

**Software Manual**  
**for**  
**Unix Z/EVES Version 2.1**

TR-97-6028-01e

Irwin Meisels

Release date: June 1996  
Latest revision date: July 2000

ORA Canada  
One Nicholas Street, Suite 1208  
Ottawa, Ontario K1N 7B7  
CANADA



## Contents

<b>1 Introduction</b>	<b>1</b>
1.1 GUI Requirements . . . . .	1
<b>2 Release Notes</b>	<b>1</b>
<b>3 Installation Procedure</b>	<b>3</b>
3.1 The Toolkit and Section Directories . . . . .	6
<b>4 Using Unix Z/EVES</b>	<b>7</b>
4.1 Using Unix Z/EVES with GNU Emacs . . . . .	7
<b>5 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Support</b>	<b>9</b>
<b>6 Contact</b>	<b>10</b>



# 1 Introduction

This document describes the procedure for installing Version 2.1 of the Z/EVES verification system on a Linux or Solaris system. The distribution format for Unix Z/EVES is a compressed tar archive, normally downloaded using ftp.

This version of Z/EVES can be used in two ways, in server mode and in Z/L<sup>A</sup>T<sub>E</sub>X mode. In server mode, it runs as a server, providing parsing, type checking, and proof service which is accessed through a graphical user interface (GUI). The GUI is included in the Z/EVES distribution and is described in [Saa 99b]. The Z/EVES GUI accepts and displays Z text in Z notation, as described in the Z Reference Manual ([Spi 92]).

In Z/L<sup>A</sup>T<sub>E</sub>X mode, Z/EVES is accessed directly, through a command-line interface provided by Z/EVES itself. In this mode, Z text is accepted and displayed in Z/L<sup>A</sup>T<sub>E</sub>X notation. Also, a GNU Emacs mode is provided for editing Z/L<sup>A</sup>T<sub>E</sub>X text and interacting with Z/EVES.

The Z/EVES distribution includes the Z font created by Richard Jones, Version 1.0.4; this font is ©Richard Jones, 1995.

## 1.1 GUI Requirements

The Z/EVES GUI is written in Python, and requires Python 1.5.2 and Tcl/Tk 8.0.

Earlier versions of Tcl/Tk 8.0 will not work correctly. The sources for suitable versions of Python and Tcl/Tk are available at [http://www.python.org/download/download\\_source.html](http://www.python.org/download/download_source.html) and <http://dev.scriptics.com/software/tcltk/8.0.tml>, respectively. Instructions for building these are included below. Later versions of Tcl/Tk and Python are unsuitable: because of changes to font handling, Z characters are not displayed properly.

The following table gives information about Python and Tcl/Tk versions for systems we have had experience with.

Red Hat 6.1	Red Hat 6.1 includes suitable versions of Python and Tcl/Tk.
Red Hat 6.0	The version of Python that comes with Red Hat 6.0 is 1.5.1, which will not run the GUI correctly. However, the version of Tcl/Tk 8.0 which comes with Red Hat 6.0 is patchlevel 4, which is sufficient.
Solaris	Tarballs containing suitable versions of Python and Tcl/Tk are available with the Z/EVES distribution, or you can download the sources at the above URLs and build the programs.

## 2 Release Notes

Several changes have been made since Version 2.0:

- The GUI can export a specification or paragraph in RTF or Postscript format. (Export of Z/L<sup>A</sup>T<sub>E</sub>X is planned for the next release.)
- The editor in the GUI has a complete palette, and provides “flyover” help for Z symbols.
- The GUI’s import function is more complete, and can now import proof scripts as well as paragraphs. (Proof scripts must appear in a zproof environment to be recognized.)
- Many errors in the GUI have been fixed, including a problem that made saving unreliable.
- The `conjunctive` and `disjunctive` commands are faster and more effective.

- Error recovery has been improved.
- The GUI sets colours more carefully, so that its elements are visible even if an unusual desktop theme is used.

The last major public release of Z/EVES was Version 1.5. The following changes were made between versions 1.5 and 2.0 of Z/EVES.

