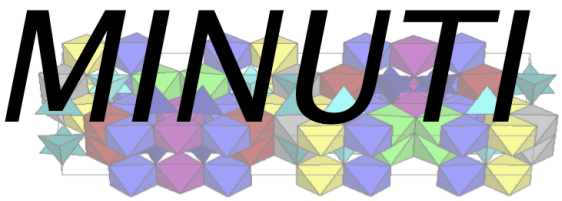

Version 2.0.0

Installing from Source

October 31, 2017

Wolfgang Sturhahn

http://www.nrixs.com support@nrixs.com Follow @NRIXS on Twitter

Contents

1 License information

MINUTI-2.0.0 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MINUTI-2.0.0 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is distributed with MINUTI-2.0.0 and can also be seen at `http://www.gnu.org/licenses/`.

Any use of results obtained using MINUTI-2.0.0 in related or unrelated publications have to be properly acknowledged by reference to the name of the package, to the name of the developer(s), and to the *NRIXS software* site `http://www.nrixs.com`.

2 What is MINUTI ?

The MINUTI (MINeral physics UTIlities) software is a collection of programs aimed at use by mineral physicists. At present three independent modules are provided.

- `seos`
  calculation and data evaluation of compression curves including temperature dependence and spin crossover effects;
- `simx`
  calculation and data evaluation of melting curves obtained by observation of time-integrated nuclear forward scattering intensity;
- `svec`
  calculation of sound velocity surfaces and averages using the elastic tensor.

The software was created by W. Sturhahn to offer traceable evaluation codes for publications in mineral physics. The MINUTI core programs are written in Fortran90 with wrappers using c-shell scripts. The GUI is implemented as Tck/Tk script and requires Tcl version 8.6 or higher to run. The MINUTI software has been extensively tested over the last four years. Just like the other software packages of *NRIXS software*, MINUTI installs on UNIX-like operating systems: Sun's Solaris, Apple's Mac OS X, Ubuntu, and various other Linux versions. Installation on Microsoft's Windows operating systems requires a Linux/Unix emulator software, e.g., cygwin, or preferably a virtual machine hosting a Linux-type guest-system, e.g.,VirtualBox.

3 Requirements

Before installation please verify the following list of requirements.

- Fortran90 compiler. The recommended compiler is gfortran, version 4.6 or higher is required;
- Tcl/Tk version 8.6 or higher to run the GUI ;
- Optional: Fortran graphics tools libraries `libg2.a` and `libgd.a` - these libraries can be obtained at `http://g2.sourceforge.net/` and `http://www.libgd.org`.

- Optional: Grace plotting tool, on many linux-type systems available in native software installer. On MacOS it is available via the port facility.

On MacOS systems there are the following additional requirements.

- `Xcode`, Apple's developer package freely available from the Appstore. Run `Xcode` to accept the license and make sure 'linecommand tools' are installed: open Terminal app and type `xcode-select --install` on the command line (you need administrator privileges for this). Install `Xcode` before you install the Fortran compiler.
- `X11` is not distributed with MacOS. It is available at `http://xquartz.macosforge.org` .

4 How to install

MINUTI is distributed as a compressed tar-ball named `MINUTI-2.0.0.tar.gz`. Several steps are needed to install the MINUTI software. If this is an update from an earlier version of MINUTI it is recommended that you uninstall the earlier version as described in section 6.

4.1 Extract files

Depending on the available system utilities you may have several options to extract the MINUTI files. For example, the line command '`gzip -dc MINUTI-2.0.0 | tar xf -`' will recover the files on most systems. Depending on your system capabilities double-click on `MINUTI-2.0.0.tar.gz` to recover the folder `MINUTI-2.0.0`.

4.2 Configure and install

Next, run the configuration script to create the configuration-dependent files, to compile of the source code, and to install the executables. For installation, proper authorization is required. On MacOS, authorization is requested when needed during the installation. Otherwise the configure script might have to be run as a 'sudo' command.

Several options can be specified to adjust to local circumstances. For most cases, options are probably not needed. Enter the following line commands to see available options.

```
> cd MINUTI-2.0.0
> ./configure --help
Usage: configure [OPTION]...
Configure, compile, and install MINUTI

      --foption=<list> colon separated list of compiler options
      --fortran=<exe>  use Fortran compiler <exe>
                       <exe> must exist in search path
      --nographics     no graphics support
      --help           display this help and exit
      --loption=<list> colon separated list of ld options
      --lpath=<list>   prepend directories to library search path
                       <list> is a colon separated list of directories
      --prefix=<dir>   set <dir> as installation location
                       proper authorization is required
                       preset location is <home directory>
      --spath          show search path
      --spath=<list>   prepend directories to search path
                       <list> is a colon separated list of directories
      --static         create statically linked executables

Examples:
```

```
  configure --lpath=$HOME/lib   prepend $HOME/lib to library path
  configure --prefix=/usr/local install into directory /usr/local
>
```

