McStas installation instructions

Peter Willendrup,
peter.willendrup@risoe.dk

March 5, 2004

Abstract

This document describes installation of the McStas package, including some information on installing other required pieces of software.

The text below is also included as a chapter in the McStas manual.

1 Licensing

The conditions on the use of McStas can be read in the files `LICENSE` and `LICENSE.LIB` in the distribution. Essentially, McStas may be used and modified freely, and copies of the McStas source code may be distributed to others. New or modified component and instrument files may be shared by the user community, and we shall be glad to include user contributions in the package.

2 Installing support Apps

To get a fully functional McStas installation, a few support applications are required. Essentially, you will need a C compiler, Perl and Perl-Tk, as well as a plotter such as Matlab, Scilab or PGPLOT (which we do not support but is still functional). These should better be installed before McStas.

Under Windows, we recommand to avoid using spaces when specifying the location of packages during their installation.

2.1 C compiler

The C compiler is used for building the instrument simulation executable, which does the hard job. It is called transparently by the mcgui and mcrun McStas tools.

- Win32: Bloodshed Dev-C++ (Win32). To install Bloodshed Dev-C++, download the installer package from

  http://www.bloodshed.net/dev/devcpp.html.

  When installed, add the `C:\Dev-Cpp\bin` directory to your PATH using

  'Start/Settings/Control Panel/System/Advanced/Environment Variables'.
- Unix/Linux: standard C compiler. Most Unix/Linux systems come with a system C compiler, usually named 'cc' or 'gcc'. In case this would not be the case, install such a compiler from Linux packages (RedHat, SuSe or Debian), or compile it from scratch.

2.2 Gui tools (Perl + Tk)

The McStas tools are written using the Perl language, with calls to the perl-Tk library.

- Win32:
  - Get and install the ActivePerl package from http://www.activestate.com/Products/Download/Register.plex?id=ActivePerl (Registration not required) This package provides Perl and Perl-Tk. IMPORTANT: Get the 'msi' type package!
  - Optionally, get and install the ActiveTcl package from http://www.activestate.com/Products/Download/Register.plex?id=ActiveTcl (Registration not required) This package provides Tcl/Tk.
  - You may need to update/install the Microsoft Windows Installer. Get it from Windows Support Downloads at http://www.microsoft.com/.
  - Make sure that during installation, you have requested to attach the .pl and .tcl extensions to Perl and Tcl, and the `C:\Perl\bin` is in the PATH.
- Unix/Linux:
  - Install Perl, Tcl/Tk and perl-Tk. Prebuilt packages exist for most Linux distributions and also most other Unix-like operating systems. You will find these packages at http://freshmeat.net/, http://www.rpmfind.net/linux/RPM/ (for SuSe and RedHat) and using Fink Commander (for Mac OS X). Some Linux like system do not provide perl-Tk as an RPM/Debian file. You will then have to compile it from the source code (with ./configure; make; make install).

  - Consult the McStas webpage at http://mcstas.risoe.dk for updated links to the source code distributions.

2.3 Plotting backends (All platforms)

For plotting with McStas, different support packages can be used:

- PGPLOT/PDL (perl-PDL)/pgperl (perl-PGPLOT) (Unix only) - Binary builds of the packages exist for various Linux distributions (for instance Debian comes with prebuilt versions). Prebuilt versions also exist for some commercial Unix'es. Refer to distributor/vendor for documentation. The packages can also be built from source using some (in many cases much) effort. See the PGPLOT documentation for further details. You may need to define the `PGPLOT_DIR` variable to the location of the Pgplot library.
- Matlab (Some Unix/Win32) - refer to http://www.mathworks.com. Matlab licenses are rather costly, but discount programmes for university and research departments exist.
- Scilab (Unix/Win32/Mac...) - a free 'Matlab-like' package, available from http://www-rocq.inria.fr/scilab/.
  On Windows systems, Scilab should be installed in a directory such as `C:\Scilab` (do NOT use spaces in names) and you should then add the `C:\Scilab\bin` directory to your PATH using

  'Start/Settings/Control Panel/System/Advanced/Environment Variables'.

3 Getting McStas

The McStas package is available in three different distribution packages, from the project website at http://mcstas.risoe.dk, e.g.

- `mcstas-1.8-src.tar.gz`
  Source code package for building McStas on (at least) Linux and Windows 2000. This package should compile on most Unix platforms with an `ANSI-c` compiler. - Refer to section 4
- `mcstas-1.8-i686-unknown-Linux.tar.gz`
  Binary package for Linux systems, currently built on Debian GNU/Linux 3.0 'woody'. Should work on most Linux setups. - Refer to section 5
- `mcstas-1.8-i686-unknown-Win32.zip`
  Binary package for Win32 systems, currently built on Microsoft

Windows 2000 professional, using the gcc 2.95 compiler from Bloodshed Dev-C++ 5 Beta 7 - Refer to section 6

4 Source code build

The McStas package is beeing co-developed for mainly Linux and Windows systems, however the Linux build instructions below will work on most Unix systems, including Mac OS X). For an updated list of platforms on which McStas has been built, refer to the project website.

