MINUTI

Version 2.3.4

Installation Guide
Source Distribution

January 1, 2025

Wolfgang Sturhahn

Contents

1 License information

MINUTI-2.3.4 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-2.3.4 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-2.3.4 and can also be seen at `http://www.gnu.org/licenses/gpl.html`.

Any use of results obtained using MINUTI-2.3.4 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 `https://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;

- `smix`
  match a set of mineral properties such as elastic constants and density to their target properties using linear mixing;

- `svec`
  calculation of sound velocity surfaces and averages using the elastic tensor;

- `sbck`
  background determination and subtraction for a related set of XRD patterns;

- `sfit`
  calculation of general functions including XRD patterns and fitting to experimental data.

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 bash shell scripts. The GUI is implemented as Tcl/Tk script and requires Tcl version 8.6 or higher to run. The MINUTI software has been extensively tested and improved since 2013. Just like the other software packages of *NRIXS software*, MINUTI installs on UNIX-like operating systems: Sun's Solaris, Apple's MacOS, Ubuntu, and various other Linux versions. For installation on Microsoft's Windows operating systems, a virtual machine, e.g.,VirtualBox (free) or VMware (paid), hosting a Linux-type guest-system is needed.

3 Requirements

Before installation please verify the following list of requirements. You must install these before installation of MINUTI.

3.1 Linux and Unix

- bash shell, should be installed already.

- Fortran90 compiler. The recommended compiler is gfortran 4.8 or later, available in native software installer.

- Tcl/Tk version 8.6 or higher to run the GUI, available in native software installer.

- Optional: xmgrace plotting tool, available in native software installer.

- Optional: ImageMagick for image conversions, available in native software installer.

3.2 MacOS

The order of installation is most important here.

- `Xcode`, Apple's developer package freely available from the Appstore.
  Run `Xcode` to accept the license and install the 'linecommand tools': open Terminal app and type `sudo xcode-select --install` on the command line. You must have administrator privileges for this.

- bash shell, should be installed by Xcode.

- macports or homebrew package installer.

- Fortran90 compiler. The recommended compiler is gfortran, version 4.8 or later, available via macports or homebrew package installer. Binary packages are available here.

- Tcl/Tk version 8.6 or higher to run the GUI, available via homebrew package installer.

- Optional: Xquartz for run-time graphics.

- Optional: xmgrace plotting tool, available via macports or homebrew package installer.

- Optional: ImageMagick for image conversions, available via macports or homebrew package installer.

4   How to install

MINUTI is distributed as a compressed tar-ball named `MINUTI-2.3.4_source.tar.gz`. Several steps are needed to install the MINUTI software. If this is an update from a MINUTI version earlier than 2.3.1 it is recommended that you uninstall the earlier version as described in section 6.

4.1   MacOS: remove restrictions

Before you extract the files, you must defeat the Gate-keeper mechanism and you need administrator privileges for this. Assume Dir as the location of MINUTI-`2.3.4_source.tar.gz`. Open Terminal app and enter

```
> cd Dir
> xattr MINUTI -2.3.4_source.tar.gz
>    ... list of extended attributes ...
> sudo xattr -c MINUTI -2.3.4_source.tar.gz
> Password:
> xattr MINUTI -2.3.4_source.tar.gz
>
```

4.2   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-2.3.4_source.tar.gz | tar xf -' will recover the files on most systems. Also a double-click on `MINUTI-2.3.4_source.tar.gz` recovers the folder `MINUTI-2.3.4` in many cases.

4.3   Configure, compile, and install

Next, run the installation script to create the configuration-dependent files, to compile the source code, and to install the executables. It is generally less complicated to install locally, i.e., into a user directory. For system-wide installation, proper authorization is required. On MacOS, authorization is requested when needed during the installation. Otherwise the install script might have to be run as a 'sudo' command. The installation process can be terminated by Ctrl-c.

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 -2.3.4
> ./install --help
Usage: install [OPTION]...
Compile and install MINUTI -2.3.4

     --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
     --prefix=<dir>   set <dir> as installation location
                      default location is <home directory >
     --spath          show search path
     --spath=<list>   append or prepend directories to search path
                      to prepend add leading _ to directory name
                      <list> is a colon separated list of directories
     --static         create statically linked executables