- A new front end was added to Z/EVES, which implements an API for Z/EVES and allows Z/EVES to be used as a server. Communication between the Z/EVES server and a client is over a TCP/IP socket, in XML.
- Save/restore of Z/EVES state has been implemented in the server mode and GUI.
- A GUI, which uses the new Z/EVES API front end, has been implemented.
- Some changes to the underlying EVES prover were made:
  - `if` expressions appearing in function arguments are “lifted” out of the function application. This allows provides the prover with more opportunities for case analysis.
  - Forward rule patterns have been generalized to be the same as rewrite rule patterns. This new generality is not used much now, but will allow future Toolkit improvements.
- A  $\LaTeX$  command “`\znote`” with one argument has been added; the command and argument are treated like whitespace. This allows informal comments be placed inside formal text.
- The Z/EVES Mathematical Toolkit has been revised. The new Toolkit documentation [Saa 99a] describes the changes.
- Miscellaneous speedups and bug fixes have been made.

### 3 Installation Procedure

1. Download the Z/EVES distribution file via ftp. Make sure that you download the correct version for your system, and that you download the file in binary mode. (For more specific instructions on how to download Z/EVES, please contact us at the address given in Section 6.)
2. Create and go to an installation directory. As long as there is enough space (about 15 MB), it doesn't matter where the directory is located. We will call this directory `<zeves>`.
3. Extract the Z/EVES system files from the distribution file. The tar file has been compressed with compress (for Solaris) or GNU gzip (for Linux); GNU tar will automatically uncompress it if you specify the `z` option:

```
tar xvfz file.tar.Z
```

or

```
tar xvfz file.tgz
```

If you don't have GNU tar, uncompress it to standard output and pipe the result to tar:

```
uncompress -c file.tar.Z | tar xvf -
```

or

```
gzip -cd file.tgz | tar xvf -
```

4. Make a copy of the file `system/z-eves.sh` named `z-eves`. Edit the file `z-eves` to set the `zevesdir` variable to the Z/EVES directory `<zeves>` and to set the `zeves` variable to the name of the Z/EVES executable in the `system/` directory, and move the `z-eves` file to a directory on your search path. Make sure that the `z-eves` file is executable. (The Z/EVES executable is the file named `z-eves-system-lisptype`, e.g., `z-eves-pc-linux-gcl`.)
5. [optional] To use the Z/EVES GUI, you have to install the GUI shell script and Z font:
  - (a) Make a copy of the file `system/z-eves-gui.sh` named `z-eves-gui`. Edit `z-eves-gui` to set the `zevesdir` variable to the Z/EVES directory `<zeves>`, to set the `zeves` variable to the name of the Z/EVES executable in the `system/` directory, and to set the `python` variable to the location of the Python interpreter, and move the `z-eves-gui` file to a directory on your search path. Make sure that the `z-eves-gui` file is executable.
  - (b) The directory `zedfont/` contains the Jones Z font in PostScript Type1 and TrueType formats, and in the Solaris distribution, the directory `zedfontb/` contains a set of bitmap fonts generated from the Jones font. We have found that the PostScript font works with the Linux X server and X font server (XFree86), but only the bitmap fonts work for the X server on the version of Solaris that we have. (The TrueType font should work for X servers and X font servers that have FreeType support, but we have not tried this.)  
In the following instructions, `<zfontdir>` is the full path to either the PostScript font directory (`zedfont`) or the bitmap font directory (`zfontb`). The PostScript font is preferable, but if your X server or X font server can't handle it, you'll have to use the bitmap fonts.

The Z font may be installed as a user font or as a system font. To install the Z font as a user font, run the command

```
xset fp+ <zfontdir>
```

every time the X server is started. This may be done manually, or the command may be added to your X initialization file (typically, `~/.xinitrc`).

Installation as a user font is recommended, because no root privilege is required, and you don't need to know system-dependent details about your X server (e.g., whether or not the X font server is used, and where the configuration files are). However, if you want to install the Z font as a system font, you can (as root) add `<zfontdir>` to the font catalogue (for the X font server) or to the system font path (`/etc/X11/XF86Config` file for Linux) and restart the font server or rehash the X server font list. For Solaris, there is also an OpenWindows demo called `fontadmin` which you can use to add the font.

