

Computronics ZIP (for UNIX)

Table of Contents

How to Use This Manual	1
Section One / ZIP Basics	6
What is Compression?	6
What Do the ZIP Programs Do?.....	6
Section Two / Installing ZIP	7
Before Starting.....	7
ZIP Program Installation	8
Step One / Restore the ZIP Product Directory	8
Step Two / Run the Installation Script	8
Step Three / Create the ZIP.CFG file.....	8
Section Three / ZIP Compression Program	9
Quick Reference	9
Before Starting.....	10
Controlling the Default File Name Mapping	11
Retaining UNIX File Names and Attributes.....	12
Command Options	14
Tutorial.....	31
Step One / Add All Files.....	31
Step Two / Add Two Files.....	32
Step Three / Freshen With Files Matching a Wildcard	32
Section Four / UNZIP Uncompression Program.....	33
Quick Reference	33
Before Starting.....	34
Controlling the Default File Name Mapping	35
Retaining UNIX File Names and Attributes.....	36
Command Options	38
Tutorial.....	47
Step One / Extract All Files.....	47
Step Two / Extract Files Matching a Wildcard	47
Step Three / Overwrite Existing Files.....	48
Appendix A / Creating a Pre-Selected File List.....	49
What is a File List?	49
Tutorial.....	49
Step One / Startup Your Text Editor	49
Step Two / Enter the File Names	49
Step Three / Save the List File.....	49

Appendix B / The Configuration File	50
What is the Configuration File?.....	50
Configuration Options.....	50
Appendix C / Error Messages.....	53
ZIP Error Messages.....	53
UNZIP Error Messages.....	55
Appendix D / Returned Error Codes.....	57
Appendix E / Self Extracting ZIP Files (for DOS)	58
Appendix F / ZIP for Unix and ZIP for PRIMOS	60
Appendix G / Common Problems.....	61

Computronics ZIP for UNIX Documentation

How to Use This Manual

This manual is for Computronics ZIP for UNIX™. It discusses version 3.0 or later of this program. ZIP for UNIX will run on a variety of Unix based systems.

This manual is intended for use by a knowledgeable UNIX user or System Administrator. System Administrators are encouraged to read this manual thoroughly.

One of the advantages of ZIP for UNIX is that it can use ZIP format files which are compatible with those created by PKWARE, Inc.'s PKZIP and PKUNZIP programs, which are widely available for the IBM PC and compatibles. Should you require the use of this feature you must know the procedure for uploading and downloading files between the Unix system and the PC. A communications program must be used to transfer the file; ZIP for UNIX will not transfer the file for you. However, ZIP for UNIX will reduce upload and download times by compressing large data files before transmission.

There are many communications programs which will allow you to transfer files between the PC and the UNIX system. If you need to do this, you should be familiar with the following tasks:

- ✓ Using DOS 3.x or above.

If you are not familiar with the use of a version of MS-DOS Version 3.1 or above, please consult the following manual:

- *IBM Disk Operating System Version 3.10 Reference Manual*, © 1986 International Business Machines Corporation.

- ✓ Transferring a file between the Unix system and a PC using a communications program.

If you have not performed this activity read the manuals for your communications program. Read all pertinent sections in these manuals related to installation and setup, accessing the host system, and transferring files. You should be able to configure the communications program, including specifying these parameters:

- Baud rate
 - Data bits
 - Stop bits
 - Start bits
 - Parity
 - Duplex
 - Flow control
- ✓ Configuring and connecting your IBM PC or compatible to the UNIX system using an asynchronous cable.
- ✓ If you have not performed these activities, consult the technical reference for your IBM PC or compatible. For information on configuring the asynchronous lines for your UNIX system, read the appropriate manuals relating to communication lines and controllers.

Syntax Conventions

Syntax represents the order in which you must type a UNIX command and any options that follow it. Elements that appear in bold type must be typed exactly as they appear in the text; elements that appear in *italic* are placeholders representing specific information that you supply.

Unless specified otherwise, you can type commands and options in either uppercase or lowercase letters. The word "type," as used in this manual, means to press a key, or sequence or keys, and then press the [**Enter**] key. The word "press" means to press a specific key, without pressing the [**Enter**] key.

Other conventions used in this manual are as follows:

<u>Convention</u>	<u>Meaning</u>
[]	Indicates an item that is optional. To include optional information within the brackets, type only the information, not the brackets themselves.
	Separates two mutually exclusive choices in a syntax line, as shown in the following example:

-m[u|f]

Type only one of these choices (that is, **-mu** or **-mf**); do not type the vertical bar (|) itself.

... Indicates that the previous element can be repeated several times in a command. Type only the information, not the ellipsis (...) itself.

filename Specifies the name of a file. UNIX file names use the slash ("/") as a separator between the *path* portion and the final element of the file name. The final element of the file name may be up to 1024 characters (depending on your Unix system, this may be limited to a smaller number, but it will always be at least 14) and contain one or more periods (for example `YOUR.FILE.TXT`).

The names of files within a ZIP file can be up to eight characters long and can be followed by a period (.) and an extension of up to three characters (for example `YOURFILE.TXT`). The *path* portion of a file name is normally not omitted, but when present, is specified as a forward slash (/).

path Specifies the route the operating system is to follow through the directory structure to locate a directory or file. You need to specify a path with a *filename* only if the file is not in the current directory, or you wish to copy a file to a location other than the current directory.

options Specifies one or more optional command parameters or switches. A switch usually begins with a hyphen (-), for example, **-a**.

Denotes special notes or other observations in the documentation.

Surrounds examples of the proper use of various commands and their syntax.

Denotes information that relates to configuration parameter settings, as discussed in Appendix B.

Command Line Format

The ZIP command line has the following basic format:

```
command [options[ ...]] zipfile [filename[ ...]]
```

All ZIP program commands are typed at the UNIX prompt; they can also be invoked from an application with access to a shell.

Note: The command line usage differs slightly from normal UNIX conventions so that the command is compatible with its DOS-based equivalent.

options must be preceded by a hyphen (-) character. Most options, except where noted, can be combined (i.e. **-x -y** or **-xy**). The command *options* can generally be placed anywhere on the command line. Typed information can generally be in upper case, lower case, or a mixture, except where noted in the manual. The following options are case sensitive: **-J -j**, **-W -w**, **-C -c**, **-P -p**, **-A -a**, **-T -t**, **-N -n** and the password for the **-s** option. It is recommended however, to use lower case when possible in order to be compatible with future versions of the programs.

zipfile is the name of the ZIP file to be processed. It is NOT necessary to type the extension **.zip**, it will be assumed. The extension is used in this documentation for clarity.

filename is the name of the file to be added to, or extracted from, the ZIP file. Multiple names may be listed by separating the names with one or more spaces. The default is *****, or all files, when specific files are not named (except for the **-d** delete files option).

A help screen is displayed when you type the program name (**ZIP** or **UNZIP**) at the prompt and then press [**Enter**]. Also typing **-h** on the command line will call up the help screen.

The default area for the temporary files that are created while the original is being updated is the same path and directory location as the original ZIP file. As this can cause a space shortage due to quotas or limited file system sizes, the **-b** option is recommended.

UNIX file names are converted to MS-DOS conventions when stored in a ZIP file. Similarly, when the contents of foreign ZIP files are extracted, illegal file name characters (for UNIX) are mapped to the valid UNIX equivalents. UNIX wildcard conventions (***** and **?**) are used for wildcard processing. Please note that you may need to escape the ***** and **?** characters on the ZIP or UNZIP command lines. If a wildcarded name does not work properly, enclose the name in quotation marks or place a backslash (****) before the special character.

Note that the **-An** option can be used to retain the UNIX file name. Other options allow other UNIX file system attributes, such as the permissions and date/time accessed, to also be retained in the zip'ed file.

Organization of the Manual

This manual contains the following sections:

Section One / ZIP Basics provides an overview of the ZIP programs.

Section Two / Installing ZIP describes how to install the ZIP programs so that you compress and uncompress your files.

Section Three / ZIP Compression Program outlines how to start the ZIP program, describes the command line options, and presents a tutorial on its use.

Section Four / UNZIP Uncompression Program outlines how to start the UNZIP program, describes the command line options, and presents a tutorial on its use.

Appendix A / Creating a Pre-Selected File List describes how you can save a list of files for processing later on.

Appendix B / The Configuration File shows how a system administrator can use a configuration file to set the program defaults for his users.

Appendix C / Error Messages lists the program error messages and their probable causes.

Appendix D / Returned Error Codes discusses the use of ZIP and UNZIP in scripts, and proper error handling codes.

Appendix E / Self Extracting ZIP Files (for DOS) discusses the program that allows you to convert ZIP files into DOS self-extracting executable files.

Appendix F / ZIP for UNIX and ZIP for PRIMOS discusses the cross compatibility features of Computronics' ZIP software running under PRIMOS and under UNIX.

Appendix G / Common Problems answers the most commonly asked questions about ZIP and UNZIP.

Section One / ZIP Basics

This section gives an overview of the ZIP programs and how they can be used.

What is Compression?

The ZIP programs allow you to compress your computer files. The term *compression* means to reduce in size. *Computer file compression* refers to reducing files in size so that they take up less storage space on disk. The compressed files are then stored in a special file called a ZIP file.

There are several advantages to using ZIP files:

1. They use less disk space than normal files. Storing files in compressed format allows you to increase the utilization of your system's disk.
2. The ZIP file format is commonly used on IBM PCs and compatibles, as well as many other systems such as PRIMOS, VMS, Amiga, and Macintosh. The ZIP programs allow your UNIX system to exchange information with these systems.
3. Many individual files can be compressed into a single ZIP file. This makes file group identification, copying, and transporting faster and easier.
4. Compressed files can be more easily transferred between the UNIX system and your PC. This is important when sending a large amount of information over a relatively slow asynchronous connection (typically from 9.2 to 19.2Kbps for PCs directly connected to the minicomputer). Compressed files also travel faster via modem which reduces long distance telephone charges. Many BBS (computer bulletin board services) use ZIP files as their standard. This enables the BBS to store more files and enables you to transfer files faster and more easily.

