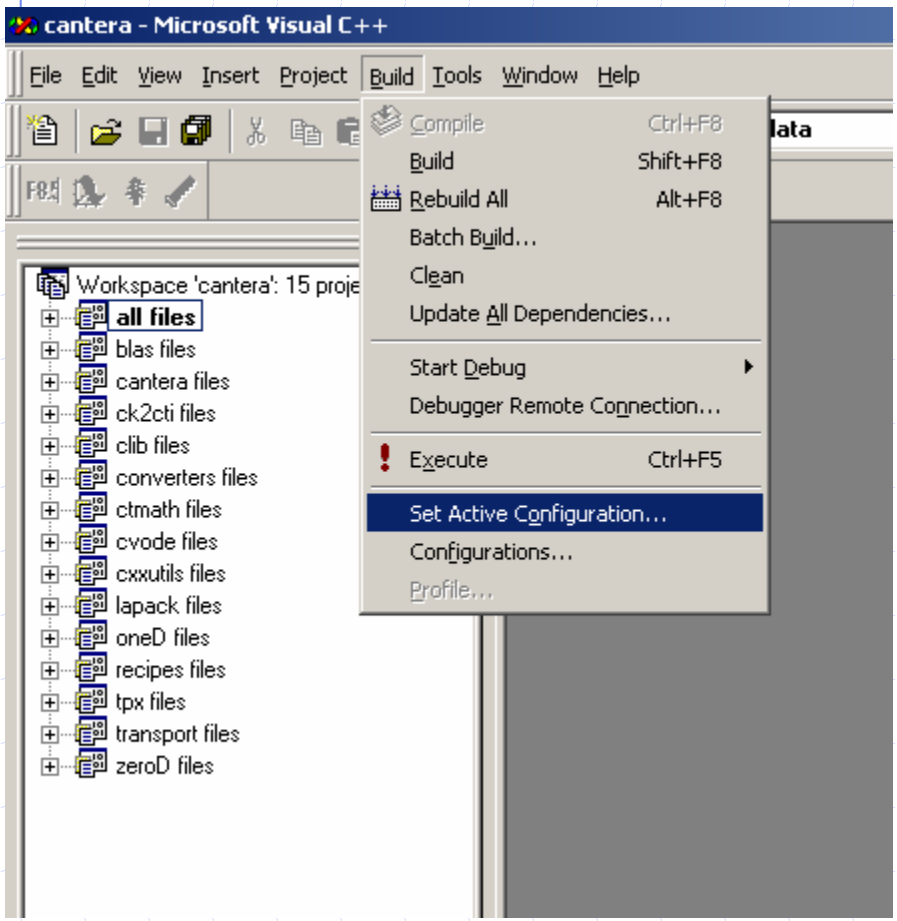
- By default, Cantera will be installed in c:\cantera
- To change this default, edit the configure script before running it
  - Change variable CANTERA_INSTALL_DIR to the desired installation location

Now open the Cantera Visual Studio project file

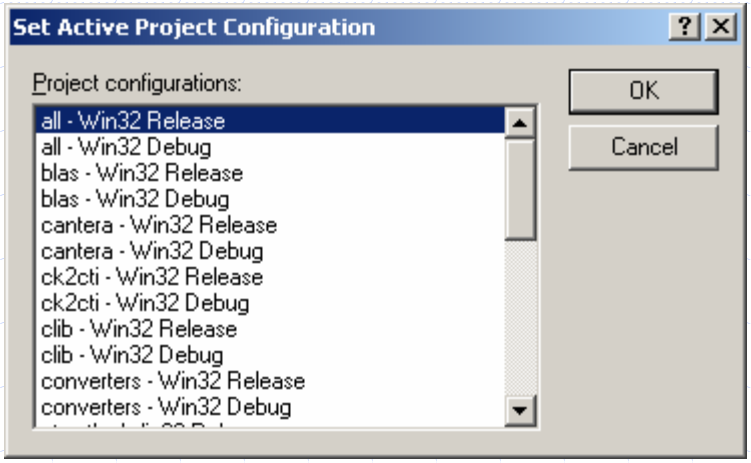
file **cantera.dsw** in directory **win32**



