Emacspeak User's Guide

Jennifer Jobst

jobst@us.ibm.com

December 4, 2001

Revision History

Revision 1.2	December 3, 2001	Revised by: JEJ
Changed license to GFDL		
Revision 1.1	November 12, 2001	Revised by: JEJ
Revision 1.0 DRAFT	October 19, 2001	Revised by: JEJ

This document helps Emacspeak users become familiar with Emacs as an audio desktop and provides tutorials on many common tasks and the Emacs applications available to perform those tasks.

Table of Contents

Table of Contents

1. Legal Notice

2. Introduction

2.1. What is Emacspeak?

Many users are familiar with Emacs as a text editor and have used it extensively in that capacity. But Emacs is not just a text editor −− it is actually a desktop, in some ways like the graphical desktops that many use today. As a desktop, Emacs comes with all kinds of built−in functions, much like the Windows " desktop, including an e−mail application, calendar/appointment program, cd−player, games, and more. And like Windows ", there are many additional applications that you can download and add to Emacs that expand its functionality −− web browsers, the LaTeX text editor, mp3 players, and others. Unlike Windows, all of these powerful tools are under the GPL (Gnu Public License), so you can use them free of charge.

For visually impaired users, adding Emacspeak to Emacs might be compared to adding Jaws to Windows, except that instead of simply reading the screen to you as a standard "screenreader" might do, Emacspeak treats speech as first−class output. Because Emacspeak interacts directly with Emacs instead of just being an add−on, it provides much more context−specific information about what is going on than a typical screenreader would. In addition, there are many special commands just for Emacspeak that enhance interaction with a variety of Emacs applications.

2.2. About this tutorial

Because Emacs and Emacspeak are unlike any environments you may have used before, it helps to know a little bit about how they work. This tutorial assumes that you have either used and are familiar with Emacs and Emacspeak or that you have completed some of the Emacs and Emacspeak tutorials. For a list of recommended tutorials, please refer to Section 3.1.

This tutorial is organized by task and sub−task. For example, the Internet section contains sub−sections on browsing the Internet, using email, chatting online, and other tasks. The File Manipulation section contains information on downloading and installing files, finding files, etc. For a complete listing of the available tasks, refer to the table of contents.

Within this tutorial, you will find references to a number of Emacspeak−enabled applications, some that are included within the Emacs application, and some that are add−ons and must be downloaded. Please note that the applications listed in this tutorial should not be considered a complete collection of applications but only a small subset. The complete list of Emacspeak−enabled applications is available at http://emacspeak.sourceforge.net/applications.html.

3. Before you begin

3.1. Getting started with Emacs and Emacspeak

Before you get started with Emacs and Emacspeak, you must install both. Most distributions come with Emacs, or you can download it from the Emacs home page at http://www.gnu.org/software/emacs/emacs.html. If you have not already installed Emacspeak, you can download it from http://emacspeak.sourceforge.net/. For Emacspeak installation information, please refer to the Emacspeak Installation HOWTO available at the Emacspeak home page, http://emacspeak.sourceforge.net.

If you've never used Emacs or Emacspeak before, you'll probably want to try some of the following tutorials:

- "A gentle introduction to Emacspeak," by Gary Lawrence Murphy,http://emacspeak−guide.sourceforge.net/tutorial.html/. Gary provides a lighthearted but thorough introduction to Emacs and Emacspeak, focusing on users who are not familiar with either application. This introduction is recommend as a starting point, even before you do the Emacs tutorial (see next item).
- "The (Official) Emacs Tutorial," by the Free Software Foundation. I call this the "official" tutorial because it is included with Emacs. To access this tutorial, start Emacs, then type **C−h t** (Control h t). This tutorial discusses the basic Emacs commands, including navigation, using Emacs as a text editor, and a bit about how Emacs works "under the hood." For new users this tutorial is strongly recommended, and even those users who are familiar with Emacs might find something they didn't already know.
- "Emacspeak Tutorial," by Nita Van Zandt, available at http://www.mv.com/ipusers/vanzandt/emacspeak−tutorial−1.0.tar.gz. This tutorial helps new Emacspeak users get up and running and includes plenty of examples and step−by−step instructions. Once you've gotten a grasp of Emacs, this tutorial is a must.
- "The Emacs Beginner's HOWTO," by Jeremy D. Zawodny. It is available at http://www.linuxdoc.org/HOWTO/Emacs−Beginner−HOWTO.html and is also bundled into some of the newer Linux distributions. The Beginner's HOWTO discusses many of the topics discussed in the Emacs tutorial, and also includes overviews of some of the more popular Emacs packages, including VM, Gnus, BBDB, and AucTeX (some of which are discussed in this HOWTO).

Finally, you might want to peruse http://www.gnusoftware.com/WebRing/zone.cgi?list, which provides a list of Emacs resource sites.

3.2. Emacs Command Conventions

Emacs commands are denoted by holding down either the **Control** key or the **Meta** key, then pressing one or more alphanumeric keys. On today's keyboards, the **Alt** key provides the same functionality as the old **Meta** key, or alternatively you can press the **Esc** key followed by the alphanumeric keys (you don't have to hold down the **Esc** key). Because different documents sometimes use different notation to denote these sequences, this section explains the notation used in this HOWTO. For consistency, this HOWTO uses the same notation as is used in the Emacs menus.

There are a few different types of key sequences you may see. The first is commonly written in the form **C−x C−s**. The letter "C," followed by a hypen, literally means "hold down the Control key" (the letter "M" would denote the Meta (Alt or Esc) key), and the additional letters denote specific commands. This example, which saves the current file, should be interpreted as, "Hold down the Control key, press the letter x followed by the letter s, then release the Control key."

Another common sequence is written in the form **C−x d**. Again, the letter "C" followed by a hyphen denotes holding down the Control key. However, because the second letter of the sequence does not have a "C−" in front of it, you do not hold down the Control key while pressing the second letter. Thus, this command should be interpreted as, "Hold down the Control key and type x, then release the Control key and type d." Sometimes you may see commands with more than one letter after them, such as **C−e d w**. The same rules apply: you should hold down the Control key, press "e," release the Control key, then press "d" and "w".