4.1 Windows build

- Start by unpacking the `mcstas-1.8-src.tar.gz` package using e.g. Winzip.
- Compile the McStas package using the `build.bat` script of the `mcstas-1.8` directory you just unpacked. Follow the on screen instructions.
- When the build has been done (e.g. `mcstas.exe` has been produced), proceed to install (Section 6).

4.2 Unix build

McStas uses autoconf to detect the system configuration and creates the proper Makefiles needed for compilation. On Unix-like systems, you should be able to compile and install McStas using the following steps:

1. Unpack the sources to somewhere convenient and change to the source directory:
   ```
   gunzip -c mcstas-1.8-src.tar.gz — tar xf -
   cd mcstas-1.8/
   ```
2. Configure and compile McStas:
   ```
   ./configure
   make
   ```
3. Install McStas (as superuser):
   ```
   make install
   ```

The installation of McStas in step 3 by default installs in the /usr/local/ directory, which on most systems requires superuser (root) privileges. To install in another directory, use the –prefix= option to configure in step 2. For example,

```
./configure –prefix=/home/joe
```

will install the McStas programs in /home/joe/bin/ and the library files needed by McStas in /home/joe/lib/mcstas/.

In case ./configure makes an incorrect guess, some environment variables can be set to override the defaults:

- The CC environment variable may be set to the name of the C compiler to use (this must be an ANSI C compiler). This will also be used for the automatic compilation of McStas simulations in mcgui and mcrun.
- CFLAGS may be set to any options needed by the compiler (eg. for optimization or ANSI C conformance). Also used by mcgui/mcrun.
- PERL may be set to the path of the Perl interpreter to use.

To use these options, set the variables before running ./configure. Eg.

```
setenv PERL /pub/bin/perl5
./configure
```

It may be necessary to remove configure's cache of old choices first:

```
rm -f config.cache
```

If you experience any problems, or have some questions or ideas concerning McStas, please contact peter.willendup@risoe.dk or the McStas mailing list at neutron-mc@risoe.dk.

You should try to make sure that the directory containing the McStas binaries (mcstas, gscan, mcdisplay, etc.) is contained in the PATH environment variable. The default directory is /usr/local/bin, which is usually, but not always, included in PATH. Alternatively, you can reference the McStas programs using the full path name, ie.

```
/usr/local/bin/mcstas my.instr
perl /usr/local/bin/mcrun -N10 -n1e5 mysim -f output ARG=42
perl /usr/local/bin/mcdisplay --multi mysim ARG=42
```

This may also be necessary for the front-end programs if the install procedure could not determine the location of the perl interpreter on your system.

If McStas is installed properly, it should be able to find the files it needs automatically. If not, you should set the MCSTAS environment variable to the directory containing the runtime files "mcstas-r.c" and

”mcstas-r.h” and the standard components (*.comp). Use one of

```
MCSTAS=/usr/local/lib/mcstas; export MCSTAS # sh, bash
setenv MCSTAS /usr/local/lib/mcstas         # csh, tcsh
```

5 Binary install, Linux

Should be very easy, simply start from ’make install’ in Section 4.

6 Binary install, Windows

- Start by unpacking the `mcstas-1.8-i686-unknown-Win32.zip` package using e.g. Winzip.
- Execute the `install.bat` installation script. Follow the on screen instructions.
- Set the required (see output of `install.bat`) environment variables using

  ’Start/Settings/Control Panel/System/Advanced/Environment Variables’

  - `PATH` Append e.g. `C:\mcstas\bin`
  - `MCSTAS` Create it as e.g. `C:\mcstas\lib`

It is important that Perl is correctly installed to execute all the McStas tools (e.g. mcdoc.pl, mcrun.pl, mcgui.pl, ...). Create a shortcut of the `C:\mcstas\bin\mcgui.pl` on your Desktop (the icon should be a yellow dot). Whenever launched from the Windows Command window (cmd), you must specify the .pl extension to all McStas Perl script commands (e.g. ’mcrun.pl’ and ’mcgui.pl’, not ’mcrun’ or ’mcgui’) except for `mcstas` itself.

Actually, the modifications to your environnement variables should be

- `PATH` Append e.g.
  `C:\mcstas\bin;c:\Program Files\Scilab-2.7.2\bin;C:\Perl\bin\;C:\Dev-Cpp\bin;`
- `MCSTAS` Create it as e.g. `C:\mcstas\lib`

using menu item ’Start/Settings/Control Panel/System/Advanced/Environment Variables’. On some Windows systems, it may be necessary to logout/login (no need to restart the computer) to make these changes active for the whole session.

7 Finishing and Testing the McStas distribution

The `examples` directory of the distribution contains a set of instrument examples. These are used for the McStas self test procedure, which is executed with

```
mcrun --test
```

This test takes a few minutes to complete, and ends with a short report on the installation itself, the simulation accuracy and the plotter check.

You should now be able to use McStas. For some examples to try, see the examples/ directory. Start 'mcgui' (mcgui.pl on Windows), and select one of the examples in the 'Neutron Sites' menu.

Important note for Windows users: It is a known problem that some of the McStas tools do not support filenames / directories with spaces. We are working on a more general approach to this problem, which will hopefully be solved in the next release.