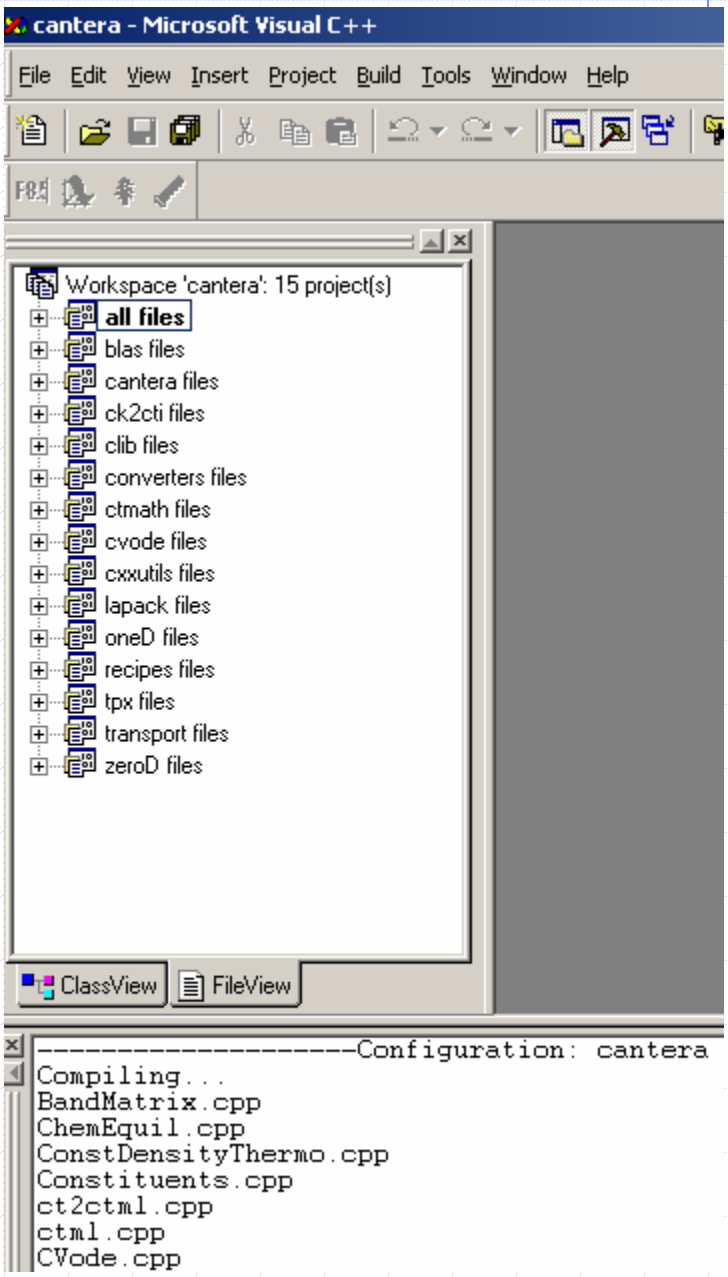
Set the active configuration



set the active configuration to 'all – Win32 Release'