Finally, you may see commands that use actual words, such as **M−x emacspeak−toggle−word−echo**. Fortunately, most of these longer commands have a shorthand command (in this case, **C−e d w**. However, should you opt to type the entire command, you would hold down the Meta or Alt key, type x, release the Meta key, then type emacspeak−toggle−word−echo.

3.3. Getting Help

There are several different types of help you can get from within Emacs. If you are unfamiliar with Emacs or Emacspeak, refer to Getting Started with Emacs and Emacspeak in Section 3.1 for information on several tutorials that can help get you up and running. If you've completed those tutorials and are looking for more in−depth help on specific topics, consider the following sources.

3.3.1. Getting help in Emacs

The most comprehensive source for Emacs is the GNU Emacs Manual, available at http://www.delorie.com/gnu/docs/emacs/emacs_toc.html. Not only does this document contain information on using Emacs as a text editor, but it also contains information on how to use many of the Emacs applications.

In addition, there are many help options within Emacs, all of which can be accessed using **C−h**. Some of these options, and the types of help they provide, are listed below.

C−h i

Invokes the online hypertext help system, also called "info." Not only does info contain Emacs−specific help, it also contains the man page help (refer to Section 3.3.3, Viewing the Linux online man pages, for more information). When in info, you can type **h** to open a primer for first−time users. Type **q** to quit.

C−h a

Runs "apropos" and asks for a word to search on. It then gives you a list of all commands that contain that word.

C−h C−f

When you enter a command name, info jumps to the documentation for that command.

C−h c

When you press any key or key sequence, info provides the name of the function that key invokes.

C−h k

Similar to the **C−h c** command but provides more detailed information about the function or action that occurs when you press a key or sequence of keys.

C−h w

Similar to the **C−h c** and **C−h k** commands, except that when you type in the name of a command, it returns the key sequence needed to invoke that command. For example, if you type **C−h w**, then finder−by−keyword, it will return **C−h p**.

C−h p

Lists all the packages available in Emacs. Packages are defined by the applications they contain, for example hypermedia, games, tools (for programming), etc.

C−h n

Opens a history of user−visible changes to Emacs.

C−h F

Displays the Emacs frequently asked questions. Should you have questions about or problems with Emacs, you are urged to review the FAQ in an effort to cut down on noise in the Emacs−related newsgroups. Note that this FAQ is only for Emacs, and not Emacspeak.

3.3.2. Getting help on Emacspeak commands

When you install Emacspeak, additional Emacspeak−specific help is also installed. In general, Emacspeak commands begin with **C−e**. The more commonly used help commands are listed below:

C−h C−e

Lists the Emacspeak commands that are available.

C−e F

Opens the Emacspeak FAQ.

C−e C−h

Enters a special mode where every key stroke you type is spoken but nothing actually happens. You can use this mode if you're new to Emacspeak and want to test out a few keystrokes. When you're finished, you can exit this mode using **C−g**.

In addition to the general Emacs commands that are Emacspeak enabled, many applications have their own Emacspeak−enabled commands. To get a list of these commands, invoke the application of choice (for example, W3 for viewing web sites) and then type **C−h k** for a list of key bindings, or **C−h m** to list the commands associated with the cirretn active modes.

3.3.3. Viewing the Linux online man pages

Linux comes with an extensive manual, better known as the "man" pages. These pages cover just about every Linux command. The only drawback to using the man pages is that you must know the name of the command you want information about. To read a man entry:

1. From inside Emacs, type **M−x man**. You will be prompted for the manual entry you want to view.
2. Type the manual entry name (for example, **mount**). The entry for that option will open in Emacs.

For more information on man pages themselves, you can type **man** when prompted for the manual entry name.

---

4. System Administration

This section contains information on common system administration tasks. It is by no means an exhaustive list of what goes on in Linux system administration but rather a collection of a few common tasks that a new user may find useful. If you would like more information on Linux system administration, refer to the online Linux System Administrators' Guide, found at http://www.linuxdoc.org/LDP/sag/book1.html.

4.1. Changing your password

Every time you log onto a Linux system, you must provide a password for your account. Occasionally you may wish to change your password, and many systems require that you do so regularly for added security.

Changing your password is done in the Emacs shell. Normally, changing your password in the Emacs shell displays it on the screen and also speaks it aloud. However, Emacspeak adds a line to your `.emacs` file that keeps Emacs from echoing your password either visually or orally. Of course, it's still a good idea to change your password in a secure area where no−one is likely to watch you type it in. To change your password:

1. From inside Emacs, type **M−x shell** to launch an Emacs shell.
2. At the prompt, type **passwd**. You will be prompted for both your current password and your new password.

Note that some systems require passwords to be a certain length or contain certain characters such as numbers. If your password entry is rejected for some reason, keep trying. In general, most 6−8 letter passwords that contain at least one number work well. Of course, be sure it's easy for you to remember!

4.2. Installing applications

4.2.1. About Emacs applications

As an Emacs user, you can mentally divide applications into two categories: those that are native to Emacs (that is, those applications written specifically for Emacs), and those that are not (for example, command−line applications). If an application is not Emacs−native, you can still run it from the Emacs shell command line. To run the application, type the command **M−x shell** to start the shell then the appropriate command to start the application. If the application is native to Emacs, you can type **M−x name−of−application** to start the application. The application will run directly in the Emacs buffer.

Sometimes, even though an application wasn't designed specifically for Emacs, you can install a special wrapper file that makes Emacs think the application is Emacs−native. These special files have a `.el`, or Emacs−lisp, extension. After you've downloaded and installed a new application, if a `.el` file exists for it, you can download that file too. Place it in the correct directory, add the appropriate line to your `.emacs` file so that Emacs will recognize the application, and you'll be able to run the application as if it were native to Emacs.