What Do the ZIP Programs Do?

The ZIP software is comprised of the ZIP and UNZIP utilities.

ZIP is the program that compresses files. This shrinking process is often referred to as data compression. Terms you will see during the compressing process are Storing, Shrinking, Deflating, and Imploding. ZIP also handles all file maintenance including adding and deleting files, as well as reporting on technical information from within the compressed file.

UNZIP is the program that uncompresses or extracts compressed files. In addition to extracting a complete ZIP file, it can selectively release individual files, show files on the screen for fast viewing, or print them out on your system printer.

Section Two / Installing ZIP

This section describes how to install the ZIP family of file compression utilities. There are three steps.

Before Starting

The ZIP utilities produce files which are compatible with the PKZIP and PKUNZIP programs from PKWARE, Inc., the file formats of which are dedicated to the public domain.

The following materials are required for installation of the ZIP programs:

- ZIP for UNIX Program Tape or Floppy
- ZIP for UNIX Reference Manual (this document)
- A UNIX system

If you wish to use the utilities to exchange the compressed ZIP files between the UNIX system and an IBM PC or compatible, you will also need the following: (This will not be needed for Unix to Unix communications only, or for Unix to other platforms)

- A PC workstation with an available COM1 or COM2 port
- A cable properly configured for connecting the serial port of the PC to the Unix
- You will also need the PKZIP and PKUNZIP programs from PKWARE, Inc., or other software which runs under MS-DOS and accepts files compressed using this format. PKWARE, Inc. can be reached at:

PKWARE, Inc.
7545 North Port Washington Road
Glendale, Wisconsin 53217 USA
Phone: (414) 354-8699 (as of October 1991)
BBS: (414) 354-8670
FAX: (414) 354-8559

- A communications program which allows the PC to login to a Unix system and transfer files. Crosstalk, Crosstalk for Windows, PROCOMM, or a number of other packages can be used for this purpose. This can also require software on the Unix system, such as Kermit. Note that binary files will need to be transferred (ZIP files are binary).

Computronics will provide an "as-is" copy of Kermit to your site at no charge, upon request. This can be used with PROCOMM, Crosstalk, or other programs and thus no extra software may need to be purchased, if you have appropriate PC software already.

ZIP Program Installation

Step One / Restore the ZIP Product Directory

The ZIP programs must be installed on your Unix system.

The supplied tape or floppy contains the `zip` directory. Restore the contents of the media into a temporary area. You should read the files in the `INFO` subdirectory before installing the new software, in case special notes are available.

Step Two / Run the Installation Script

Attach to the `zip` product directory that you have restored from tape and run the installation procedure.

```
Enter:      cd zip
           ./install_zip
```

Step Three / Create the ZIP.CFG file

Appendix B discusses the `zip.cfg` file. If you will be altering any of the default settings used by ZIP and UNZIP, create the file `zip.cfg` in the `/etc` directory now.

Section Three / ZIP Compression Program

This section describes how to use the ZIP program. The ZIP program may be used to create and maintain compressed ZIP data files.

Quick Reference

The following table will provide a quick reference guide to all the options of ZIP. The command syntax is also shown below.

Syntax

```
ZIP [options[ ...]] zipfile [@list] [filename[ ...]]
```

<i>zipfile</i>	ZIP file name; default extension is .zip
<i>filename</i>	Names of files to compress; wildcards * , ? are ok; default is all files
<i>list</i>	An optional list file name. See Appendix A for information on how to use a list file.
<i>Options</i>	An option list, containing one or more of the following:

-a[s][n][z]	Add files
-A[n][p][l][m]	Saves UNIX file names, permissions, date/time modified, etc.
-b[path]	Create temporary ZIP file in alternate directory
-c -C	Create/edit file comments
-d	Delete files
-e[x s a b][s][x]	Method of compression
-f	Freshen files
-G	Store full native filenames in ZIP file
-h	Call up help screen
-i	Add changed files
-j -J[h][r][s]	Mask/unmask file attributes
-k	Retain original date of ZIP file
-l	Display license information
-m[u f]	Move files
-N	Controls mapping of UNIX to ZIP file names
-o	Set time and date of ZIP file to oldest file
-p -P	Store recursed pathnames
-q	Enable ANSI comments
-r	Recurse subdirectories
-spassword	Scramble files with password protection
-T[d][f][s]	Saves specific UNIX file types, including directories, files, and segment directories (special for PI/Open software).
-u	Update files
-v[b t][c d e n o p s][r]	View files
-V[b][e][i][-]	Controls verbosity of output to screen
-w -W[h][s]	Include hidden/system files
-x	Exclude files from ZIP operation
-z	Create/edit ZIP comment

Before Starting

There are several size factors you must be aware of when using the ZIP command. The maximum size of a ZIP file is limited by the storage capacity of your disk (additionally there is a further limitation of 4,000 files for this implementation). You need unused storage space at least equal to the size of the existing ZIP file in order to complete any updating process.

For example, if the ZIP file to be updated is currently 250K, you need 250K for actual storage space and another 250K for the updating process. This is because ZIP does not delete the original ZIP file until the new updated ZIP file has been created. So at the instant the new updated ZIP file is completed the original ZIP file will still exist. Therefore, you would need disk storage space twice the actual size of the ZIP file.

If this poses a problem to you, it can be counteracted by using the **-b** option. This process is explained in greater detail later in this section.

The ZIP command creates compressed files which are compatible with PKWARE, Inc.'s PKZIP and PKUNZIP programs which run under MS-DOS. During the compression process, the file names are converted to conform to MS-DOS rules: The file names are truncated to 8 characters, extensions are truncated to 3 characters, and illegal characters such as the slash (/) are converted to dollar (\$). Additionally, if the **-p** or **-P** options are used, the directory names must be converted using the same rules. If you notice that the file names saved to the ZIP file are not the same as the original UNIX names, they have been *mapped* by this process.

The mapping algorithm may cause problems when UNIX file names *map* to the same ZIP file name. For example, the files FILENAMEA and FILENAMEB would both map to FILENAME when they are truncated to 8 characters. To avoid this, when ZIP detects a *collision* (in other words, when both files map to the same name), ZIP will replace the file's extension with a revision counter: FILENAMEA would first be saved as FILENAME, FILENAMEB would be saved next as FILENAME.\$00, FILENAMEC would be saved as FILENAME.\$01, etc.

However, during the mapping process, the reference to the original UNIX file name is normally lost. When the files are uncompressed, they will be restored using their new ZIP file names: FILENAME, FILENAME.\$00, FILENAME.\$01. This also affects the **-u** (Update) and **-f** (Freshen) options as they cannot match the original UNIX name with the file names saved in the ZIP file. If ZIP detects a file collision during the **-u** or **-f** options it will issue a warning and skip the file in question.

See the **-A** option below. It causes the full Unix filename to be stored in a hidden place within the zip file, and allows you to work around the limited DOS filename size. Also, see the **-G** option. It has a similar function.

Controlling the Default File Name Mapping

You may wish to alter the rules for file name mapping. If you are transporting the ZIP files to other Unix systems, or if you are using ZIP for archival purposes, then read the next sections, entitled *Retaining UNIX File Names and Attributes*. This will allow you to keep the exact 14 to 1024 character Unix file name. However, if you are moving the ZIP files to a non-Unix system but you wish to control the file name mapping, read this section.

ZIP supports explicit mapping of UNIX file names to and from ZIP DOS style names. Mapping rules may be specified on the ZIP or UNZIP command line, or they may be listed in a file. DOS-style wildcarding characters (* and ?) may be used to allow more flexibility when creating your mapping rules.

UNIX to ZIP file name mappings may be specified on the command line by using the `-N` option (`N` must be uppercase). For example:

```
zip z1 this_file -Nthis_file=this.fil
```

This example would add the UNIX file "`THIS_FILE`" to the ZIP file `z1` using the name "`THIS.FIL`". `-N` may be used multiple times to specify different mapping rules for other files. Alternatively, the rules may be listed in a file (one per line) and the file name may be designated by preceding it with the `@` character. For example,

```
zip z1 this_file -N@mapping_rules
```

UNZIP supports the same `-N` option. The order is identical to that of the ZIP command (UNIX file name followed by the ZIP file name) so that the same mapping rules file may be used by both commands.

Default mapping rules may be specified in the `zip.cfg` file using the `NAMES` directive:

```
NAMES=filename
```

where "`filename`" is the name of the mapping rules file.

Each mapping rule may be one of:

- (a) A file name only; e.g. `UNIX.KSH=UNIX.BAT`
- (b) A directory name only; e.g. `USR/=BIN/`
- (c) A full directory and file name; e.g. `USR/BIN/LD.RUN=BIN/LD.EXE`

When ZIP translates a full directory and file name (as in when the `-P` or `-p` options have been specified), the mapping table is first checked to see if it contains an entry for the full name. If no matching entry is found, the full name is broken into its directory and file name components and the table is searched again for corresponding mapping rule for both. This allows you to specify mappings for given directories, for given file names, or for specific absolute path names.

Mapping rules may include DOS-style wildcarding characters (*, ?):

***.KSH=*.BAT**

This example would rename all UNIX Korn shell script files to DOS BAT files. Wildcarding allows you to set up more general mapping rules. Wildcards may be escaped with the backslash (\) character.