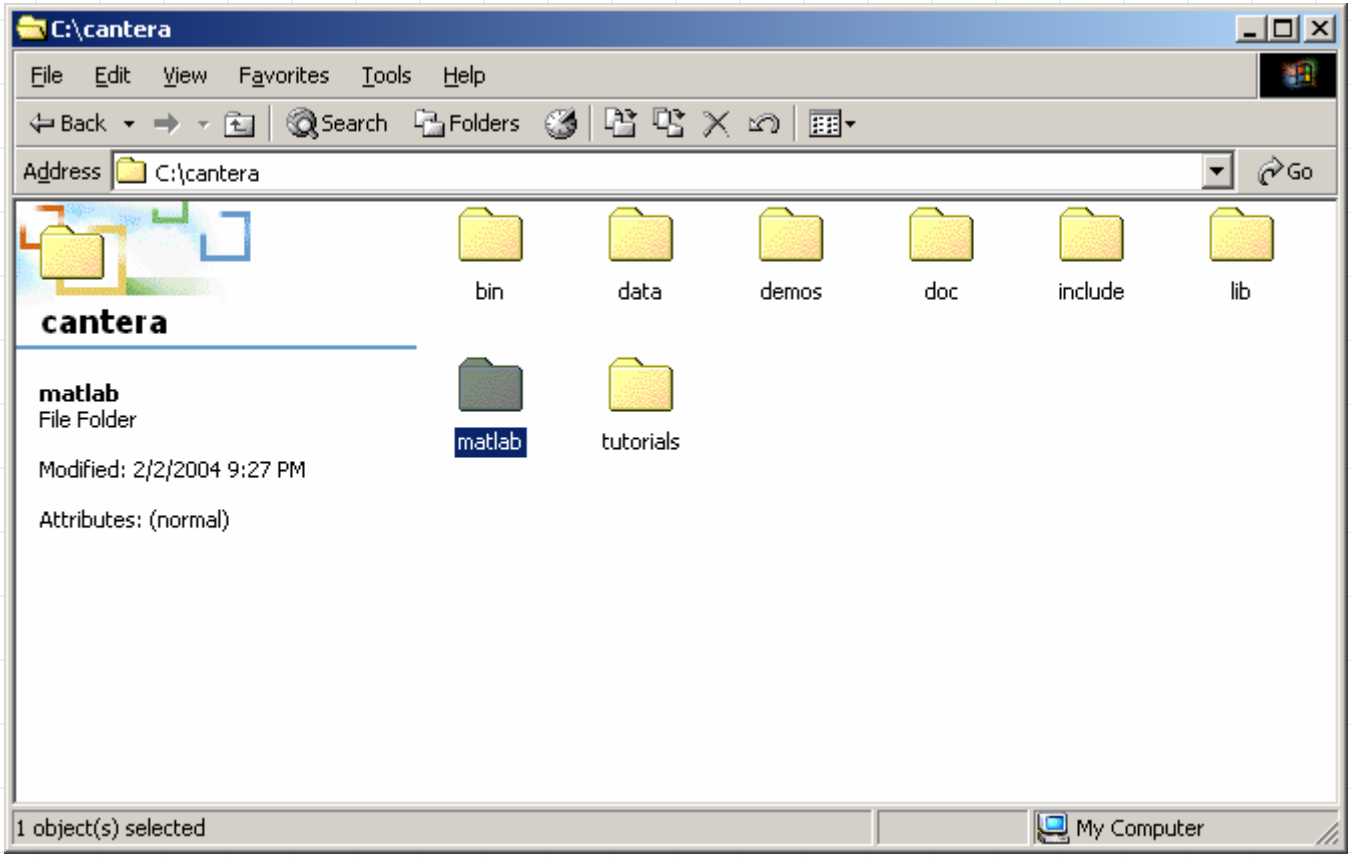
Build project ‘all’



Finish the Installation

- When the build of project 'all' finishes, return to the cygwin window
- type '**make win**' to build the Python interface and MATLAB toolbox
- type '**make win-install**' to install everything
- you should now have a functional Cantera installation

The installation directory should look like this when ‘make win-install’ finishes