As an Emacspeak user, you'll also want to know which applications are Emacspeak−enabled. These Emacspeak−enabled applications have a second special `.el` wrapper file that contains Emacspeak−specific commands that provide audio feedback just for that application. For a complete list of Emacspeak−enabled applications, see http://emacspeak.sourceforge.net/applications.html. All avaliable `emacspeak-*.el` files are included with Emacspeak, so you won't have to download them.

So, when you install a new application, you'll need the following three things:

- The application.
- A `.el` file for that application, if you want to run it directly from Emacs instead of from the Emacs shell command line.
- An `emacspeak-*.el` file for the application, if you want to run it directly from Emacs and have it Emacspeak−enabled. Check in the /emacspeak/lisp directory to see if one exists for your application.

Whenever possible, this HOWTO tells you specifically whether or not an application is Emacs−native, and if not, where to get the special `.el` wrapper files if they are available.

4.2.2. Downloading your application

Once you've chosen a new application to install, you'll need to download it. Typically you'll use either FTP application or the wget application to download your new Linux application, since most of them are distributed only on the Internet. If you are unfamiliar with the FTP and wget applications, refer to Section 5.1, Downloading files, for information on how to use these applications to get your files.

Usually, the files you download will have some sort of extension (the letters after the "dot" or "period" in the file name) that denotes what type of file it is. If your file has a .gz or .tar extension, refer to Section 4.2.3 on uncompressing files. If your file has a .rpm extension, refer to Section 4.2.4 on installing RPMs.

4.2.3. Uncompressing files

Most files are compressed in at least one (and sometimes multiple) ways. You can tell how a file was compressed, and thus how to uncompress it, by the extensions on the file. The most common extensions are .tar and .gz, though there are others.

If your downloaded file has multiple extensions, you'll have to uncompress it one step at a time, uncompressing the last extension first. For example, if the file is named `foo.tar.gz`, complete the following steps:

1. Start the Emacs shell using the command **M−x shell**.
2. If necessary, use the **cd** (change directory) command to move to the directory that contains your downloaded file.
3. At the Emacs shell command prompt, type **gunzip foo.tar.gz**. The file will be unzipped, leaving you with a file called `foo.tar` in your directory. Files with a .tar extension are commonly known as "tarballs."
4. At the command prompt, type **tar −xvf foo.tar**. The tarball will be uncompressed, usually leaving either a single file or a new directory that contains a number of files.

   Some recent versions of the **tar** command will both uncompress and untar a file if you add the **−z** option. Thus, the command **tar −xvfz foo.tar.gz** would both uncompress and untar the file.

If you ended up with a new directory after you uncompressed your files, it probably contains files with names like `INSTALL`, `NOTES`, or `README`, among others. These files usually include instructions for installing that specific application. As each application may have a different installation process, you should read these files carefully and follow any instructions they may provide. You'll probably need to be the root user to actually install any applications, as only root usually has the necessary permissions.

If you ended up with a single file after uncompressing, it probably has either a .rpm extension or some other type of extension, such as .el. If it has a .rpm extension, refer to Section 4.2.4 on installing RPMs. If the file has some other extension, refer to the website where you downloaded it for more information.

4.2.4. Installing RPMs

If you downloaded or uncompressed a file that has a .rpm extension, you're in luck! It is a Red Hat Package Manager file that contains all the information it needs to install itself automatically. Assuming that your distribution supports RPMs, you can install the file as follows:

1. Open a root shell using the command **M−x emacspeak−root**, which is bound to **C−e C−r**.
2. At the command prompt, type **rpm −xfv foo.rpm**. A number of messages will scroll by, and you'll be returned to the command prompt once RPM finishes installing the files.

For additional information on RPMs, refer to Red Hat's official website, http://www.redhat.com or the official Red Hat Linux reference guide, at http://www.redhat.com/support/manuals/RHL−7−Manual/ref−guide/ch−rpm.html.

---

5. Working with files

In addition to performing common file manipulation tasks, such as moving, copying, and deleting, you'll also want to download files from the Internet, search for files in your directories, and FTP files. Some of these functions are built into Emacs, while some of these tasks require additional applications that can be downloaded.

5.1. Downloading files

Once you know the URL of the file to download, you can use the **wget** command from within the Emacs shell to retrieve the file. If you do not know the URL of the file to download, use the W3 application to browse the web and find the URL of the file (refer to Section 6.1 for information on using the W3 application). Then use the command **wget** to download it. For more information on the **wget** command, refer to the **wget** man page.

To download a file:

1. From inside Emacs, type **M−x shell** to start the shell.
2. Since the **wget** command places the downloaded file into your current directory, change directories to the desired directory.
3. When you are in the appropriate directory, type **wget [url]**, then press Enter. Remember that you can copy and paste the URL of the file to download from the W3 web browser application.

As the file is being downloaded, one dot appears on the screen for each kilobyte of data received. The **wget** command lets you know when the download is complete.

Another option for downloading (or uploading) a file is via FTP. Refer to Section 5.3 on FTPing a file for more information.

5.2. Finding a file

There are several ways you can find a file. This section discusses two options.

If you know the name of the file, or at least remember part of it, you can use the **find−file** command:

1. In Emacs, type **M−x find−file**.

   Result: You will be prompted for the name of the file in the default working directory.

2. Type the name of the file (or use wildcards) you wish to find.

Alternatively, you may want to browse your directories to look for a file. You can do this using the Emacs directory editor, dired. To start dired:

1. In Emacs, type **C−x d** or **M−x dired**. You will be prompted for the name of the file. You can type the name (or part of the name) of the file to find, using the Tab key to complete the file name. dired will

list any files that match the name you entered. Alternatively, you can view a full listing of the directory by providing a directory name (or pressing Return to view the current directory) when prompted.

For more information on using the directory editor, refer to Section 5.4.2. For a complete listing of dired's features and capabilities, refer to http://www.delorie.com/gnu/docs/emacs/emacs_360.html.

5.3. FTPing a file

Emacs contains a built–in file transfer protocol, or FTP, application. The FTP application can be used to upload and download files to and from other machines. In order to use FTP, you will need to know the hostname of the machine to contact.