6. [optional,Solaris] To install the Solaris versions of Python and Tcl/Tk, download `python-1.5.2.tgz` and `tcltk-8.0.5.tgz` from the Z/EVES download directory. These files have been compressed with GNU `gzip`, and if you don't have it, you can download the `gzip` program from the download directory also. As root, go to the directory `/usr/local`, and uncompress and untar the files:

```
gzip -cd python-1.5.2.tgz | tar xvf -
gzip -cd tcltk-8.0.5.tgz | tar xvf -
```

7. [optional] To run the GUI, you need Tcl/Tk 8.0 and Python 1.5.2 with support for Tkinter and threading. If your system does not already include these, or if you are on Solaris and you want to build the programs yourself, it will be necessary to build suitable versions.

- (a) Download the source distribution for Tcl/Tk 8.0 from `dev.scripatics.com` (see Section 1.1 for the full URL).
- (b) Unpack the Tcl/Tk distribution, build, and install it. There is no special configuration to do — `configure`, `make`, and `make install` worked for us.
- (c) Download the source distribution for Python 1.5.2 from `www.python.org` (see Section 1.1 for the full URL).
- (d) Unpack the Python distribution and build it. Python must be configured to use threads before building:

```
./configure --with-thread
make Python
```

- (e) Add the Tcl/Tk module. This is done by editing the file `Modules/Setup` to configure `_tkinter`. Uncomment or edit the following lines, as in the following example, to tell Python where the Tcl/Tk and X headers and libraries are on your system.

```
# *** Always uncomment this (leave the leading underscore in!):
  _tkinter _tkinter.c tkappinit.c -DWITH_APPINIT \
# *** Uncomment and edit to reflect where your Tcl/Tk headers are:
#     -I/usr/local/include \
#     -I/usr/include \
# *** Uncomment and edit to reflect where your X11 header files are:
#     -I/usr/X11R6/include \
#     -I/usr/X11R6/include/X11 \
...
# *** Uncomment and edit to reflect where your Tcl/Tk libraries are:
#     -L/usr/local/lib \
```

```

        -L/usr/lib \
# *** Uncomment and edit to reflect your Tcl/Tk versions:
#     -ltk8.0 -ltcl8.0 \
        -ltk8.0 -ltcl8.0 \
# *** Uncomment and edit to reflect where your X11 libraries are:
#     -L/usr/X11R6/lib \
        -L/usr/X11R6/lib \
...
# *** Always uncomment this; X11 libraries to link with:
        -lX11

```

- (f) Rebuild Python and install it:

```

make
make install

```

The install step creates the new executable `/usr/local/bin/python`.

- (g) If your system already had a version of Python installed, you may now have two versions of Python on your system; make sure that the file `z-eves-gui` refers to the right one:

```
python=/usr/local/bin/python
```

8. [optional] The file `system/z-eves.el` contains Emacs Lisp code for the GNU Emacs Z/L<sup>A</sup>T<sub>E</sub>X and Z/EVES modes. To use these modes, we recommend that you byte-compile the file first. The Z/L<sup>A</sup>T<sub>E</sub>X and Z/EVES modes work only with GNU Emacs Version 19 and above. To compile and install this file, do the following:

- (a) Go to the `system/` directory and run Emacs on the file `z-eves.el`:

```

cd system
emacs -batch --eval '(byte-compile-file "z-eves.el")'

```

You will get some warnings about assignments and references to free variables; these may be ignored. You can also edit `z-eves.el` with Emacs and byte-compile it from within the editor with `M-x byte-compile-file`; do this if the nested quotes in the above command confuse your shell.

- (b) Move the `z-eves.el` and resulting `z-eves.elc` files to your Emacs site-lisp directory.  
(c) Add the following to your `~/.emacs` file:

```

(setq auto-mode-alist (append auto-mode-alist
                              '(("\\.z$" . z-latex-mode)
                                ("\\.zed$" . z-latex-mode))))
(autoload 'z-latex-mode "z-eves" "Z-EVES LaTeX mode." t)
(autoload 'run-z-eves "z-eves" "Run Z-EVES." t)
(setq z-eves-program "z-eves")

```

This assumes the shell script `z-eves` has been moved to a directory on your search path. If not, specify the full path name of the shell script file in the string in the last line.