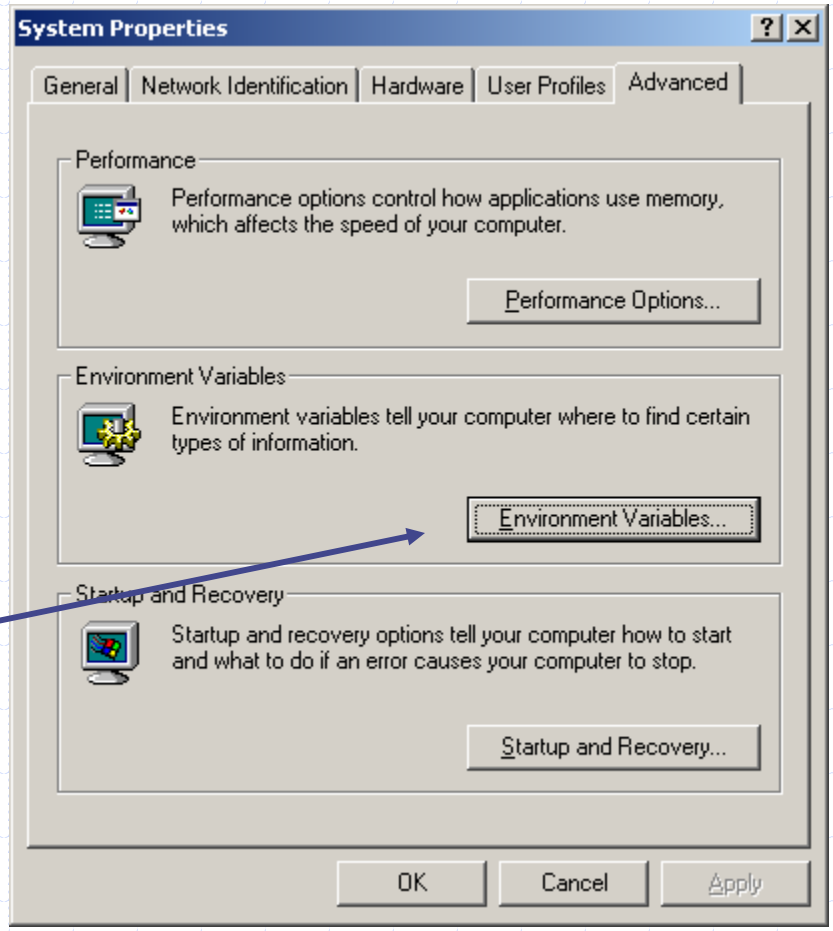


Finishing Up

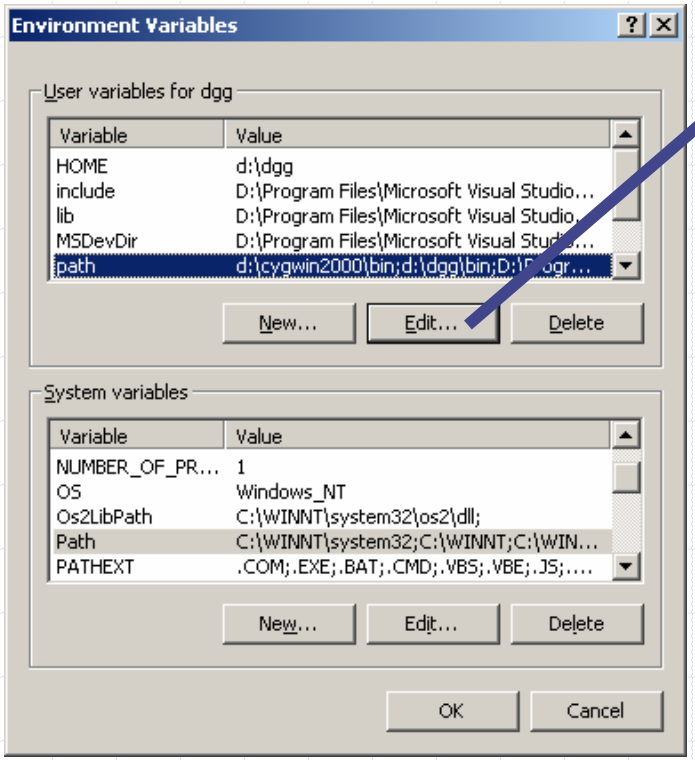
Cantera needs to know where to find the Python interpreter, since it uses Python to process '.cti' input files

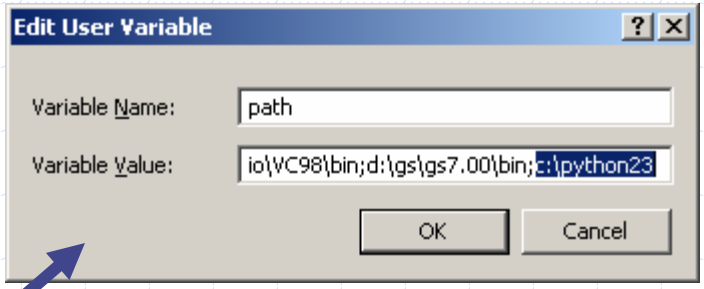
To edit the system search path, select "System" on the Control Panel to pull up this dialogue box

Press the 'Environment Variables' button



Edit the PATH environment variable

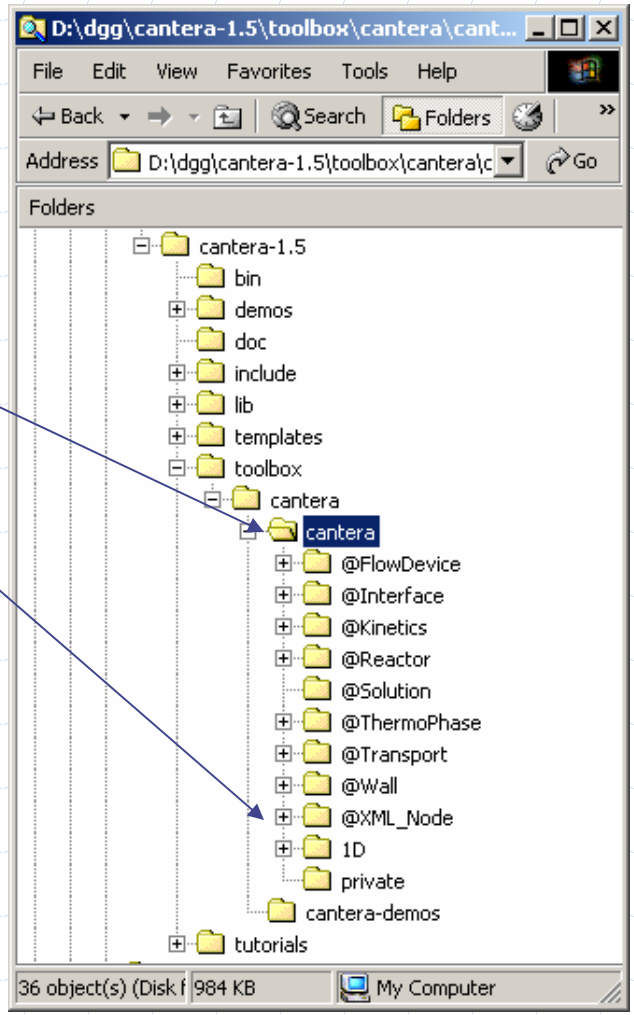




- Add C:\PYTHON23 (or wherever you installed Python 2.3) to the PATH environment variable (user or system)
- Use a semicolon between paths