To invoke the FTP application:

1. Within Emacs, type **M−x ftp**.

   Result: You will be prompted for an ftp hostname.

Alternatively, you can start an Emacs shell using **M−x shell** and use FTP from the command line. There is a short tutorial on using the command line FTP at http://unix.about.com/library/weekly/aa121800a.htm?once=true&.

5.4. Manipulating files

This section contains information on moving, copying, and deleting files from within Emacs. In general, there are two ways you can go about these tasks. One option is to use the command line from an Emacs shell, and the other is to use dired. Both methods are discussed in this section.

5.4.1. Using the command line

If you are familiar with Unix command–line arguments, you may prefer to use the command line to manipulate files. To use the command line, open an Emacs command shell using **M−x shell**. The following list shows some basic file manipulation commands and what they do:

cp ***`file1`*** ***`file2`***

Copies file1 to a new file named file2.

mv ***`source`*** ***`target`***

Moves a file from the source directory to the target directory.

rm ***`file`***

Deletes a file.

mkdir ***`directoryname`***

Creates a directory with the name `directoryname`.

rmdir ***`directoryname`***

Deletes the directory `directoryname` if it is empty.

ls

Lists all the files in the current directory.

You can find more information on any of these commands (most can be run with various options that allow for more user control) by looking them up in the online man pages. Since the list presented here covers only the most basic commands, you may want to consult the Linux user's guide, available at http://www.ibiblio.org/pub/Linux/docs/linux−doc−project/users−guide/, for more information.

5.4.2. Using the directory editor

Although many Linux users use the command line to manipulate files, Emacs provides a powerful alternative, called dired. Dired is the Emacs directory editor, and is great for manipulating files if you'd rather not use the command line or aren't familiar with it. To start dired:

1. In Emacs, type **M−x dired**. You will be prompted for the directory to open.
2. Type in the directory to display, or press Return to open the default directory.

Once you've started dired, you can move around in the Emacs buffer to have the file names, permissions, owners, sizes, and dates created/edited read aloud.

In dired, some commands mark files for manipulation (for example, you can mark several files, then delete them all), and some commands (such as the copy command) are executed immediately. Note that, in dired, case does make a difference for many commands. Some of the more common dired commands and what they do are listed below:

*press the **Return**key*

Pressing **Return** when the focus is on a particular file opens that file in the appropriate major mode.

d

Marks a single file for deletion, but doesn't actually delete the file. You can mark multiple files for deletion, then delete them all simultaneously.

u

If a file is marked for deletion, this option unmarks it.

x

Deletes all files marked for deletion. You will be prompted to make sure you want to delete the files in question; type **yes** or **no** as appropriate.

C

Copies a file. You will be asked for the name of the new file to create when you copy the old file.

R

Renames a file. You will be prompted for the name of the new file to create when you rename the file.

Because dired is so extensive, including the full documentation for it here is not feasible. However, complete instructions can be found in the GNU Emacs Manual at http://www.delorie.com/gnu/docs/emacs/emacs_360.html.

---

6. Working online

This section explores many of the tasks you would commonly perform using the Internet, including browsing the web, chatting online, reading e−mail, etc.

6.1. Browsing the Internet

The best way to browse the Internet from within Emacs is using W3. W3 is a full−featured web browser written just for Emacs. It does not come with Emacs, but it can be downloaded from ftp://ftp.xemacs.org/pub/xemacs/emacs−w3/w3.tar.gz. The latest release of W3 is version 4.0, and you can get more information about it from the W3 at http://www.cs.indiana.edu/elisp/w3/docs.html.

Once you've downloaded W3, you'll need to perform the following tasks to install it:

1. Open an Emacs shell as root, using the command **M−x emacspeak−root**.
2. Use the **cd** command to change to the directory where the `w3.tar.gz` file is.
3. Uncompress the `w3.tar.gz` file, using the following commands:

    a. At the root command prompt, type **gunzip w3.tar.gz**. You'll be left with a file called `w3.tar`.
    b. At the root command prompt, type **tar −xvf w3.tar**. You'll be left with a directory called `w3−4.0pre.46`.
4. Change directories to the `w3−4.0pre.46` directory.

    Note that the next several steps are also covered in extensive detail in the INSTALL file included in the `w3−4.0pre.46` directory. If you'd like more detailed instructions, please refer to that file.

5. At the command prompt, type **/.configure**. Your makefile will be configured as is appropriate for your system.
6. When your machine has finished creating the makefile, type **make install** at the command prompt. The application will be compiled so that it can be executed.

    W3 is a native Emacs application. In addition, W3 has already been speech−enabled, and the `emacspeak−w3.el` file that provides speech was pre−installed with Emacspeak into your `/emacs/site−lisp/emacspeak/lisp/` directory.

7. Once your machine has finished making the application, open your `.emacs` file (located in your home directory) and add the following line:

    ```
    (autoload 'w3 "w3" "Interface for w3 on Emacs." t)
    ```

    Including this line in your `.emacs` file causes W3 to load automatically when you start an Emacs session.

8. Save and close your `.emacs` file when you are finished editing.
9. To start using w3, type **M−x w3−fetch** and press Return.
10. Supply the start URL.

When browsing, you will find files that you want to download. Although it is possible to download files using W3 by pressing "D" when the cursor is over a link, the wget application is a much better way to do downloads. Using the **wget** command to download files is discussed in Section 5.1, Downloading files.

As a web browser, W3 has many functions that can be accessed by typing **M−x w3−** and then pressing the Tab key. Emacs will provide you with a list of options to complete the string.

Some of the common commands for navigating in W3 are listed below:

Return

Pressing **Return** when over a hyperlink follows that hyperlink. Note that if the hyperlink goes to an FTP site, you may get an error.

Tab

Tabs between the various links on the page.

<

Goes to the beginning of the document.

>

Goes to the end of the document.

M−s

Saves the current document to the local disk as HTML source, formatted text, LaTeX source, or binary.

space

Scrolls down in the buffer.

backspace

Scrolls up in the buffer.

q

Kills the buffer.