If your source file has an extension of `.tex`, you will have to turn on Z/L<sup>A</sup>T<sub>E</sub>X mode after reading it, or add the cons `(\\".tex$" . z-latex-mode)` to your mode alist to turn on Z/L<sup>A</sup>T<sub>E</sub>X mode for these files.

### 3.1 The Toolkit and Section Directories

**NOTE:** Sections are fully implemented only in Z/L<sup>A</sup>T<sub>E</sub>X mode. The Z/EVES server starts up with the Toolkit section loaded, and it is not possible to replace this section or load other sections.

The directory `library/` contains the section file for the Z/EVES Mathematical Toolkit. Z/EVES starts with the Toolkit already loaded, but if Z/EVES needs to load the Toolkit again, it must know where to find the section file.

Z/EVES looks for section files in the following directories, in the following order:

1. Directories in the “system” library directory list, which is passed to Z/EVES by the `z-eves` shell script. This is a colon-separated list of directories; Z/EVES searches each directory in the list, in the order specified. In the distributed `z-eves` script, this list contains only the directory `$zevesdir/library/`, and can be changed by changing the definition of the `zlibdir` variable in the script.
2. Directories in the “user” library directory list, which is specified in the `zsection path` command. Z/EVES searches each directory in the list, in the order specified.

Directory names must be specified with a trailing slash, in both the `z-eves` shell script, and in the `zsection path` command.

If the `zsection path` command is given with no arguments, the current system and user directory lists are printed.

## 4 Using Unix Z/EVES

To run Z/EVES in Z/L<sup>A</sup>T<sub>E</sub>X mode, run the Z/EVES shell script `z-eves`. Z/EVES will type an introductory message, and present you with its top-level prompt `>`. Z/EVES is now in “interactive” mode, and accepts declarations and commands. To abort a proof command that is taking too long or to stop Z/EVES when printing a too-long formula, use the Unix interrupt key `^C` (control-C). Z/EVES will stop whatever it is doing and return to the top level.

To run the Z/EVES GUI, run the Z/EVES shell script `z-eves-gui`. Chapter 2 of the User’s Guide [Saa 99b] describes how to use the GUI.

To use Z/EVES effectively, you will need to have a good knowledge of the Z language and the syntax of Z/EVES input. The Z reference manual [Spi 92] describes the former, and the Z/EVES reference manual [Mei 97] describes the latter. When proving theorems and domain checking conditions, you will find the theorems described in the documentation for the Z/EVES Mathematical Toolkit [Saa 99a] useful.

### 4.1 Using Unix Z/EVES with GNU Emacs

There are GNU Emacs major modes for editing Z/EVES source and for running Z/EVES. These modes are available if you have installed the `z-eves.el` Emacs Lisp file and modified your `~/.emacs` file appropriately (Section 3). When editing a file whose name ends in `.z` or `.zed`, the buffer will automatically be placed in Z/L<sup>A</sup>T<sub>E</sub>X mode.

The following commands are available to start Z/L<sup>A</sup>T<sub>E</sub>X mode in a buffer, and to start an inferior Z/EVES process.

```
M-x z-latex-mode  start Z/LATEX mode
M-x run-z-eves    start an inferior Z/EVES process
```

The following commands are available in Z/L<sup>A</sup>T<sub>E</sub>X mode:

```
C-c C-l  start an inferior Z/EVES process
C-c C-b  move to the previous command or declaration
C-c C-f  move to the next command or declaration
DEL     delete previous char; if a tab, convert to spaces first
C-j     insert a newline and indent to previous line
TAB     insert a tab, or complete symbol
```

For a Z paragraph to be recognized by Z/L<sup>A</sup>T<sub>E</sub>X mode, the `\begin` and `\end` text must start in the first column of a line. The text of a command must also start in the first column of a line.