If you will use MATLAB, put the Cantera Toolbox on the MATLAB path

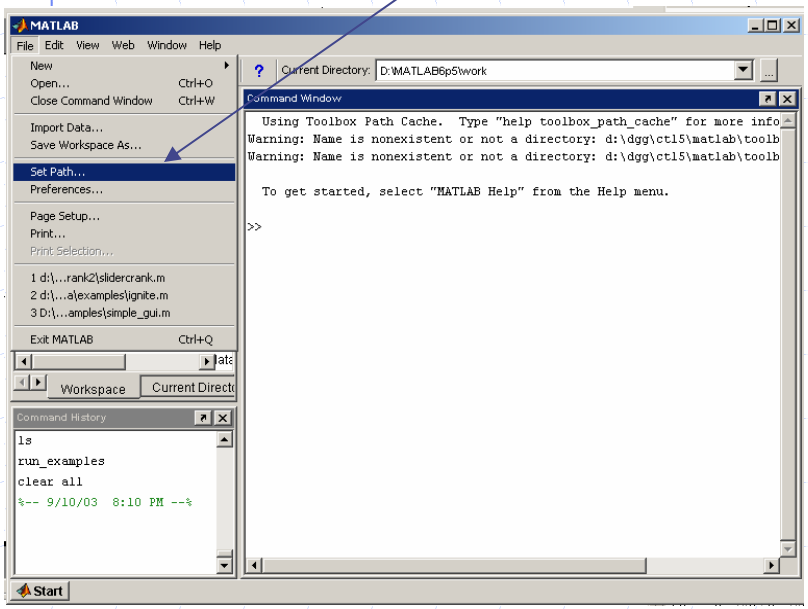
- Add these **two** folders to the MATLAB path



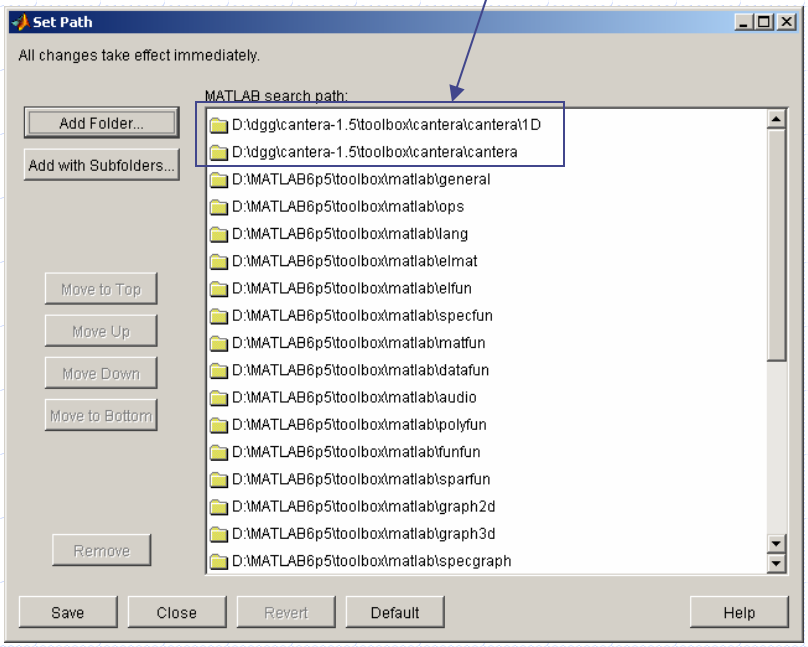
Folder '1D' adds support for one-dimensional reacting flows

Setting the Path in Matlab

choose 'Set Path…' on the File menu



the Cantera Toolbox



If desired, the Cantera toolbox can be moved to the folder containing other toolboxes (or anywhere else)

Now try it out!

- If you have done everything described here, you should have a functioning Cantera 1.5.4 installation! (Congratulations.)
- Running the Python and MATLAB demonstration scripts is a good way to test your installation