For a complete listing of W3 commands, refer to the User's Manual located at http://www.cs.indiana.edu/elisp/w3/docs.html

---

6.2. Chatting online

There are any number of IRC (Internet Relay Chat) applications that you can install, two of which are recommended: AOL Instant Messenger for Emacs (called TNT), and ERC.

6.2.1. AOL Instant Messenger for Emacs (TNT)

There's a special version of AOL Instant Messenger (AIM) written just for Emacs, called TNT. The main site for TNT is at http://sourceforge.net/projects/tnt/, and you can download the necessary files from http://download.sourceforge.net/tnt/tnt−2.3.2.tar.gz. Don't download any files from AOL's site, as you'll end up with the graphical version of AIM instead of the Emacs version. In addition to installing TNT, you'll also need to set up an Instant Messenger account with AOL, and you can do so at http://www.aol.com/aim/homenew.adp.

Once you've signed up for an account and downloaded the files, you'll need to complete the following steps:

1. Open an Emacs command shell as root, using the command **M−x emacspeak−root**.
2. Change directories using the **cd** command to the directory containing the downloaded file.
3. Unzip the file using the command **gunzip tnt−2.3.2.tar.gz**, then expand the resulting tarball using the command **tar −xvf tnt−2.3.2.tar**.
4. Change directories into the new `tnt−2.3.2` directory.

   The next several steps are covered in detail in the INSTALL file that can be found in the `tnt−2.3.2` directory. If you'd like more details than are covered here, please refer to that file.

5. Copy all the .el and .elc files into a directory that is in your Emacs load−path. Typically this directory is `/emacs/site−lisp/` (you should be able to see a number of other .el and .elc files in there).
6. Open your `.emacs` file and add the following lines:

   ```
   (setq load−path (cons "/full/path/to/tnt" load−path))
   (load "tnt")
   ```

   When you add these lines to your `.emacs` file, be sure to change the `"full/path/to/tnt"` to the directory where you placed the .el and .elc files.

7. Save and close your `.emacs` file.
8. Restart Emacs so that the changes take place.

To start TNT, type **M−x tnt** and follow the instructions. You'll probably want to read the README file contained in the `tnt−2.3.2` directory, as it contains excellent descriptions of all the commands used by TNT.

6.2.2. ERC

ERC is an IRC (Internet Relay Chat) client written especially for Emacs. The main site for ERC is located at http://sourceforge.net/projects/erc, and you can download and install the appropriate files as follows:

1. Go to http://prdownloads.sourceforge.net/erc/erc.el. Save the page (it is the ERC application) into a new file called `erc.el`.
2. Place the new file in your Emacs load−path, typically in the directory

/usr/share/emacs/site-lisp.

3. Open your .emacs file and add the following line:

```
(require 'erc)
```

4. Save and close your .emacs file.
5. Exit and restart Emacs so that the changes take place. You won't need to download any Emacspeak−enabled .el files, as those are already included in your /emacspeak/lisp directory.

Once you have installed ERC, it can be started from Emacs using the command **M−x erc−select** to select an IRC server.

For more documentation, refer to the erc.el file. It contains installation and usage instructions in the comments at the top of the file.

6.3. Using e−mail

There are multiple mail programs available for e−mail purposes. One that is recommended is VM (View Mail), an Emacs−native application that allows you to do all the things you'd expect of an e−mail application. Alternatively, you can use Rmail, a slightly less sophisticated but built−in e−mail application. Both are speech−enabled and discussed in this section.

6.3.1. VM

Although VM is Emacs−native, it does not come with Emacs. You can download it from the VM homepage at http://www.wonderworks.com/vm. This page also contains links to the VM user's manual, FAQ, and a list of sites where you can download VM, depending on your location. If you want to modifying source code, download one of the sources; otherwise, download one of the binaries as it will be easier to install.

Assuming that you downloaded a binary, it's probably named vm.elc.gz. To install, follow these steps:

1. Open an Emacs shell using the command **M−x shell**.
2. Unzip the file using the command **gunzip vm.elc.gz**. You should be left with a file named vm.elc.
3. Move the file vm.elc to a directory specified in your Emacs load−path, such as /usr/share/emacs/site−lisp/.
4. Open your .emacs file and add the following line:

```
(autoload 'vm "vm" "Start VM on your primary inbox" t)
```

5. Save and close your .emacs file. Then restart Emacs so that the changes will take effect.

If you decide to install the source files instead of the binary, you'll download vm.tar.gz. Installation instructions are in the README file included in the download.

Once you've installed VM, you can start it using the command **M−x vm**.

6.3.2. Rmail

An alternative to VM is Rmail, a built−in Emacs mail reader. As such, you don't need to download or install anything to make it work. To start Rmail, use the command **M−x rmail**.

There is a chapter about Rmail in the GNU Emacs documentation, available at http://www.gnu.org/manual/emacs−20.3/html_chapter/emacs_31.html. This chapter includes all the information and commands you might need to use Rmail.

7. Productivity

Emacs provides many applications that can help you get your work done. From coding to writing a dissertation to scheduling appointments, you can do just about anything from within Emacs.

7.1. Coding in Emacs

Emacs has "major modes" (essentially editors) for Lisp, Scheme, Awk, C, C++, FORTRAN, Icon, Java, Objective−C, Pascal, Perl, and Tcl. You can invoke a major mode by creating and/or opening a file with the appropriate extension. For example, to invoke the c++ major mode, create a file with a `.cpp` extension and then open that file. You can also download additional major modes, such as SGML.

Because of the breadth of coding options, it is beyond the scope of this document to cover them all. However, there is an extensive section in the Gnu Emacs Manual on coding available at http://www.delorie.com/gnu/docs/emacs/emacs_238.html.

7.2. Customizing Emacspeak

Emacspeak contains many options for customization. The most commonly requested options are covered here, but if there's something else that should be included, please do not hesitate to suggest it.