Here follows a description of the functionality of each option for the configure script.

foption list of options that will be passed to the Fortran compiler. This might be needed for some compilers. The correct syntax would be, for example, '--foption=-x1=a:-x2=b'. Everything after the first '=' character will be passed to the compiler with all ':' characters being replaced by spaces.

fortran name of the Fortran compiler executable. By default the configuration script assigns a the first compiler that is found in the search path (see description below) from this list: gfortran, g95. The correct syntax would be, for example, '--fortran=/crazypath/fort'.

nographics disable graphics support. By default the configuration script tries to find graphics libraries and compile accordingly. This option disables this behavior. The MINUTI executables still support the `xmgrace` visualization tool if found in the searchpath (see below).

loption list of options that will be passed to the linker; see 'foption' above for syntax rules.

lpath prepend directories to the library search path. By default the path contains the following directories: /lib, /usr/lib, /usr/local/lib. Adding directories may for example be required if the graphics libraries `libg2.a` and `libgd.a` were installed at a different location, and graphics support is desired. The correct syntax would be, for example, '--lpath=/crazypath/dir1:~/dir2', where '~' symbolizes the user's home directory.

prefix set the location for installation of the MINUTI executables and their support files. By default the installation location is the installer's home directory: under MacOS the MINUTI app is copied into $HOME/Application and support files are copied into $HOME/Library/NRIXS/MINUTI; under other Unix systems the executables are copied into $HOME/bin and the support files are copied into $HOME/.NRIXS/`MINUTI`. If the prefix is set to another directory it is important to have appropriate permissions as installer, e.g., '--prefix=/usr/local' requires administrator or root privileges. Under MacOS support files are installed either into $HOME as described above or into '/' where the MINUTI application is copied into '/Applications' and support files are copied into '/Library/NRIXS/MINUTI'.

spath prepend directories to the search path for essential support programs including the compiler. By default the path contains the following directories: /bin, /usr/bin, /usr/local/bin, /opt/bin, /opt/local/bin, /usr/ccs/bin, /usr/ucb, /usr/openwin/bin, /usr/X11/bin. If the configuration script fails to locate essential support programs in this path (reported upon execution of the script) directories have to be added; see 'lpath' above for syntax rules.

static create statically linked executables. This option is meant for developers and not recommended for normal use.

4.3 Create and install executables

Execute the configuration script with appropriate options.

```
> ./configure [OPTIONS...]
  ....
  (messages)
  ....
>
```

The configuration script may be re-run with different options as needed. The last configuration command is saved in the file 'configcmd'. The script produces various messages on the progress of the configuration, compilation, and installation.

If DIR is the installation directory the following files are copied into 'DIR/bin': minuti (Linux OS only), seos, simx, svec, seos-`2.0.0`, simx-`2.0.0`, svec-`2.0.0`. If graphics libraries were found also sdsp and sdsp-`2.0.0` should have been copied into 'DIR/bin'. On MacOS operating systems also the application Minuti.app should have been created or overwritten in '$HOME/Applications' as described above or into '/Applications'.

The installed executables are only accessible by line command if DIR/bin is part of the '`path`' setting in the login resource file in your home directory. This can be tested by typing '`echo $PATH`' or '`echo $path`' at a terminal prompt. If the directory DIR/bin is not part of the listing then the login resource file, usually something like '`.bash_profile`', '`.profile`', or '`.login`', must be edited to include DIR/bin in the '`path`' setting. After that you have to logout and login again to update the '`path`' settings.

5 File locations

5.1 Binaries

The executable binaries are placed into the directory 'DIR/bin' for a install into directory DIR, i.e., './install --prefix=DIR'.

5.2 GUI

For MacOS operating systems, the GUI is created as Minuti.app and placed into directories '$HOME-/Applications' or '/Applications' for a system-wide install. The app shows up automatically on Launchpad. For local installations, '$HOME/Applications' has to be linked into '/Applications' for MacOS to see the app. Open a Terminal, then 'cd /Applications' and 'sudo ln -s $HOME/Applications localApps'. Admin privilege is needed for the latter command.

On Unix-type systems, the GUI is created as minuti and placed into the directory 'DIR/bin' for a install into directory DIR. Depending on the particular type of window management (Gnome, KDE, etc) you may create a launcher that points at 'DIR/bin/minuti'.

