

**NAME**

guile - The GNU Project Extension Language

**SYNOPSIS**

**guile** [-L *DIRECTORY*] [-I *FILE*] [-e *FUNCTION*] [\] [-c *EXPR*] [-s *SCRIPT*] [--] [*SCRIPT*]  
[*ARGs for SCRIPT*]]

Only the most useful options are listed here; see below for the remainder.

**DESCRIPTION**

GNU Guile is an implementation of the Scheme programming language. It extends the R5RS and R6RS language standards, providing additional features necessary for real-world use.

Guile works well for interactive use, basic scripting, and extension of larger applications, as well as for stand-alone Scheme application development.

The **guile** executable itself provides a stand-alone interactive compiler and run-time for Scheme programs, both for interactive use and for executing Scheme scripts or programs.

This manual page provides only brief instruction in invoking **guile** from the command line. Please consult the Guile info documentation for more information, (type **info "(guile)Invoking Guile"** at a command prompt).

**OPTIONS****-L *DIRECTORY***

Add *DIRECTORY* to the front of Guile's module load path.

**-I *FILE***

Load Scheme source code from *FILE*.

**-e *FUNCTION***

After reading *SCRIPT*, apply *FUNCTION* to command-line arguments. Note that *FUNCTION* is evaluated, so, for example, (@ (**my-module**) **my-proc**) is valid here.

\ The "meta switch", used to work around limitations in #! scripts. See "The Meta Switch" in the texinfo documentation for more details.

-- Stop argument processing, and start **guile** in interactive mode.

**-c *EXPR***

Stop argument processing, and evaluate *EXPR* as a Scheme expression.

**-s** *SCRIPT-FILE*

Load Scheme source from *SCRIPT-FILE* and execute as a script. Note that in many cases it is not necessary to use **-s**; one may invoke **guile** simply as **guile** *SCRIPT-FILE* *ARG...*

**-ds** Carry out **-s** *SCRIPT* at this point in the option sequence. Note that this argument must be used in conjunction with **-s**.

**--debug**

Start **guile** with the debugging VM. By default, debugging is on when **guile** is invoked interactively; it is off otherwise.

**--no-debug**

Start **guile** without the debugging VM, even if **guile** is being run interactively.

**--auto-compile**

Compile source files automatically (default behavior).

**--no-auto-compile**

Disable automatic source file compilation.

**--listen[=*P*]**

Listen on a port or socket for remote REPL connections. See the manual for more details.

**--use-srfi=*N,M...***

Load SRFI extensions *N*, *M*, etc. For example, **--use-srfi=8,13**.

**-x** *EXTENSION*

Add *EXTENSION* to the **guile** load extension list.

**-h, --help**

Describe command-line options and exit.

**-v, --version**

Display guile version and exit.

**-q** In interactive mode, suppress loading the user's initialization file, *~/.guile*.

## ENVIRONMENT

### GUILE\_LOAD\_PATH

If `$GUILE_LOAD_PATH` is set before **guile** is started, its value is used to augment the path to search for Scheme files when loading. It should be a colon-separated list of directories, which will be prefixed to the default `%load-path`.

### GUILE\_LOAD\_COMPILED\_PATH

If `$GUILE_LOAD_COMPILED_PATH` is set before **guile** is started, its value is used to augment the path to search for compiled Scheme files (.go files) when loading. It should be a colon-separated list of directories, which will be prefixed to the default `%load-compiled-path`.

### FILES

`~/.guile`

A Guile script that is executed before any other processing occurs. For example, the following `.guile` activates guile's readline interface:

```
(use-modules (ice-9 readline))
(activate-readline)
```

### SEE ALSO

The full documentation for Guile is maintained as a Texinfo manual. If the **info** and **guile** programs are properly installed at your site, the command

```
info guile
```

should give you access to the complete manual.

<http://www.schemers.org> provides a general introduction to the Scheme language.

### REPORTING BUGS

There is a mailing list, `bug-guile@gnu.org`, for reporting Guile bugs and fixes. But before reporting something as a bug, please try to be sure that it really is a bug, not a misunderstanding or a deliberate feature. We ask you to read the section “Reporting Bugs” in the Guile reference manual (or Info system) for hints on how and when to report bugs. Also, include the version number of the Guile you are running in every bug report that you send in. Bugs tend actually to get fixed if they can be isolated, so it is in your interest to report them in such a way that they can be easily reproduced.

### COPYING

Copyright (C) 2010, 2011 Free Software Foundation, Inc.

Permission is granted to make and distribute verbatim copies of this document provided the copyright

notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this document under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this document into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

## AUTHORS

Robert Merkel <rgmerk@mira.net> wrote this manpage. Rob Browning <rlb@cs.utexas.edu> has added to it.

**guile** is GNU software. Guile is originally based on Aubrey Jaffer's SCM interpreter, and is the work of many individuals.