Retaining UNIX File Names

If you are creating a ZIP file that will be restored under Computronics ZIP for Unix, the OS/2 version of ZIP, or selected other versions that support long filenames, it is possible to zip these files and preserve the original filename. To accomplish this, simply place the option "-G" on the command line. This will tell ZIP to store the native filename in the ZIP collection. No mapping or translation will occur. This option will work with Computronics ZIP only if the file is UNZIP'ed with the same -G option. It will also work with many other versions of PKZIP that support long filenames. Note that this option is not as complete as the -A option discussed below. The -A option also can retain file accessed dates, protections, and other attributes. The -G option only relates to the file name itself.

You can make the "-G" option the default, using the zip configuration file. Use the **CONVERTNAMES** directive.

CONVERTNAMES	= ON	The default, uses the DOS style 8.3 filenames.
CONVERTNAMES	= OFF	Do not convert names; leave them in their native format.

Retaining UNIX File Names and Attributes, Cross Platform Support

If you will be transferring the ZIP file to another Unix system, or if you wish to use ZIP to save disk space on your system (that is, you are using ZIP to save space for archival purposes), you will want to retain the UNIX file names and attributes. The ZIP and UNZIP commands have special options -A and -T for these purposes. These are discussed in detail in the next paragraphs. If you ZIP a file using -A, you need to use this option on the UNZIP command to get the file names/attributes restored properly.

A ZIP file created using these options is still readable on another platform, such as the IBM PC. The file is created using the standard DOS 8 character dot 3 character file names. However, a special file, called !ATTRIB, is also written to the ZIP file when these options are used. This file will be treated as just a binary file under DOS, but is handled specially by ZIP for UNIX. It contains sufficient information to restore the UNIX file names, file types, permissions, etc. It will not appear in any listing when using ZIP on the Unix system, and it cannot be saved or restored by ZIP for UNIX.

Note: Be careful not to modify the !ATTRIB file when using DOS-based ZIP compression utilities. Otherwise the file attributes will not be restored properly.

The **-A** and **-T** options cannot be combined with other options, and they must be in uppercase to avoid conflict with other options:

File attribute settings:

- Am** Saves/restores the date/time modified and date/time accessed.
- An** Saves/restores the UNIX file name.
- Ap** Saves/restores the UNIX permissions, including the user'd uid and group id (uid/gid). The actual permissions mask is also saved.
- A** Combines the above options, that is, **-Amnp**.

You can select a default for the **-A** option in the configuration file. Use the **XATTRIB** directive.

XATTRIB = NAME	(-An) Include native file name.
XATTRIB = PROTECT	(-Ap) Include permissions.
XATTRIB = DTM	(-Am) Include date/time stamp.
XATTRIB = ALL	(-Anpm) Include all attributes.
XATTRIB = NONE	Do not include attributes.

File type selections:

- Td** Save/restore directories (assumed if **-r** and **-P** options used on ZIP)
- Tf** Save/restore files (assumed normally)
- Ts** Save/restore segment directories, such as PI/Open™ files (this is a special option for users of this product only).
- T** Save/restore all files other than CAM files. Equivalent to **-Tdfs**.

You can select a default for the **-T** option in the configuration file. Use the **FILETYPE** directive.

FILETYPE = FILE	(-Tf) Include regular files.
FILETYPE = DIR	(-Td) Include directories.
FILETYPE = SEGDIR	(-Ts) Include segment directories (PI/Open files).
FILETYPE = ALL	(-Tdfs) Include all file types.

Command Options

This section will give a detailed explanation of the options used in the ZIP command. Recall that except where noted, several options can be combined on the command line, and the options can be placed in any order.

```
zip [options[ ...]] zipfile [@list] [filename[ ...]]
```

zipfile ZIP file name. Default extension is **.zip**.

filename Specifies a particular file or group of files that are to be processed. Wildcards *****, **?** are ok. Default is all files.

list An optional list file name. See Appendix A for information on how to use a list file.

-a[t|T|n|s|z] This option will add files to either a newly created or existing ZIP file. Same-name files will overwrite those already in the ZIP file regardless of which has the latest date. This is the default option if no other options are specified.

```
zip -a Text chap1.doc chap2.doc toc.doc
```

Once the program is finished creating a new ZIP file containing the three files listed above, the extension **.zip** will be added to Text making it **TEXT.zip**.

The suboptions control text translation and other features. Use "**-at**" to cause UNIX text files to be converted to DOS format, leaving binary files alone. (This overrides any setting in the **zip.cfg** file, discussed in Appendix B.) The option "**-aT**" forces text conversion on all files. For more details on these options, see the "**-T**" and "**-t**" options below.

The suboption "**-an**" is the opposite of the text conversion settings. It specifies that no text conversion should ever occur. Use this option to achieve the effect of using a **zip.cfg** setting of **CONVERTTEXT=NONE**.

If the option is "**-az**", no control-Z will be appended to the DOS file during conversion. This is not normally a problem. However, some versions of ZIP, such as a shareware version for the Apple Macintosh™, do not expect a control-Z at the end of a file. This option will force this character to not be added.

-A[m|n|p]

The "**-A**" option (uppercase "**A**") specifies that UNIX attributes should be saved as a part of the ZIP file.

The option "**-Am**" saves the date/time modified and date/time accessed.

Specify "**-An**" to save the full 14 to 1024 character UNIX file name.

Finally, specify "**-Ap**" to save the file's permissions, including the owner's uid and group id values.

These options can be combined. Specifying simply "**-A**" will select all of these options, that is, "**-Anp**".

-b[*path*]

This option is designed to be used when insufficient disk space is a problem. Every time a ZIP file is updated, ZIP creates a new pre-ZIP file. When the updating is completed, the original ZIP file is automatically deleted and the pre-ZIP file becomes the new ZIP file. The result of this operation is that you need disk storage space equal to at least twice the actual size of the ZIP file while updating the ZIP file.

The **-b** option will locate the pre-ZIP file on the path and location specified. This file is used only in the process of updating the original ZIP file. It will replace the original ZIP file and will NOT exist on the location specified in the **-b** option, when the update process is completed. The **-b** option can be used along with any of the update options.

```
zip zipdoc *.doc -b/tmp/workarea -u
```

The above example illustrates the use of the **-b** option. A ZIP file, **zipdoc**, located in the current directory, will be updated with all the **.doc** files.

A temporary ZIP file, used in the update process will be located in the **workarea** directory on the **tmp** file system. This file will not exist when the update process is completed.

By specifying the following directive in the configuration file, you can select a default working directive. This can be overridden by the **-b** command line option whenever needed.

```
ALTERNATIVE = pathname
```

where "*pathname*" is the directory of the working directory.

-c -C This option is case sensitive. It is used to create file comments for the individual files contained in the ZIP file. The maximum length for the comments is 60 characters. You will be prompted to enter the comments. If a comment already exists, you can edit it or press the **[Enter]** key to retain the same comment. To delete an existing comment, press **[Space]** followed by the **[Enter]** key.

The **-c** (lower case c) option is used to edit all comments for existing files or files that are added.

The **-C** (upper case C) option is used when you want to create comments for only the new files that were added to the ZIP file. It is used along with the add/update options. If **-C** is used without any other options, it has the same effect as **-c**.

```
zip textfile.zip -C -u
```

In the above example, any comments on existing files will be retained and you will be prompted to enter comments only for the newly added files.

-d This option is used to delete single or multiple files within a ZIP file. The name of the file will be displayed on the screen as it is being deleted.

```
zip -d OldFile.zip Garbage.Txt Useless.Dbf
```

In this example, **Garbage.Txt** and **Useless.Dbf** will be deleted from the files in **OldFile.zip**.

-e[x|s|a|b|s|x] This option is used to specify the compression method to be used in creating the ZIP file. PKWARE, Inc.'s PKZIP program version 1.1 supports two types of compression: "Shrinking" and "Imploding". Version 2.0 of PKZIP added two new "Deflate" modes. Earlier versions supported the Reducing algorithm. In addition the file may simply be "Stored" in its native format when the compression algorithms do not yield any space reductions.

ZIP for UNIX supports all of the file formats of PKZIP. The UNZIP program will read and uncompress all file formats. The ZIP program will pick an appropriate method for each file, but you can specify your own method using this **-e** option. The command line options are explained below. Note that the options for version 3.0 or later are different than in prior versions.

- s** This specifies the deflatS (fast) compression method is to be used.
- x** This specifies the deflatX (maximal) compression method is to be used. This is the default option and yields the best results at the expense of slightly slower execution speed than some of the other methods.
- x** This specifies the Imploding method is to be used on all files being compressed.
- s** This specifies the Shrinking method is to be used on all files being compressed.
- a** This specifies the Imploding method is to be used on all ASCII files and the Shrinking method is to be used on all binary files.
- b** This specifies the Imploding method is to be used on all binary files and the Shrinking method is to be used on all ASCII files.

You should be aware of the following when specifying a compression method.

Numbers after the parameters such as **-ex4** or **-ea3** are accepted for compatibility with PKWARE, Inc.'s PKZIP 0.92 but they have no effect on the data compression in ZIP for UNIX.

Some example commands are shown below.

```
zip zipdoc -es *.DOC
```

This command would use the "deflatS" method on all the files with a **.DOC** extension, placing them into the ZIP file.

```
zip -ea zipdoc
```

This command specifies that ASCII files are to be Imploded and that binary files are to be Shrunk.

The desired default compression method can be specified in the configuration file using the **COMPRESS** directive:

COMPRESS	=	SIZE	maximal deflate method
COMPRESS	=	SPEED	fast deflate method
COMPRESS	=	SHRINK	shrink method
COMPRESS	=	IMPLODE	implode method

- f** This option updates files that already exist in the ZIP file with same-named files with later times and dates.

The Freshen option is limited to working with files that *ALREADY* exist in the ZIP file. Existing zipped files will be overwritten *ONLY* if the files have been updated since being added to the ZIP file. It will *NOT* add new files.

```
zip -f STORE.zip *
```

In the above example, all the existing files in the ZIP file, **STORE**, will be updated if a newer version of the file exists in the current directory.