Examples:
```

```
   install --lpath=$HOME/lib    prepend $HOME/lib to library path
   install --prefix=/usr/local  install into directory /usr/local
                                 MacOS only: install app into /Applications
>
```

Here follows a description of the functionality of each option for the install 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 gfortran as compiler if is found in the search path (see description below). Change the compiler by '--fortran=/crazypath/fort'.

nographics disable run-time graphics support.

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

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 append or prepend directories to the search path for essential support programs including the compiler and run-time graphics programs. By default the path contains the following directories: /bin, /usr/bin, /usr/local/bin, /opt/bin, /opt/local/bin, /opt/homebrew/bin, /opt/X11/bin, /usr/X11/bin. If the install script fails to locate essential support programs in this path (reported upon execution of the script) directories have to be added. The correct syntax would be, for example, '--spath=/crazypath/dir1:~/dir2', where '~' symbolizes the user's home directory.

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

Execute the install script with appropriate options.

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

The install script may be re-run with different options as needed. The last install command is saved in the file 'installcmd'. The script produces various messages on the progress of the configuration, compilation, and installation. After installation, you have the option to run a test of the MINUTI executables, see section 7 for more details.

The install requires write access to the installation directory, by default the users home directory. If DIR is the installation directory the following files are copied into 'DIR/bin': minuti (Linux OS only),

minuti_test, seos, simx, smix, svec, sbck, sfit, seos-2.3.4, simx-2.3.4, smix-2.3.4, svec-2.3.4, sbck-2.3.4, sfit-2.3.4, sfit_api.tcl . On MacOS operating systems also the application Minuti.app should have been created or overwritten in '$HOME/Applications' as described above or in '/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.

4.3.1 Search path

Pay attention to the search path.

```
> ./install --spath
/bin:/usr/bin:/usr/sbin:/usr/local/bin:/opt/local/bin:/opt/X11/bin:/usr/X11/bin
>
```

If the location of the compiler or your xmgrace executable is not included in this search path you must add those locations to the path. For example, you installed xmgrace into directory '~/mybin' ('~' symbolizes the user's home directory) and want the install script to use it then you execute the install script with option

```
> ./install --spath=~/mybin
   ....
   (messages)
   ....
>
```

The directory is now appended to the original search path. Sometimes you need to overwrite an installation present in the search path. Then you must prepend the directory to the search path using a preceding underscore

```
> ./install --spath=_~/mybin
   ....
   (messages)
   ....
>
```

Use the prepend option with care to avoid unwanted command overrides.

Another situation is often found on MacOS. There exists a (non-functional) system version of Tcl/Tk in the directory /usr/bin which should be avoided. You installed a working version of Tcl/Tk into directory '~/Tcl' ('~' symbolizes the user's home directory). Then you must prepend the directory to the search path using a preceding underscore

```
> ./install --spath=_~/Tcl
   ....
   (messages)
   ....
>
```

Again, use the prepend option with care to avoid unwanted command overrides.

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'. The default for DIR is the home directory.

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'. Administrator privilege is needed for the latter command.

On Linux-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) a launcher may be created and placed on the Desktop.

5.3 Support files

For MacOS operating systems, support files are placed into directories '$HOME/Library/NRIXS/MINUTI' or '/Library/NRIXS/MINUTI' for a system-wide install. On Linux-type systems, support files are placed into directories '$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

The MINUTI program package is uninstalled by the command

```
> cd MINUTI -2.3.4
> ./ uninstall
  ....
  ( messages )
  ....
>
```

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

7 How to test

Examples are provided with the MINUTI package. They are located in the 'MINUTI-2.3.4/examples' directory and after installation also in the 'examples' directory as support files. The MINUTI self-test script is located in the directory 'MINUTI-2.3.4/bin' and can be run as follows

```
> cd ~/ MINUTI -2.3.4
> bin/minuti_test  --dir=examples
  ....
  ( messages )
  ....
>
```

For each tested example, the script produces a log file, e.g., the results of a test run in examples/seos_quartz are found in the file seos_quartz.log together with potential deviations that were detected.

The example 'simx_nrixs' requires a working installation of the PHOENIX software. The test script reports a 'failed' otherwise.

7.1 Individual examples

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-2.3.4/examples/seos_quartz
> ls -px
Results/      in_seos        quartz.dat
> seos --help
Usage: seos [OPTION]...
Run MINUTI executable seos-2.3.4

     --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_seos
     --nographics         suppress visualization support
     --quiet              suppress most printout
     --version            display version number and exit
     --vtool              list supported visualization tools
     --vtool=<tool>       use <tool> for data visualization

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

> seos

++ MINUTI-2.3.4 Copyright (C) 2025 Wolfgang Sturhahn
  ....
  (messages)
  ....
-- CPU time : user    0.02 s  system    0.00 s
-- MINUTI module SEOS finished

> ls -px
Results/              in_seos                quartz.dat
quartz_bdp.dat        quartz_bdt.dat         quartz_bma.dat
quartz_bmt.dat        quartz_ccv.dat         quartz_dns.dat
quartz_dsv.csv        quartz_ffm.dat         quartz_fft.dat
quartz_int.dat        quartz_ite.csv         quartz_p02v01_eel.dat
quartz_p02v01_nel.dat quartz_p03v01_eel.dat quartz_p03v01_nel.dat
quartz_p03v02_eel.dat quartz_p03v02_nel.dat quartz_ptl.txt
quartz_rsd.dat        quartz_smv.dat         quartz_tgp.dat
quartz_thx.dat        quartz_vkk.csv         quartz_vol.dat
quartz_vrd.dat
>
```

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 Get started with the 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 the app doesn't
  start. Defeating the Gate Keeper[1]: 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. Later MacOS versions may have different mechanisms.

- other Unix
  Use Desktop launcher or 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 left-click on thumbnail in lower left or by menu actions in upper right panel.

---

[1]The Gatekeeper mechanism varies for different MacOS versions.