5.3 Support files

For MacOS operating systems, support files are placed into directories '$HOME/Library/NRIXS/MINUTI' or '/Library/NRIXS/MINUTI' for a system-wide install. On Unix-type systems, support files are placed into directories '$HOME/.NRIXS/MINUTI' or 'DIR/share/NRIXS/MINUTI' for a system-wide install into directory DIR. If a particular directory exists the content will be saved into a same-name directory with a number appended.

6 How to uninstall

6.1 Uninstall software package

The MINUTI program package is uninstalled by the command

```
> cd MINUTI-2.0.0
> ./uninstall
```

```
   ....
   (messages)
   ....
>
```

This step removes all files that were created during installation and is recommended prior to installation of a new version of MINUTI.

6.2 Clean directories

The MINUTI directories are cleaned by using the command

```
> cd MINUTI-2.0.0
> make clean
   ....
   (messages)
   ....
>
```

In addition to the executables bin/seos-2.0.0, bin/simx-2.0.0, bin/svec-2.0.0, bin/sdsp-2.0.0, and the library lib/libminuti-2.0.0.a, all compiler output will be deleted. This step is optional after installation.

7 How to test with command line interface

Examples are provided with the MINUTI package. They are located in the 'MINUTI-2.0.0/examples' directory and after installation also in the 'examples' directory as support files. Your computer system can only access the MINUTI executables if the directory that they are located in is part of the '`path`' defined for your system at the time of login. You can test this, e.g., by trying something like

```
> cd ~
> which seos
/usr/local/bin/seos
```

If the second command doesn't succeed the location of the MINUTI executables can be made known permanently to your computer. This is done by modification of the '`path`' setting in the login resource file in your home directory, usually something like '`.bash_profile`', '`.profile`', or '`.login`'. You have to logout and login again to update the '`path`' settings.

Now change into an 'examples' directory (you need write access). If you don't have write access you can copy the directory to an accessable location, e.g., your home directory. For example, enter something like

```
> cd ~/MINUTI-2.0.0/examples/seos_quartz
> ls -px
Results/     in_seos      quartz.dat
> seos --help
Usage: seos [OPTION]...
Run MINUTI executable seos-2.0.0

      --geometry=<XxY+U+W> set the window geometry for
                           graphics display.
      --help               display this help and exit
      --infile=<file>      use <file> as input file
                           the default input file is 'in_kctl'
      --nographics         suppress visualization support
      --pipe=<fifo>        use <fifo> as pipe to output graphics data
      --vdelay=<val>       set visualization startup delay to <val>
```

```
      --version             display version number and exit
      --vtool=<exe>         use <exe> for data visualization

Examples:
  seos --infile=MyFile  use 'MyFile' as input file.
  seos --nographics     disable visualization.

> seos

++ MINUTI-2.0.0 Copyright (C) 2017 Wolfgang Sturhahn
   ....
   (messages)
   ....
-- CPU time :  user   0.02 s  system   0.00 s
-- MINUTI module SEOS finished

> ls -px
Results/              in_seos               quartz_bdp.csv
quartz_bdt.csv        quartz_bms.csv        quartz_ccv.csv
quartz_csv.csv        quartz_dat.csv        quartz_dns.csv
quartz_ffm.csv        quartz_fft.csv        quartz_int.csv
quartz_ptl.csv        quartz_rsd.csv        quartz_smv.csv
quartz_thx.csv        quartz_vol.csv        quartz_vrd.csv
quartz_dsv.csv        quartz_p02v01_eel.csv quartz_p02v01_nel.csv
quartz_p03v01_eel.csv quartz_p03v01_nel.csv quartz_p03v02_eel.csv
quartz_p03v02_nel.csv quartz_vkk.csv
>
```

Several files were created during this fit of p-V data of quartz. Compare the content of output files with files provided in the '`Results`' directory. The meaning of file contents is explained briefly at the end of the quartz_ptl.txt file and in more detail in the MINUTI manual.

8 How to test with graphical user interface

If Tcl/Tk 8.6 or higher was accessible during install the GUI app was created.

- MacOS
  With Finder locate the Application folder into which the GUI app was installed. This is either '$HOME/Applications' for a local install or '/Applications' for a system-wide install. Start the GUI by double-click on Minuti.app but you might have to defeat the Gate Keeper if this doesn't work. Defeating the Gate Keeper: right-click on Minuti.app to get pull-down menu; select open; acknowledge to open. Defeating the Gate Keeper on Sierra: double-click on Minuti.app; acknowledge that it can't be opened; select Apple -> System Preferences -> Security & Privacy -> General; at the bottom right click open Minuti.app.

- other Unix
  Open terminal and type 'minuti' on the command line.

Select MINUTI -> Show Examples. A list of examples shows up in lower right panel; right-click on an item and select 'Open As Project'; click on 'run' button on top of middle panel to execute the calculation. Inspect output by double-click on thumbnail in lower left or by actions in upper right panel.