- G** This option will cause ZIP to store the native file names in the ZIP file. See the section *Retaining UNIX File Names*.

```
zip -G myfile.zip
```

- h** This command will display a help screen whenever typed on the command line.

```
zip -h
```

- i** This option will add only those files to the ZIP file that were *NOT* backed up the last time. It adds only those files that have been changed. If the program returns the notation "No files found", this means that everything had been previously backed up.

```
zip -i allfiles.zip
```

In this example, all the files in the ZIP file **allfiles** that had not been previously backed up, will be backed up.

- j[h][s][r] -J[h][s][r]**

This option is included for compatibility with PKWARE, Inc.'s PKZIP program. The PKZIP program, which runs under DOS, uses the **-j** option to mask the hidden, system, or readonly attributes of files so that they will be treated as normal writable files. To preserve these attributes, the **-J** option is used. To specify hidden files, use **h**; to specify system files, use **s**; and to specify readonly files, use **r**. An example is shown below.

```
PKZIP SAVE.ZIP -a -jhs -whs ibmbio.com ibmdos.com
```

In this example, the hidden/system/readonly files **IBMBIO.COM** and **IBMDOS.COM** will be included in **SAVE.ZIP**. However, their hidden and system attributes will be masked off in the **.ZIP** file, saving them with the readonly attribute only.

Note: There is no space between the J or j, and the h, s or r.

The UNIX operating system does not support the system and hidden attributes. The readonly attribute is simulated by setting the file permissions to readonly under UNIX. When the files in the example above are extracted using UNZIP (see section 4 for a description of the UNZIP command), the readonly attribute will be mapped to the proper permissions under UNIX.

When the hidden, system and readonly attribute of files are masked off, the `-v` View option will display the attribute as `--w` (for writable) in the attribute column.

By using the following commands in the configuration file, you can specify whether system, hidden or readonly files will be masked, and thus treated as normal writable files.

```
MASK = hidden
MASK = system
MASK = readonly
```

`-k -k-`

The `-k` option will keep the current date on the ZIP file. The `-k-` option will set the ZIP file date to the current date. This option can be used to override a command in the configuration file. By default the ZIP file being created will be given the current time and date.

Note: *Please see the `-o` option for related information.*

By using the following command in the configuration file, the current ZIP file date will be retained when the file is updated.

```
ZIPDATE = keep
```

To override the configuration string, simply include the `-k` or `-k-` option on the command line. An example is shown below.

```
zip DONE.zip -f -k- *.txt
```

In this example, all the files with a `.txt` extension will be freshened in the ZIP file. The file will then be given the current date. Assuming the ZIPDATE parameter had been set to "keep" in the configuration file, it would be overridden to "current" by the `-k-` option on the command line.

`-l`

This command will display the software license agreement.

```
zip -l
```

-m[u | f]

This option will add files to a new or existing ZIP file then automatically delete the original or source file. There are two additional options available with the Move option: Move Update and Move Freshen.

When used alone, the Move option is similar to the **-a** (Add) command option except that it automatically deletes the original files. This is a real time saver, for example, when compressing an entire directory. The **-m** option compresses all your files into a single ZIP file, then deletes the original files.

The following list describes each of the options you can use with the **-m** option. Do not separate the values with spaces.

- u** This specifies the Move option used along with the Update option. When these two options are used together, the existing ZIP file will be updated with files only if they are not currently in the ZIP file, or if they are dated later than those with the same name already in the ZIP file. After the ZIP file is updated, the original files will be deleted.
- f** This specifies the Move option used along with the Freshen option. When these two options are used together, files already existing in the ZIP file will be updated with same-named files with later times and dates. After the ZIP file is updated, the original files will be deleted.

The original files are deleted after the program verifies the ZIP file for accuracy. If an error does occur, such as DISK FULL, the original files will NOT be deleted.

```
zip -m tiny.zip DB/*.dbf
```

In this example, all the database files will be compressed into `tiny.zip` and then will be erased from the `DB` directory.

-N

This option controls the mapping of UNIX file names to DOS file names. This subject was discussed in an earlier part of this section of this manual. See the section *Controlling the Default File Name Mapping*.

-o -o-

This option can be used to override a command in the configuration file. The **-o** option will set the ZIP file date to the latest file date included. The **-o-** options will set the ZIP file date to the current date. By default the ZIP file being created will be given the current time and date.

Note: *Please see the -k option for related information.*

By using the following command in the configuration file, the ZIP file date will be determined by the date of the latest file included in the ZIP file.

```
ZIPDATE = latest
```

To override the configuration string, simply include the **-o** or **-o-** option on the command line.

Note: *This option is denoted by the letter o, not a zero. An example is shown below.*

```
zip DONE.zip -a -o- *.doc
```

In this example, all the files with a .doc extension will be added to the ZIP file. The file will then be given the current date. Assuming the ZIPDATE parameter had been set to "latest" in the configuration file, it would be overridden to "current" by the **-o-** option on the command line.

-p -P

This option is case sensitive. It allows you to store the directory and path information pertaining to each file within the ZIP file.

If **-p** (lower case p) is used, all the pathnames that are recursed into will be stored in the ZIP file. The **-p** option is meaningful only when used with the **-r** option explained below. If **-P** (upper case P) is used, all the pathnames specified on the command line and those recursed into will be stored in the ZIP file.

-q -q-

This option can be used to override a command in the configuration file. By default, ANSI sequences are disabled.

ANSI sequences are enabled (**-q**) or disabled (**-q-**) in the configuration file by the following commands.

```
ANSI = Enabled
ANSI = Disabled
```

To override the configuration string, simply include the **-q** or **-q-** option on the command line. An example is shown below.

```
zip allfile.zip -u -q-
```

In this example, the ZIP file, **allfile.zip** will be updated. If the ANSI comments had been enabled in the configuration file, they would be disabled by the **-q-** option on the command line.

-r

This command allows you more flexibility when creating or updating ZIP files. The program will recurse through the specified directories when obtaining files to compress.

```
zip alldoc -r *.doc word/pk/*.txt
```

This command will search all **.DOC** files in the current directory and all **.TXT** files on the **word/pk** directory and all directories below.

```
zip -rp stuff
```

This example will compress all the files in the current directory, and all directories below the current subdirectory. UNZIP can then restore this directory tree either in the same directory, or at any place in the directory tree. An entire directory tree can be compressed into a ZIP file and then restored by using the **-d** option in UNZIP. (See the **-d** option for UNZIP, Section 4.)

By default, the filename only will be stored, unless the **-p** or **-P** option is specified on the command line.

-spassword

This option is used to scramble, or encrypt the files in the ZIP file. It includes password protection. When extracting the ZIP file, the files will not be extracted unless the correct password is included on the command line.

When specifying the password, there is no space between the **s** and the password. Also the password is case sensitive. It must be entered exactly the same when you Unzip the file. It is important that you remember or somehow record the passwords that you use, because ZIP does not retain a record of these passwords anywhere.

```
zip PAYROLL -m -sSecret *.pyr
```

In the above example, the ZIP file `PAYROLL` will be created with all the files having a `.pyr` extension. In order to Unzip this file, the password `Secret` will have to be specified exactly as shown above.

-t -T

This option is case sensitive. It allows you to convert your UNIX text files to MS-DOS text format.

If **-t** (lower case t) is used, UNIX text files will be converted to MS-DOS text format; binary files will remain unchanged. If **-T** (upper case T) is used, all files added to the ZIP file will be converted to MS-DOS text format.

If the **-t** option is used, ZIP first scans a file to see if it contains any binary characters. If not, the file is assumed to be a text file and the file is converted to MS-DOS format. Certain non-printable characters may cause this test to fail, and you may specify that ALL files be converted by using the **-T** option. You can also speed up the operation of ZIP by controlling text conversion via the **T** or **n** options, or via the appropriate directives on **CONVERTTEXT** in the `zip.cfg` file.

This option is an extension to PKWARE, Inc.'s PKZIP program and its syntax follows nonstandard rules. The **t** or **T** must directly follow one of the following options: **a**, **f**, **i**, **u**. There must be no spaces between the option letter and the **t** or **T** (e.g. **-at**, **-iT**).

You can force text conversion to be turned off via a configuration directive, or via a special command option. If you use **-an**, no text conversion will be performed.

By default, only text files are converted. This default may be overridden by using the **CONVERTTEXT** command in the configuration file.

```
CONVERTTEXT = None
CONVERTTEXT = Text
CONVERTTEXT = All
```

-T[a|d|f|s] When a file is to be moved to another Unix system, or if the ZIP file is being created to save disk space, you may wish to control what types of files are saved. Use the "**-T**" option (upper case "**T**") to select the file types to be saved.

Specify "**-Td**" to save directories. This option is assumed if the **-r** and **-P** (or **-p**) options are used on ZIP.

The default option is "**-Tf**". This saves standard files.

The option "**-Ts**" will save segment directories, such as PI/Open™ files (this is a special option to ZIP).

The default option, if none is specified, is "**-Tf**". If the ZIP command also includes the save directories options, "**-r**" and "**-P**", then the default options are really "**-Tfd**". If you simply specify "**-T**", then files and directories will be saved. This is the same as "**-Tdfs**".

-u The existing ZIP file will be updated with files only if they are not currently in the ZIP file, or if they are dated later than those with the same name already in the ZIP file.

Although this command is similar to the **-a** (Add) option, it has an update safeguard. It will overwrite existing compressed files only if the selected files have a more recent time and date.

```
zip -u OldFile.zip *.Txt Money.Wks Funds.Wks
```

This example will update the ZIP file named **OldFile.zip**. The files listed above will be updated within the ZIP file. That is, they will be added if they do not already exist, or overwritten if they already exist in the ZIP file with an earlier date.