7.2.1. Changing the speech rate

The default Emacspeak speech rate may be too slow or fast for your tastes, so you can customize it in your `.emacs` file. To do so, add the following lines to your `.emacs` file:

```
(setq dtk-default-speech-rate 410)
(setq outloud-default-speech-rate 90)
```

The numerical values in these commands are the speech rate, in words per minute. The examples above are for the Dectalk Express and the ViaVoice Outloud synthesizers.

7.2.2. Auditory Icons for Emacspeak

Ann Parsons wrote a great explanation of the Emacspeak desktop and auditory icons that you can read at http://www.cs.vassar.edu/~priestdo/emacspeak/msg00256.html. In a nutshell, auditory icons provide you with feedback when you perform different tasks in Emacs. You can get audible feedback when you change buffers, quit a program, enter the buffers list, edit C code, etc.

If you want to use auditory icons and ViaVoice is your speech synthesizer, check to see whether or not you have a multi−channel sound card (try playing a CD and a .au file at the same time − if it works, you have a multi−channel card). If your card is not multi−channel, you must install the application stdiom as follows:

1. Download the application stdiosynth from http://www.leb.net/pub/blinux/emacspeak/blinux/stdiom.tar.gz.
2. Log in to your machine as root, then change to the directory where you placed the downloaded file `stdiom.tar.gz`.
3. Unzip the file using the command **gunzip stdiom.tar.gz**.
4. Untar the resulting file using the command **tar –xvf stdiom.tar**.
5. Change directories to the resulting `stdio_musician1.0` directory and type **make**. You should get a message that says, "stdiosynth is up to date."

Currently there are two auditory icon themes that you can download from the Emacspeak website, called Chimes and Cartoons. Chimes is made up of different chimes and short notes from various instruments. The icons are high–quality 44K–mono, and can be downloaded from http://emacspeak.sourceforge.net/chimes–mono.tar.gz. The Cartoons theme was contributed by Bryan Smart and is made up of 22K–mono sounds. You can download it from http://emacspeak.sourceforge.net/cartoons–mono.tar.gz.

To install either of these themes, download and unpack the archives into the emacspeak/sounds directory in your emacspeak installation. You can then select themes you have already installed by using the Emacspeak command **M–x emacspeak–sounds–select–theme**. Alternatively, you can add the following lines to your `.emacspeak` file:

```
(setq emacspeak-toggle-auditory-icons t)
(when (emacspeak-sounds-theme-p "chimes-mono")
        (emacspeak-sounds-select-theme "chimes-mono/" ))
```

Some people use sound cards that can play multiple channels to produce their auditory icons. For this option, set **emacspeak–aumix–multichannel–capable–p** to **t** in your `.emacs` file:

```
(setq emacspeak-aumix-multichannel-capable-p t)
```

7.3. Reading Adobe Acrobat files

There are two ways to view an Adobe Acrobat PDF file with Emacspeak. The first option is to generate a text version of the PDF using Xpdf, then read the text version. The second option is the use Adobe's PDF to HTML converter, then read the HTML file.

7.3.1. Using Xpdf

To generate a text version of the PDF, you'll need to download the Xpdf package. The main website is at http://www.foolabs.com/xpdf/, and you can download a precompiled binary (unless, of course, you're interested in playing with the source code, in which case you'd download the source) from ftp://ftp.foolabs.com/pub/xpdf/xpdf–0.92–linux2.0.tgz.

Assuming that you downloaded the binary file `xpdf-0.92-linux2.0.tgz`, to install and run Xpdf:

1. Open an Emacs command shell using **M–x shell**.

2. Change to the appropriate directory where the `xpdf-0.92-linux2.0.tgz` file is located, then unzip the file using **gunzip xpdf−0.92−linux2.0.tgz**.
3. Decompress the resulting `xpdf-0.92-linux2.0.tar` file using the command **tar −xvf xpdf−0.92−linux2.0.tar**.

The resulting directory will be called `xpdf-0.92-linux2.0`. Within this directory are a number of applications, including pdfimages, pdfinfo, pdftopbm, pdftops, pdftotext, and xpdf. You can find out what each of these applications does from the README file also included in the directory. The only application we're interested in is pdftotext that you can run from the Emacs command shell as follows:

1. Open an Emacs shell window using the command **M−x shell**.
2. At the prompt, type **pdftotext `filename.pdf`**.

   Result: The file is converted to text. You can read the resulting text file with Emacs.

7.3.2. Using Adobe's converter

As an alternative to Xpdf, you can use Adobe's PDF to HTML converter if the PDF file you want to read has a URL. To convert a file:

1. Use W3 to go to http://access.adobe.com/simple_form.html.
2. Type in the URL of the PDF file to convert.

   Result: The file is converted it HTML. You can use W3 to read and navigate within the file.

Tests show that this conversion works very well on some PDF documents, but not so well on others. If the HTML file you end up with isn't readable, try using Xpdf to convert to a plain text file.

7.4. Scheduling appointments and calendar events

Emacs has a built−in desk calendar that includes a diary for planned events. The calendar and diary support both US and European date formats, but default to the US format. The diary keeps track of appointments and reminders using a diary file, a text file that contains a list of events and their dates. You can have Emacs send you an email every day with your schedule, or you can look at the calendar to view the day's events.

To start the calendar application, use the command **M−x calendar**. Calendar navigation commands are, for the most part, identical to text navigation commands, and you can also use the arrow keys to navigate.

If you'd like to use the diary, you need to create a diary file that contains your events. The diary file is a simple text file, and, by default, Emacs uses `~/diary` as the diary file. Although you'll need to create the file manually, there are commands that allow you to add, view, and change diary entries. Most of these commands are Emacspeak−enabled.

Diary entries come in a number of formats and entries can be set to repeat. Below are some examples:

12/22/2001	Mom arrives for visit	Appears once on 12/22/2001

October 17	Joe's birthday	Repeats yearly on October 17th
Friday	Time cards due	Repeats every Friday

Of course, there are many more options than those shown here. You can find a complete listing of diary features in the GNU Emacs manual, located at http://www.delorie.com/gnu/docs/emacs/emacs_376.html.