When an inferior Z/EVES process is started, a window connected to the process is popped up, and the following additional commands are made available in Z/L<sup>A</sup>T<sub>E</sub>X mode:

```
C-c C-e  send the declaration or command which starts at or before
         point to Z/EVES
C-c C-r  send declarations and commands in the current region to
         Z/EVES
C-c C-s  insert a proof summary at point
```

The following command is available in the Z/EVES process window:

```
C-c C-c  interrupt Z/EVES
```

If the `C-c C-r` command is given a numeric argument (using the `C-u` prefix), only declarations and non-proof commands in the region will be sent to the Z/EVES process.

Output from the form(s) sent to the Z/EVES process by `C-c C-e` and `C-c C-r` will appear in the Z/EVES window. If the `C-c C-l` or `M-x run-z-eves` command is given while a Z/EVES process

is running, the Z/EVES window will be popped up on the screen if it is not currently visible. You may also move to the Z/EVES window and type Z/EVES commands there; it is a “comint window” and the commands available in comint mode are also available in the Z/EVES window.

If you have used the `ztags` command to generate a tags file for your Z/L<sup>A</sup>T<sub>E</sub>X source file(s), the tags-related Emacs commands, (e.g., `M-x tags-search`) may be used, but note the following:

- The current buffer must be in Z/L<sup>A</sup>T<sub>E</sub>X mode when the tags table is loaded. If not, you will have to kill the tags table buffer and reload the table.
- To search for a name or symbol defined in a `\syndef` declaration, enclose the name or symbol in double quotes (`"`), to distinguish it from the name defined in a declaration box.

The Z/EVES process may be terminated by exiting Z/EVES and killing the Z/EVES window.

## 5 $\text{\LaTeX} 2_{\epsilon}$ Support

The Unix Z/EVES distribution includes a  $\text{\LaTeX} 2_{\epsilon}$  package, `z-eves`, for typesetting Z/ $\text{\LaTeX}$  specifications. This package consists of the Z macros from the `zed-csp` package written by Jim Davies, plus macros for typesetting the Z/ $\text{\LaTeX}$  extensions to Z (e.g., theorems).

To install this package, copy the file `doc/z-eves.sty` to your site-local  $\text{\LaTeX}$  package directory. You may also have to run the `texhash` program so  $\text{\LaTeX}$  will be able to find this file.

## 6 Contact

If you have any problems with the system, contact us at:

Postal mail:

Z/EVES  
c/o ORA Canada  
One Nicholas St., Suite 1208  
Ottawa, Ontario K1N 7B7  
Canada

Electronic mail:

`eves@ora.on.ca`

World Wide Web:

`http://www.ora.on.ca/`

There is a mailing list (email) for Z/EVES-related announcements and news, and for general discussion about Z/EVES and Z. The list is a good way to keep in touch with the Z/EVES developers and with Z/EVES users.

- To subscribe to the list, send an email message to `zeves-request@ora.on.ca`, with no subject and the word **subscribe** in the message body.
- To unsubscribe from the list, send an email message to `zeves-request@ora.on.ca`, with no subject and the word **unsubscribe** in the message body.
- To post a message to the list, email the message to `zeves@ora.on.ca`. *Please* don't send subscription/unsubscription requests to this address.
- The Z/EVES mailing list is implemented with Majordomo. For general information on Majordomo functionality, send an email message to `majordomo@ora.on.ca`, with no subject and the word **help** in the message body.
- For help with problems, send an email message to `postmaster@ora.on.ca`.

## References

- [Mei 97] Irwin Meisels and Mark Saaltink. *The Z/EVES Reference Manual*. ORA Canada Technical Report TR-97-5493-03d, September 1997.
- [Saa 99a] Mark Saaltink. *The Z/EVES 2.0 Mathematical Toolkit*. ORA Canada Technical Report TR-97-5493-05b, October 1999.
- [Saa 99b] Mark Saaltink. *The Z/EVES 2.0 User's Guide*. ORA Canada Technical Report TR-99-5493-06a, October 1999.
- [Spi 92] J.M. Spivey. *The Z Notation: A Reference Manual*, 2nd Ed. Prentice Hall, 1992