-v[b|t][c|d|e|n|o|p|s][r]

This option will display technical information concerning the files contained within a ZIP file. There are several options available with the View option. All are explained below. The options determine how the information will be displayed and sorted.

The following information about each file contained in the ZIP file will be displayed on the screen.

<u>Field</u>	<u>Description</u>
Length	Original length of the file
Method	Type of file compression used
Size	Size of the compressed file
Ratio	Percent reduction in file size
Date	Actual date of the file
Time	Actual time of the file
CRC-32	The CRC-32 value of the file
Attribute	The attributes of the file (s=System, h=Hidden, w=Writable, r=readonly file, and *=encrypted)
Name	Name of the file

The View options for ZIP are explained below. The options determine how the information will be displayed and sorted on the screen. The following display options are available:

- b** The **-vb** Brief display option will display all the information shown above except the CRC-32 value and the file attribute.
- t** The **-vt** additional Technical display option is used to display extra technical information. When this option is active, the following information (shown on the following pages) will be displayed on the screen.

The following information will be displayed for each file in the ZIP file when the **-vt** option is used.

<u>Field</u>	<u>Description</u>
Filename	Name of the file
File type	Type of file
Encrypted	It will be noted here if the file is encrypted
Attribute	The attributes of the file (s=System, h=Hidden, w=Writable, and r=readonly file)
Comments	File comment, if present, will be listed here
Date and Time	Date and Time of the ZIP file
Compression Method	Either Imploding or Shrinking (unless an earlier version of ZIP was used)
Compressed Size	Reduced or compressed size of the file
Uncompressed Size	Original length of the file
CRC-32	The numeric CRC-32 value
Created by	The version of ZIP and the operating system used to create the ZIP file.
Needed to extract	The version of ZIP required to extract the files

- c** This option is used to display any existing file comments.
- r** Display information in reverse order from the default order. This option can be used with any of the other View options to reverse the order of sorting.

Any of the display options explained above can be used with the sorting options listed below.

The following sorting options are available with the View option. The default order of sorting is also specified. (This order can be reversed by using the particular sorting option along with the `-vr` View Reverse option.)

- `-vd` Sort by date of files, oldest to most recent.
- `-ve` Sort by file extension, alphabetically.
- `-vn` Sort by name of files, alphabetically.
- `-vo` Display in original order, i.e. the order in which the files were zipped. (This option can be used to override any configuration parameter you may have set.)
- `-vp` Sort by percentage ratio of compression, smallest to largest.
- `-vs` Sort by size of files, smallest to largest.

If no sorting option is specified, the files will be displayed in the order they were compressed.

The following example illustrates the command used to display the technical information about a ZIP file sorted by date. The output is shown below.

```
zip allfiles.zip -vd
```

Length	Method	Size	Ratio	Date	Time	CRC-32	Attr	Name
6144	Implode	2228	64%	01-04-80	13:52	a1f719af	--w	SAVE.DOC
7168	Implode	2305	68%	01-04-80	14:58	fc970ad9	--w	EWEXE.DOC
2560	Implode	1312	49%	09-03-94	11:12	b89abd8c	--w	TYPOS.DOC
423	Implode	337	21%	09-03-94	11:14	131dd142	--w	PINTS.DOC
2793	Implode	1264	55%	09-03-94	19:16	dab0a3b5	--w	DCA.DOC
6638	Implode	3082	54%	01-25-89	21:52	80046b74	--w	ADME.DOC
14848	Implode	5514	63%	03-02-89	09:03	7f2d751b	--w	IIM1.DOC
10240	Implode	4312	58%	03-02-89	09:04	70324a2f	--w	IIM2.DOC
75264	Implode	23261	70%	03-02-89	09:16	8f5baa0f	--w	IIM3.DOC
-----		-----	---	-----				
126078		43615	57%	12				

In the example shown below, the additional technical information will be displayed. The files are sorted alphabetically by name. The output displayed below would be repeated for each file within the ZIP file.

```
zip -vtn docs.zip
```

```
      Filename:  M3.DOC
      File type:  text
      Attributes:  --w
      Date and Time:  Jul 17,1994  09:15:00
      Compression Method:  Implore
      Compressed Size:  8258
      Uncompressed Size:  25600
      32 bit CRC value:  27418eb3
      Created by:  PKZIP: 1.0 under MS-DOS
      Needed to extract:  PKUNZIP: 1.0
```

Note that the example above states that the ZIP file was created using PKZIP under MS-DOS, and that PKUNZIP is needed to extract the files. ZIP for UNIX sets these values so that the ZIP files created will be compatible with PKWARE, Inc.'s PKZIP and PKUNZIP programs. ZIP for UNIX will create and read ZIP files using PKWARE, Inc's standard format.

Any of the View options listed above can be set in the configuration file. For example, if you wanted to always display the files contained in a ZIP file sorted by extension, this can be set in the configuration file. If no View options are specified, the default View setting is the order in which the files were compressed in the ZIP file. Any view options specified on the command line will however, override the configuration parameters.

The following View options can be set in the configuration file. The corresponding command line option is also listed for reference.

```
VIEW = date           -vd
VIEW = extension     -ve
VIEW = name          -vn
VIEW = ratio         -vp
VIEW = size          -vs
VIEW = natural       -vo
VIEW = reverse       -vr
VIEW = brief         -vb
VIEW = long          -vt
VIEW = comments      -vc
```

As in the view options specified on the command line, the last four options, **reverse**, **brief**, **long** and **comments**, can be combined with any of the sorting options listed above. For more detailed information on the configuration file, see Appendix B.

-w[h][s] -W[h][s]

This option is included for compatibility with PKWARE, Inc's PKZIP program. PKZIP uses this option to specify whether or not the hidden or system files will be included in the ZIP file. The **-w** option is used to include hidden or system files. The **-W** option excludes hidden or system files. It can be used to override a command in the configuration file. By default, hidden or system files will not be included.

To specify hidden files, use **h**; and to specify system files, use **s**. An example is shown below.

```
PKZIP ALL.ZIP -a -Whs
```

In this example, all of the files in the current directory will be compressed into `ALL.ZIP` except the hidden and system files.

Note: *There is no space between the **w** or **W**, and the **h** or **s**.*

UNIX does not support the hidden or system file attributes; the **-w** option can be specified but it will have no effect. UNIX treats files whose names begin with a period as hidden. However, ZIP does not change the name of the file to make the file "hidden". (This would cause too many other problems).

By using one of the following commands in the configuration file, you can specify whether system or hidden files will be included by default.

```
INCLUDE = hidden
INCLUDE = system
```

For more detailed information on the configuration file, see Appendix B.

-x

This option is used to exclude files from the current operation. It can be used along with most of the ZIP command options such as add, update, view, delete, etc.

```
zip stuff * -x*.OLD
```

In the above example, all of the files in the current directory will be compressed into the file `stuff.zip` except the files with a `.OLD` extension.

-z This option allows you to create a descriptive label for your ZIP files. The ZIP comment will then automatically be displayed by ZIP or UNZIP whenever the specific ZIP file is processed.

When you type in the command, as shown below, you will be prompted to enter the ZIP comment. The **-z** option can also be used to edit an existing ZIP comment.

The **-z** option can be used alone on an existing ZIP file or in combination with any of the add/update commands. Examples are shown below.

```
zip zipdoc -z
```

After entering the above command, the program will prompt you to type in a ZIP comment for the ZIP file `zipdoc.zip`. After entering the ZIP comment, press the [Enter] key to save it.

```
zip zipdoc -z -a *.doc
```

In this example both files and ZIP comments will be added. To add files and add a ZIP comment, both the **-z** and one of the add/update commands must be entered.

-v[b][e][i]- This option allows you to control the verbosity of the output to your screen (the information that is output). You may want to suppress certain items if you are using a script, for example, to cut down on the amount of information sent to the user.

The option **-vb** shows the banner when the program is started.
-ve causes error messages to be displayed.
-vi causes informational level messages to be displayed.

Using a hyphen (-) disables an option.

Thus **-vb-** disables the banner.
-ve- disables error messages. The program sets error return codes to an appropriate value to indicate an error. (See Appendix D).
-vi- disables informational level message.
-v- disables banner, error and informational level messages.

The suboptions may be combined to provide the desired level of reporting; e.g. **-vb-i-** disables the banner and informational level messages. The default, if no options are specified, is **-vbei**.

Tutorial

This section will describe some simple examples, illustrating the command format for creating a ZIP file.

Use ZIP to reduce or compress the size of your files. All compressed files are stored in a ZIP file, while in their compressed state.

The benefits of compressing are:

- Better use of disk storage area
- Sharing files between the Unix and the PC
- Collecting numerous files under a single file name
- Faster telecommunication file transfer via a modem

ZIP uses four areas typed at the prompt:

```
zip [options[ ...]] zipfile [filename[ ...]]
```

- | | | |
|----|---|--|
| 1. | The actual program name | zip |
| 2. | The command <i>options</i> | letter commands
example: -a = add files |
| 3. | The name you give your ZIP file and its location. If no extension specified, default is .zip . | any valid UNIX file name |
| 4. | The files that get compressed into the ZIP file. | a list of UNIX file names, including the wildcard characters * and ? |

Step One / Add All Files

In this exercise you will create a file named **NEWFILE.zip** which will contain all of the files in the current directory in compressed format.

Example: `zip -a NEWFILE *`

The wildcard ***** is a UNIX wildcard, which means add all files in the current directory. The **.zip** extension will be added automatically when the ZIP file has been completed.

Step Two / Add Two Files

You may enter one or more file names on the command line for the ZIP program.

Example: `zip -a MYDIR/BUDGET LOTUS/CHECKS.WKS LOTUS/MONEY.WKS`

This will create a ZIP file named **BUDGET.zip** in the directory **MYDIR**. It will contain two files, both currently located in the **LOTUS** directory.