You can view the diary entries for the current date using the command **M−x diary**. Alternatively, you can have the day's appointments automatically displayed when you enter Emacs by adding the following line to your `.emacs` file:

```
(diary)
```

You can also have your events emailed to you on a daily basis by running the command **M−x diary−mail−entries**. You'll probably get an error message the first time you do this that says, "No buffer named *Fancy Diary Entries*." This error message was reported as a bug to the bug−gnu−emacs mailing list by Tim Hesterberg. However, the command does in fact work, so you can just ignore the error message. By default, you'll be e−mailed the calendar entries for the next seven days. Also, when you enter Emacs it will automatically show you the calendar entries for the next seven days. According to the GNU Emacs Manual, there is a way to change the number of days the email covers, but according to Tim that feature also has bugs and doesn't seem to work. At the time of writing, these problems have not been addressed.

For complete instructions on using the calendar and diary, refer to the section on the calendar and diary in the GNU Emacs Manual, located at http://www.delorie.com/gnu/docs/emacs/emacs_376.html.

7.5. Writing text in Emacs

Emacs provides for many modes of text editing, and the tools you should use are task−dependent.

7.5.1. Writing for print, PDF, or postscript

If you want to output your writing in print, postscript, or PDF, LaTeX is the way to go. LaTeX is essentially a markup language that produces high−quality print, PDF, or postscript output. LaTeX was designed for the production of technical and scientific documentation, and in addition to allowing for detailed formatting, it also provides support for mathematical functions, automatic generation of bibliographies and indexes, graphics support, and much more. You can find out more about LaTeX from http://www.latex−project.org. In the documentation section at the LaTeX site, you will find an introduction to LaTeX, as well as a complete reference manual.

AUC Tex is the Emacs editing mode for LaTeX. The AUC Tex home page is located at http://mirrors.sunsite.dk/auctex/www/auctex/. You can download AUC Tex from ftp://sunsite.dk/packages/auctex/auctex.tar.gz. Once you've downloaded the file, you'll need to install it as follows:

1. Open an Emacs command shell as root, using the command **M−x emacspeak−root**.
2. Change directories to where the downloaded file `auctex.tar.gz` is located.

3. Unzip the file using the command **gunzip auctex.tar.gz**.
4. Unpack the resulting tarball using the command **tar −xvf auctex.tar**.
5. Change directories into the new `auctex−10.0g` directory, then type the command **make**.

   The next several steps are covered in greater detail in the `INSTALLATION` file included in the `auctex−10.0g` directory.

6. When the Makefile is finished building, type the command **make lispdir=/usr/local/share/emacs/site−lisp install**. If the Emacs path defined as "lispdir" in this command is not correct for your machine, supply the correct path to your Emacs installation.
7. Check to make sure the `tex−*.el` files were placed in the directory specified in the `lispdir` command above. If not, copy them there.
8. Add the following line to your `.emacs` file:

   ```
   (require 'tex−site)
   ```

   When you are finished, save and close your `.emacs` file.

9. To invoke the AUC Tex editing mode, create a new file with a `.tex` extension.

You'll probably also want to install the online documentation. This is located in the `doc` directory within the `auctex−10.0g` directory. To install the documentation:

1. Change directories to the `doc` directory.
2. At the command prompt, type **make**. When the makefile is finished, type **make install**.

   Result: The documentation is installed in the `/usr/local/info/` directory.

You can also get this information in the online manual at http://mirrors.sunsite.dk/auctex/www/auctex/doc/.

Since AUC TeX is a part of LaTeX, you'll need to know LaTeX to use it. There's a good LaTeX manual by David R. Wilkins called "Getting Started with LaTeX" located at http://www.maths.tcd.ie/~dwilkins/LaTeXPrimer/.

7.5.2. Writing for online viewing

If you're writing a document that will be read online, you have several choices: HTML, SGML, and XML, to name a few. Emacs comes with a built−in HTML editor, called html−mode. This editor provides you with additional commands for adding appropriate HTML tags to your document. To start html−mode, type **M−x html−mode**.

If you do a lot of HTML coding, you may wish to use html−helper−mode. This mode has a slightly different interface and many more features than html−mode. The home page for html−helper−mode is located at http://www.santafe.edu/~nelson/tools/. You can either download just the `html−helper−mode.el` file from http://www.santafe.edu/~nelson/tools/html−helper−mode.el, or you can download the entire distribution, including documentation, from ftp://ftp.reed.edu/pub/src/html−helper−mode.tar.gz. Installation instructions for both of these options are located at the html−helper−mode website at http://www.gest.unipd.it/~saint/hth.html.

If you're authoring larger documents or want to author in SGML, the PSGML mode for Emacs is recommended. The home page for PSGML is located at http://www.lysator.liu.se/projects/about_psgml.html.

There's also an XML editor for Emacs, called sxml−mode. The home page is located at http://koala.ilog.fr/plh/sxml.html. You must have a Java Virtual Machine, as well as PSGML 1.0.1 or better to run sxml−mode. Complete instructions for downloading and installing are available on the web site.

---

8. Entertainment

While Emacs provides lots of applications and functionality to help get work done, it also provides a medium to play games, listen to music, and do other fun activities.

8.1. Burning a CD

It's become almost necessary to have a CD burner these days, especially if you want to download iso images to install new versions of Linux. The cdrecord application is the recommended tool for burning both audio and data CDs, and it is included in many distributions. However, if you'd like to get the latest version, go to http://www.fokus.gmd.de/research/cc/glone/employees/joerg.schilling/private/cdrecord.html.

cdrecord is a command−line application, so you'll need to open an Emacs command shell (**M−x shell**) to use it.

The complete directions on how to use cdrecord burn a CD are beyond the scope of this document; however Winfried Trümper wrote an excellent HOWTO on making both data and audio CDs, which is available at http://www.linuxdoc.org/HOWTO/CD−Writing−HOWTO.html. If you are new to burning CDs in Linux, this document is indispensable.

8.2. Playing CDs

The best tool for playing CDs in Emacs is cd−tool. An Emacspeak−enabled version is included with your Emacspeak distribution.