Step Three / Freshen With Files Matching a Wildcard

Once created, additional files can be added to a ZIP file. You may selectively update compressed files within a ZIP file based on criteria such as the file name or date and time modified.

Example: `zip -f FILES *.TXT`

This will update an existing ZIP file named **FILES.zip**. The files with a **.TXT** extension that already exist in the ZIP file, and that are also dated later than those already within **FILES.zip**, will be updated.

Section Four / UNZIP Uncompression Program

This section describes how to use the UNZIP program. The UNZIP program may be used to extract files from compressed ZIP data files.

Quick Reference

The following table will provide a quick reference guide to all the options of UNZIP. The command syntax is also shown below.

Syntax

```
unzip [options[ ...]] zipfile [path/] [filename[ ...]]
```

Options

<i>zipfile</i>	ZIP file name; default extension is .zip
<i>path</i>	Directory to which files are to be copied; must include forward slash (/) as the last character
<i>filename</i>	Names of files to uncompress; wildcards *, ? are ok; default is all files
<i>Options</i>	An option list, containing one or more of the following:
-A[n][p][m]	Restores UNIX file names, permissions, date/time modified, etc.
-c[m]	Extract files to the screen [with more]
-d	Create directories on extraction from ZIP file
-G	Restore using native filenames in archive
-h	Call up help screen
-j -J[h][r][s]	Mask/Unmask file attributes
-l	Display license information
-n	Extract only newer files
-N	Controls mapping of UNIX to ZIP file names
-o	Overwrite existing files
-q	Enable ANSI comments
-spassword	Unscrambles files with password protection
-t	Test ZIP file integrity
-ucharacter	Cause specified special character in a file name not to translate to a ! character
-v[b t][c d e n o p s][r]	View file information
-x	Extract files (the default option)
-V[b][e][i][-]	Controls verbosity of output to screen

Before Starting

The command options used in extracting a ZIP file will be explained in detail in the next section. Listed below is a summary of important facts to keep in mind.

- ✓ The command options can generally be placed anywhere on the command line.
- ✓ The first filename listed on the command line will be interpreted as the ZIP file to be extracted. Subsequent filenames will be interpreted as specific files to be extracted.
- ✓ The destination directory name, if present, is indicated by a forward slash (/) as the last character of the directory name. The directory name must conform to the general rules for UNIX file names.
- ✓ All command options must be preceded by a hyphen (-) (e.g. `-v`).
- ✓ Unlike PKWARE, Inc.'s program PKUNZIP, unrelated options of UNZIP *can* be combined.

The default values for the command line are as follows:

- ☛ If no options are entered, the default is `-x`, the extract files option.
- ☛ If no extension is specified for the ZIP file the `.zip` extension is assumed.
- ☛ If no destination is specified for the extracted files, they will be located in the directory that the program is being run from.
- ☛ If no file specs (filenames, with or without wildcards) are listed, the default is `*`.

Controlling the Default File Name Mapping

You may wish to alter the rules for file name mapping. If you are transporting the ZIP files from other Unix systems, or if you are using UNZIP for archival purposes, then read the next section, entitled *Retaining UNIX File Names and Attributes*. This will allow you to extract the exact 14 to 1024 character file name, if the file was ZIP'ed appropriately. However, if you are moving the ZIP files from a non-Unix system but you wish to control the file name mapping, read this section.

UNZIP supports explicit mapping of UNIX file names to and from ZIP DOS style names. Mapping rules may be specified on the ZIP or UNZIP command line, or they may be listed in a file. DOS-style wildcarding characters (* and ?) may be used to allow more flexibility when creating your mapping rules.

The mappings of ZIP file names to UNIX names may be specified on the command line by using the `-N` option (`N` must be uppercase). For example:

```
unzip z1 this_file -Nthis_file=this.fil
```

This example would extract the DOS file "this.fil" from the ZIP file `z1` using the UNIX name "THIS_FILE". `-N` may be used multiple times to specify different mapping rules for other files. Alternatively, the rules may be listed in a file (one per line) and the file name may be designated by preceding it with the `@` character. For example,

```
zip z1 this_file -N@mapping_rules
```

Note that ZIP supports the same `-N` option. The order is identical to that of the UNZIP command (UNIX file name followed by the ZIP file name) so that the same mapping rules file may be used by both commands.

Default mapping rules may be specified in the `zip.cfg` file using the `NAMES` directive:

```
NAMES=filename
```

where "`filename`" is the name of the mapping rules file.

Each mapping rule may be one of:

- (a) A file name only; e.g. `UNIX.KSH=UNIX.BAT`
- (b) A directory name only; e.g. `usr/=BIN/`
- (c) A full directory and file name; e.g. `/usr/bin/LD.RUN=BIN/LD.EXE`

When UNZIP translates a full directory and file name (as in when the `-d` option has been specified on UNZIP), the mapping table is first checked to see if it contains an entry for the full name. If no matching entry is found, the full name is broken into its directory and file name components and the table is searched again for corresponding mapping rule for both. This allows you to specify mappings for given directories, for given file names, or for specific absolute path names.

Mapping rules may include DOS-style wildcarding characters (*, ?):

***.KSH=*.BAT**

This example would rename all DOS BAT files to UNIX Korn Shell script files. Wildcarding allows you to set up more general mapping rules. Wildcards may be escaped with the backslash (\) character.

Certain special characters (#, &, *, ? and \$) are normally translated to a ! when unzipping a file. Use the option *-ucharacter*, such as *-u#*, to turn off this behaviour for a specific character, and to cause it to remain in the unzipped file name. Multiple *-ucharacter* options can be specified on the same command line. Some characters have special meaning to the shell, and will need to be prefaced by a backslash. This is true for *, for example. Use the option *-u**.

Retaining UNIX File Names

If you are restoring a ZIP file that was created under Computronics ZIP for Unix, the OS/2 version of ZIP, or selected other versions that support long filenames, it is possible to unzip these files and preserve the original filename. To accomplish this, simply place the option **"-G"** on the command line. This will tell UNZIP to use the native filename, as stored in the ZIP collection. No mapping or translation will occur. This option will work with Computronics ZIP only if the file was originally ZIP'ed with the same **-G** option. It will also work with many other versions of PKZIP that support long filenames. Note that this option is not as complete as the **-A** option discussed below. The **-A** option also can retain file accessed dates, protections, and other attributes. The **-G** option only relates to the file name itself.

You can make the **"-G"** option the default, using the zip configuration file. Use the **CONVERTNAMES** directive.

CONVERTNAMES	= ON	The default, uses the DOS style 8.3 filenames.
CONVERTNAMES	= OFF	Do not convert names; leave them in their native format.

Retaining UNIX File Names and Attributes, Cross Platform Support

If you are transferring the ZIP file from another Unix system, or if you are using UNZIP to save disk space on your system (that is, you are using ZIP/UNZIP to save space for archival purposes), you will want to retain the UNIX file names and attributes. The ZIP and UNZIP commands have a special option **-A** for this purpose. This is discussed in detail in the next paragraphs. If you ZIP'ed a file using **-A**, you need to use this option on the UNZIP command to get the file names/attributes restored properly.

A ZIP file created using these options is still readable on another platform, such as the IBM PC. The file is created using the standard DOS 8 character dot 3 character file names. However, a special file, called **!ATTRIB**, is also written to the ZIP file when these options are used. This file will be treated as just a binary file under DOS, but is handled specially by ZIP for UNIX. It contains sufficient information to restore the UNIX file names, permissions, etc. It will not appear in any listing when using ZIP on the Unix system, and it cannot be saved or restored by ZIP for UNIX.

Note: Be careful not to modify the **!ATTRIB** file when using DOS-based ZIP compression utilities. Otherwise the file attributes will not be restored properly.

The **-A** option cannot be combined with other options, and it must be in uppercase to avoid conflict with other options:

File attribute settings:

- Am** Saves/restores the date/time modified and date/time accessed.
- An** Saves/restores the UNIX file name.
- Ap** Saves/restores the UNIX file permissions, as well as the user's uid and group id (gid).
- A** Combines the above options, that is, **-Amnp**.

You can select a default for the **-A** option in the configuration file. Use the **XATTRIB** directive.

XATTRIB = NAME	(-An) Include native file name.
XATTRIB = PROTECT	(-Ap) Include permissions.
XATTRIB = DTM	(-Am) Include date/time stamp.
XATTRIB = ALL	(-Anpm) Include all attributes.
XATTRIB = NONE	Do not include attributes.

Command Options

This section will give a detailed explanation of the command options used in the UNZIP command.

```
unzip [options[ ...]] zipfile [path/] [filename[ ...]]
```

zipfile ZIP file name. Default extension is **.zip**.

path An optional directory name. If present, all files extracted will be copied to the specified directory. *path* must conform to the rules for a valid UNIX file name (see the *Unix User's Guide* for details). The last character in the directory name must be a forward slash (/) to indicate that it is a UNIX directory name rather than a ZIP file name.

filename Specifies a particular file or group of files that are to be processed. Wildcards * and ? are ok. Default is all files.

-A[m|n|p] The "**-A**" option (uppercase "**A**") specifies that UNIX attributes should be restored from the ZIP file. You can only use this option if the file was created by ZIP for UNIX using the "**-A**" option of ZIP.

The option "**-Am**" restores the date/time modified and date/time accessed.

Specify "**-An**" to restore the full 14 to 1024 character UNIX file name.

Finally, specify "**-Ap**" to restore the file's permissions, and the file's user id (uid) and group id (gid).

These options can be combined. Specifying simply "**-A**" will select all of these options, that is, "**-Amnp**".

-c[m] This command will extract files from a ZIP file and display them on the terminal. The extracted files will scroll continuously on the screen. The software can be told to pause after each screen of output by using the **-cm** (more) option.

```
unzip STUFF -c dates.doc
```

In this example, the file `dates.doc` will be extracted from the noted ZIP file, and displayed on the screen.

-d

This command option uses the pathnames that have been stored in the ZIP file. The directory pathnames are stored in the ZIP file by the use of **-p**, **-P** and **-r** options in the list of ZIP commands explained in Section 3. If the stored output paths do not exist at the extraction destination, this command will create them as they are stored in the ZIP file.

For example: If the following directory tree exists on file system **tmppar**:

```
tmppar-----+--abc---+--dir1
              |          |
              |          +--dir2--+--dir3
              |          |          |
              +--xyz--    +--dir4
              |
              +--pdq
```

If you then executed:

```
zip -r -p STUFF /tmppar/abc/
```

And then you executed:

```
unzip -d STUFF /tmppar/pdq/newabc/
```

After the UNZIP command is processed, the **tmppar** file system would then look like this:

```
tmppar-----+--abc---+--dir1
              |          |
              |          +--dir2--+--dir3
              |          |          |
              +--xyz--    +--dir4
              |
              +--pdq---+--newabc--+--dir1
                              |
                              +--dir2--+--dir3
                                      |
                                      +--dir4
```

All of the files originally in **/tmppar/abc** and its subdirectories (**dir1**, **dir2**, **dir3**, and **dir4**) have been restored to the new directory **/tmppar/pdq/newabc**, with the directory tree being recreated.

If this **-d** option is not specified and pathnames have been stored in the ZIP file, only the filenames stored in the ZIP file will be used; any pathnames will be ignored.

- G** This option will cause UNZIP to use the native filename, as stored in the ZIP file, with no translation. See the section *Retaining UNIX File Names*.

```
zip -G myfile.ZIP
```

- h** This command will display a help screen whenever typed on the command line. The help screen will also occur when a command has been entered incorrectly.

```
zip -h
```

- j[h][s][r] -J[h][s][r]**

This option is included for compatibility with PKWARE, Inc.'s PKUNZIP program. The PKUNZIP program, which runs under DOS, uses the **-j** option to mask the hidden, system, or readonly attributes of files so that they will be treated as normal writable files. By default the hidden, system, and readonly attributes are masked off upon extraction. To preserve these attributes, the **-J** option is used.

The UNIX operating system does not support the system and hidden attributes. The readonly attribute is simulated by setting appropriate permissions under UNIX.

```
unzip ALL.zip -Jr
```

In this example, the readonly file attribute of the files being extracted will be preserved.

Note: *There is no space between the J or j, and the h, s or r.*

- l** This command will display the software license agreement.

- n** This command will extract files from the ZIP file only if they are newer than the same-named ones already on the disk, or if they do not already exist on the disk. This option serves as an overwriting safeguard. It will not allow an older version of a file in a ZIP file to overwrite a newer version on disk. *Newer* is defined as the most recent time and date modified.

```
unzip -n DB/lists *.dbf
```

In this example, all of the files with a **.dbf** extension, that are dated more recently than any same-named files on disk, or that do not already exist on disk, will be extracted from the ZIP file **lists**.

- N** This option controls the mapping of DOS file names to UNIX file names. This subject was discussed in an earlier part of this section of this manual. See the section *Controlling the Default File Name Mapping*.

- o** This command will extract files from a ZIP file and automatically overwrite any same-named files found in the destination where the extracted files are being located. The program will *not* prompt you for confirmation before overwriting these files.

Either the entire ZIP file can be extracted or certain files can be specified to be extracted. The extracted files will be located in the current directory unless you specify a destination.

```
unzip zfile *.doc -o
```

In the above example, all of the files with a .doc extension will be extracted from the ZIP file called **zfile** (the .zip extension is assumed). As the files are being extracted any files with the same name in the current directory will be automatically overwritten.

- q** By default, ANSI sequences are filtered from the comments. This option allows any ANSI sequences to be displayed unfiltered.

```
unzip ACCT.zip -o -q
```

In this example, the ZIP file will be uncompressed (overwriting files existing on disk) and the ANSI comments of the file being extracted will be enabled.

- spassword** This option is used to unscramble, or decrypt the files in the ZIP file. It is used in combination with the **-s** Scramble option of the ZIP command. It includes password protection.

When unzipping a ZIP file that has been encrypted, the files will not be extracted unless the correct password is included on the command line. UNZIP skips encrypted files if the password is not stated on the command line.

When specifying the password, there is no space between the **s** and the password. Also the password is case sensitive. It must be entered exactly the same as when you zipped the file. It is important that you remember or somehow record the passwords that you use, because ZIP does not retain a record of these passwords anywhere.

```
unzip PAYROLL -o -sSecret *.pyr
```

In the above example, all of the files with a .pyr extension will be extracted from the ZIP file **PAYROLL**. In this case the password specified when the file was zipped was **Secret**.

- t** This option is used to test the files to make sure they are valid and have not been corrupted. The files will be listed as they are being tested. 'OK' will listed after each file if it is not corrupted. This option does not extract files from the ZIP file; it only tests them.

```
unzip zipdoc *.doc -t
```

In the above example all the .DOC files will be tested from the ZIP file zipdoc.

-v[b|t][c|d|e|n|o|p|s][r]

This option will display technical information concerning the files contained within a ZIP file. There are several options available with the View option. All are explained below. The options determine how the information will be displayed and sorted.

The following information about each file contained in the ZIP file will be displayed on the screen.

<u>Field</u>	<u>Description</u>
Length	Original length of the file
Method	Type of file compression used
Size	Size of the compressed file
Ratio	Percent reduction in file size
Date	Actual date of the file
Time	Actual time of the file
CRC-32	The CRC-32 value of the file
Attribute	The attributes of the file (s=System, h=Hidden, w=Writable, r=readonly file, and *=encrypted)
Name	Name of the file

The View options for UNZIP are explained below. The options determine how the information will be displayed and sorted on the screen. The following display options are available:

- b** The **-vb** Brief display option will display all the information shown above except the CRC-32 value and the file attribute.
- t** The **-vt** additional Technical display option is used to display extra technical information. When this option is active, the following information (shown on the following pages) will be displayed on the screen.

The following information will be displayed for each file in the ZIP file when the `-vt` option is used.

<u>Field</u>	<u>Description</u>
Filename	Name of the file
File type	Type of file
Encrypted	It will be noted here if the file is encrypted
Attribute	The attributes of the file (s=System, h=Hidden, w=Writable, and r=read only file)
Comments	File comment, if present, will be listed here
Date and Time	Date and Time of the ZIP file
Compression Method	Either Imploding or Shrinking (unless an earlier version of ZIP was used)
Compressed Size	Reduced or compressed size of the file
Uncompressed Size	Original length of the file
CRC-32	The numeric CRC-32 value
Created by	The version of ZIP and the operating system used to create the ZIP file.
Needed to extract	The version of ZIP required to extract the files

- c** This option is used to display any existing file comments.
- r** Display information in reverse order from the default order. This option can be used with any of the other View options to reverse the order of sorting.

Any of the display options explained above can be used with the sorting options listed below.

The following sorting options are available with the View option. The default order of sorting is also specified. (This order can be reversed by using the particular sorting option along with the `-vr` View Reverse option.)

- vd** Sort by date of files, oldest to most recent.
- ve** Sort by file extension, alphabetically.
- vn** Sort by name of files, alphabetically.
- vo** Display in original order, i.e. the order in which the files were zipped. (This option can be used to override any configuration parameter you may have set.)
- vp** Sort by percentage ratio of compression, smallest to largest.
- vs** Sort by size of files, smallest to largest.

If no sorting option is specified, the files will be displayed in the order they were compressed.

The following example illustrates the command used to display the technical information about a ZIP file sorted by date. The output is shown below.

```
unzip allfiles.zip -vd
```

Length	Method	Size	Ratio	Date	Time	CRC-32	Attr	Name
6144	Implode	2228	64%	01-04-80	13:52	a1f719af	--w	SAVE.DOC
7168	Implode	2305	68%	01-04-80	14:58	fc970ad9	--w	EWEXE.DOC
2560	Implode	1312	49%	09-03-94	11:12	b89abd8c	--w	TYPOS.DOC
423	Implode	337	21%	09-03-94	11:14	131dd142	--w	PINTS.DOC
2793	Implode	1264	55%	09-03-94	19:16	dab0a3b5	--w	DCA.DOC
6638	Implode	3082	54%	01-25-89	21:52	80046b74	--w	ADME.DOC
14848	Implode	5514	63%	03-02-89	09:03	7f2d751b	--w	IIM1.DOC
10240	Implode	4312	58%	03-02-89	09:04	70324a2f	--w	IIM2.DOC
75264	Implode	23261	70%	03-02-89	09:16	8f5baa0f	--w	IIM3.DOC
-----		-----	---	-----				
126078		43615	57%	12				

In the example shown below, the additional technical information will be displayed. The files are sorted alphabetically by name. The output displayed below would be repeated for each file within the ZIP file.

```
unzip -vtn docs.zip
```

```

      Filename:  M3.DOC
      File type:  text
      Attributes:  --w
      Date and Time:  Jul 17,1994  09:15:00
      Compression Method:  Implode
      Compressed Size:  8258
      Uncompressed Size:  25600
      32 bit CRC value:  27418eb3
      Created by:  PKZIP: 1.0 under MS-DOS
      Needed to extract:  PKUNZIP: 1.0

```

Note that the example above states that the ZIP file was created using PKZIP under MS-DOS, and that PKUNZIP is needed to extract the files. ZIP for UNIX sets these values so that the ZIP files created will be compatible with PKWARE, Inc.'s PKZIP and PKUNZIP programs. ZIP for UNIX will create and read ZIP files using PKWARE, Inc's standard format.