All of the commands that control cd−tool must be prefaced by the command **M−x cd−tool**. After you type this, you will be prompted to enter the command to execute. For example, if you wanted to start playing a CD, you would type **M−x cd−tool**, then at the prompt type **p** (for 'play'). The CD will then start playing.

The commands for cd−tool are listed below:

p

Play the CD.

=

Play the CD in random order (shuffle).

+

Skip to the next track.

−

Return to the previous track.

SPC

Pause or resume play.

i

Show the CD info.

s

Stop playing the CD.

e

Eject the CD (must be stopped).

8.3. Playing mp3s

To play an mp3 music file, you'll need the application mpg123. Although the application is included in a number of distributions, the standard version is not Emacs−native, so you'll need to download an install the Emacs−enabled version. You can download the file `mpg123-el-1.24-1.tar.gz` (the latest Emacs−enabled version) from Debian at http://http.us.debian.org/debian/pool/main/m/mpg123−el/mpg123−el_1.24−1.tar.gz.

Once you've downloaded the file, you'll need to install it using the following steps:

1. Open an Emacs shell using the command **M−x shell**.
2. Change directories to the directory containing the new file.
3. Unzip the file using the command **gunzip mpg123−el−1.24−1.tar.gz**.
4. Untar the resulting `mpg123-el-1.24-1.tar` file using the command **tar −xvf mpg123−el−1.24−1.tar**.
5. Change directories into the resulting `mpg123-el-1.24-1.tar` directory. In this directory, you'll see the `mpg123.el` file. Copy this file into the `/usr/shar/emacs/site-lisp` directory.
6. Open your `.emacs` file and add the following line:
   ```
   (autoload 'mpg123 "mpg123" "A Front-end to
   mpg123" t)
   ```
7. Save and close your `.emacs` file. Then restart Emacs so that the changes will take effect.

To start playing mp3s, use the command **M−x mpg123**. You'll be asked for a file name to play. Type in the name of the mp3 to play, then press Return. It should start playing immediately.

There are a number of commands that can be used with mpg123; a few of the more commonly used ones are listed below:

SPC

Play or pause

RET

Play

V

Volume up

v

Volume down

q

Quit

For a full list of mpg123 commands, you can scroll up and down within the application to have them read aloud.

One note about volume in mpg123: in my tests, I found that often the system volume would go way down when I started mpg123, to the point where I could barely hear it. If I exited mpg123, the volume in Emacspeak would continue to be extremely low, regardless of how it was set before I started mpg123. However, if I turned up the volume while in mpg123, not only was it at an acceptable level in mpg123, it was also loud after I quit mpg123.

Another problem that I noticed when using mpg123 is that sometimes when you quit the application, it doesn't seem to relinquish control of the sound card. You'll see evidence of this behavior if you try to play a CD or restart mpg123 − you'll get an error message saying that your sound card is busy or that it's not installed correctly. If you check the list of open Emacs buffers, you'll see that mpg123 is still there, although it's apparently in a questionable state. I have not yet figured out how to remedy this problem; if anyone has any suggestions, please let me know so it can be added to the HOWTO.

If you find yourself listening to a lot of MP3s, there is an mp3 jukebox written by Karl Dahlke available at http://www.eklhad.net/linux/app/jukebox, and a sample config file for the jukebox at http://www.eklhad.net/linux/app/sample.jukerc. At the time of writing, the jukebox was in version 1.0.

If you want to rip your own MP3s, under Linux it is a two step process. The first step is to convert the tracks on an audio CD to `.wav` format, then convert the `.wav` files to MP3s. There is a great web page by Nathan Robertson on how to do this, located at http://www.nathanr.net/thisout/articles/linux−mp3/.

8.4. Playing games

Emacs has a number of popular games built into it, including Solitaire, Tetris, Dunnet (a text−based adventure game) and Gomoku. For a complete listing of games:

1. Type **C−h p**. A list of the available Emacs packages opens in the buffer.
2. Scroll down to the games entry, then press Enter. A list of all the available games is displayed.

To start a game:

1. Type **M−x**, then the name of the game you want to play. For example, to play Dunnet, the text−based adventure game, you would type **M−x dunnet**. But only do this if you have plenty of time to spare!

9. Acknowledgments

As with any effort for the Linux project, there are a number of people who contributed to this HOWTO in a roundabout way. Without their help, this document would not exist.

Dr. T.V. Raman, who not only suggested a task–oriented HOWTO, but who always answered my Emacspeak questions, no matter how odd.

Mr. James Van Zandt, the original author of the Emacspeak HOWTO, who graciously allowed me maintainership of the original HOWTO from which this document stemmed.

Robert J. Chassell, Saqib Shaikh, Doug Smith, Tyler Spivey, Kristin Thomas, and Jason White, for their comments, suggestions, and excellent proofreading skills.

All the folks on the Emacspeak and blinux mailing lists who contributed suggestions, answered my questions, and gently showed me the errors in my ways.

Throughout this document, I have tried to give credit to other authors when referring to their work. If I missed a citation, please let me know and I will be sure to correct it. If you did the work, you certainly deserve the credit!

---

10. Wish list for version 2.0

The following is a list of suggestions (in no particular order) that I did not have time to implement for this version of the User's Guide. I hope to include these items in a future release.

- New section on remote Emacspeak.
- New section on getting Emacspeak and Speakup to work together.
- New section on using Real Audio.
- New section on the Emacs configuration/customization wizard (M−x customize, Applications group).
- Section on using the enriched package for working with RTF files.
- Information on what to add to your .emacs file for rmail or vm to know your smtp/pop3 settings. Also information on how to tell W3 to work with a proxy server.
- Add information on using M−x rpm to manipulate RPM files.
- Add information on using jide (java ide).
- New sections on using calc and bbdb.
- New section on spell checking using ispell.
- Add information on using M−x tar to manipulate tar files.
- Discuss the outline and folding−mode features of Emacs.
- Include key bindings along with the Emacs/Emacspeak command names.
- List or reference the Emacspeak−specific commands available in various Emacs applications like vm, W3, etc.