Any of the View options listed above can be set in the configuration file. For example, if you wanted to always display the files contained in a ZIP file sorted by extension, this can be set in the configuration file. If no View options are specified, the default View setting is the order in which the files were compressed in the ZIP file. Any view options specified on the command line will however, override the configuration parameters.

The following View options can be set in the configuration file. The corresponding command line option is also listed for reference.

VIEW	= date	-vd
VIEW	= extension	-ve
VIEW	= name	-vn
VIEW	= ratio	-vp
VIEW	= size	-vs
VIEW	= natural	-vo
VIEW	= reverse	-vr
VIEW	= brief	-vb
VIEW	= long	-vt
VIEW	= comments	-vc

As in the view options specified on the command line, the last four options, reverse, brief, long and comments, can be combined with any of the sorting options listed above. For more detailed information on the configuration file, see Appendix B.

-x[t|T|n|s]

This option will extract files from a ZIP file. Either the entire ZIP file can be extracted or certain files can be specified to be extracted. The extracted files will be located in the current directory unless you specify a destination. The **-x** option is the default option for the UNZIP command. If no option is listed on the command **-x** will be assumed.

If a file being extracted already exists at the same destination where the extracted files are being located, the program will prompt you for confirmation before it overwrites the existing file.

```
unzip -x lists.zip DB/ *.dbf *.txt
```

In the above example, all of the files with a **.dbf** or a **.txt** extension will be extracted from the ZIP file called **lists**. The ZIP file is located on the current directory. When the noted files are extracted, they will be located in the **DB** directory.

Many suboptions are available to UNZIP's **-x** option. Use **"-xt"** to cause DOS files to be converted to UNIX text format, leaving binary files alone. (This overrides any setting in the **zip.cfg** file, discussed in Appendix B.) The option **"-xT"** forces text conversion on all files.

The suboption **"-xn"** is the opposite of the text conversion settings. It specifies that no text conversion should ever occur. Use this option to achieve the effect of a using a **zip.cfg** setting of **CONVERTTEXT=NONE**.

Note that UNZIP looks at the ZIP directory to determine if a file should be treated as text or binary. Use the view option, **"-vt"**, to see the settings within a given ZIP file. If you wish to always perform text conversion, use the option **"-xT"**. If you wish to always suppress text conversion, use the option **"-xn"**. If you wish to tell ZIP to use the setting within the ZIP file and to process it accordingly, use the option **"-xt"**. The **CONVERTTEXT** configuration file setting is used in the absence of any specification after the **"-x"** on the UNZIP command.

Specify **"-xs"** to cause spooler control codes to not be touched by UNZIP. Normally text conversion could alter these characters if you were uncompressing a file created for a spooler. This option will allow normal text conversion without touching the spooler codes.

-v[b][e][i]- This option allows you to control the verbosity of the output to your screen (the information that is output). You may want to suppress certain items if you are using a script, for example, to cut down on the amount of information sent to the user.

The option **-vb** shows the banner when the program is started.
-ve causes error messages to be displayed.
-vi causes informational level messages to be displayed.

Using a hyphen (-) disables an option.

Thus **-vb-** disables the banner.
-ve- disables error messages. The program sets error return codes to an appropriate values to indicate an error. (See Appendix D).
-vi- disables informational level message.
-v- disables banner, error and informational level messages.

The suboptions may be combined to provide the desired level of reporting; e.g. **-vb-i-** disables the banner and informational level messages. The default, if no options are specified, is **-vbei**.

Tutorial

This section will describe some simple examples, illustrating the command format for extracting files from a ZIP file.

A ZIP file is one which holds much information in a greatly reduced space. Use the UNZIP program to extract compressed files from a ZIP file. An entire ZIP file can be extracted, or specific files can be designated to be extracted.

There are five main parameters in the UNZIP command:

```
unzip [options[ ...]] zipfile [path/] [filename[ ...]]
```

- | | | |
|----|--|--|
| 1. | The actual program name. | unzip |
| 2. | The command <i>options</i> (optional). | letter commands
example: -v = view files |
| 3. | The ZIP file name | <i>zipfile</i> |
| 4. | The destination where you want the extracted information to be located (optional). | Default = current directory |
| 5. | Selected files to be extracted (optional) | Default = all files |

Step One / Extract All Files

In this exercise you will extract all files from the ZIP file **AnyFile.zip**.

Example: **unzip AnyFile.zip /tmpdsk/mydir/**

This command will locate the extracted files in the directory **/tmpdsk/mydir**. Note the use of the forward slash character (/) after **/tmpdsk/mydir**. The forward slash is used to indicate that **/tmpdsk/mydir** is a UNIX directory name, rather than the name of a ZIP file to be extracted.

Step Two / Extract Files Matching a Wildcard

In this example the ZIP file is located in the **TRANSFER** directory. The destination is the current directory by default. Only the ***.c** files will be extracted.

Example: **unzip TRANSFER/AnyFile.zip *.c**

Step Three / Overwrite Existing Files

In this example, the path location of the ZIP file is specified. The files will be extracted to the **MYDIR** directory.

Example: **unzip Collect/AnyFile.zip MYDIR/ -o**

With the **-o** option, the file with the same name as those existing in the **MYDIR** directory, will overwrite the already existing files, without asking for user confirmation.

Appendix A / Creating a Pre-Selected File List

This section describes how to create and use a file list with the ZIP and UNZIP programs.

What is a File List?

A file list is a shorthand method of recording file selections that you use frequently. Names of the files that you want to process are entered into a file called a *List File*. This *List File* can then be typed on the command line instead of typing all of the file names separately. *List Files* can be used with all the ZIP and UNZIP command options.

Tutorial

This section will describe some simple examples, showing the steps necessary for creating a *List File* and its use.

Step One / Startup Your Text Editor

List Files are simple text files that may be created using any text editor.

Step Two / Enter the File Names

Type in the names of the files that you want to select. If necessary, include the directory locations of the files.

```
Enter:      letters.txt
           /usr/lotus/*
           /usr/db/monthly.*
           /var/*.*WKS
```

Step Three / Save the List File

Save the *List File* under the name you choose. There is no default extension.

When using a *List File*, it is preceded by the @ symbol on the command line. If the *List File* is not in the directory, the @ symbol is put before the *path* designation. More than one list may be used on the command line.

The *List File* can be mixed with other options, examples are shown below.

```
unzip  -r  NewFile.zip    @Group
unzip  AnyFile.zip      Today.Txt    @Group  @Data
zip    -u  MoreFile     @Group  @/usr/tmp/Accounts
zip    -a  OldFile      *.Dbf     @OldStuff
```

Appendix B / The Configuration File

This section describes how to define the default runtime settings for the ZIP and UNZIP programs.

What is the Configuration File?

The configuration file can be used to set parameters to values other than the normal default values. These values will then be the defaults for the specified parameters. You can override any parameters that are set in the configuration file by entering contrary instructions on the command line.

ZIP and UNZIP will first look in the current directory for the configuration file called `zip.cfg`. If not found, then ZIP will look in the directory of the name `/etc` and will search that directory for `zip.cfg`.

IMPORTANT NOTE:

When you use the file `/etc/zip.cfg`, make sure all users of ZIP and UNZIP have read access to this file. This is essential if you wish the settings to be read for each user. Use a command such as

```
chmod a+r /etc/zip.cfg
```

after creating the file to set the proper permissions.

Configuration Options

This is a brief explanation of the configuration options for ZIP and their equivalent command line options. A checkmark (✓) indicates that this is the default for ZIP.

<u>Option</u>	<u>Values</u>	<u>Command Line</u>	<u>Default</u>
ANSI	= enabled	-q	
	= disabled	-q-.....	✓
CONVERTTEXT	= none		
	= text	-t.....	✓
	= all	-T	
ZIPDATE	= latest	-o	
	= keep	-k	
	= current	-k- -o-.....	✓
NAME	= pathname of mapping rules file	-N	

Note: See the discussion of file name mapping under ZIP or UNZIP.

INCLUDE	= hidden	-wh	
	= system	-ws	
EXCLUDE	= hidden	-WH.....	✓
	= system	-WS.....	✓

Note: *These options can be combined. (For example: INCLUDE = hidden/system.)*

MASK	= readonly	-jr	
	= hidden	-jh	
	= system	-js	
PASS	= readonly	-Jr.....	✓
	= hidden	-Jh.....	✓
	= system	-Js.....	✓

Note: *These options can be combined. (For example: MASK = hidden, system.)*

VIEW	= comments	-vc	
	= extension	-ve	
	= size	-vs	
	= date	-vd	
	= name	-vn	
	= ratio	-vp	
	= natural	-vo.....	✓
	= reverse	-vr	
	= brief	-vb	
	= long	-vt	

Note: *Some of these options can be combined. The last three options, reverse, brief and long, can be combined with any of the above options.*

PATHS	= none	-p-.....	✓
	= all	-P	
	= recurse	-p	

RECURSE	= on	-r	
	= off	-r-.....	✓

CONVERTNAMES	= on	-G-.....	✓
	= off	-G	

ALTERNATIVE	= pathname of temporary working directory	-b	
-------------	--	----	--

XATTRIB	= name	-An	
	= protect	-Ap	
	= dtm	-Am	
	= all	-Anpm	
	= none	✓

FILETYPE	= file	-Tf.....	✓
	= dir	-Td	
	= segdir	-Ts	
	= all	-Tdfs	

Note: If the *-rp* or *-rP* options are used, the *-Td* option is assumed.

COMPRESS	= size	-ex.....	✓
	= speed	-es	
	= shrink	-eS	
	= implode	-eX	

For some of the parameters, only one value can be specified. For example, ANSI can be either enabled or disabled. In other cases, more than one variable can be set per parameter. For example, INCLUDE can be set to either *system* and/or *hidden* files.

The parameters can be typed in uppercase, lowercase, or a combination. When more than one variable can be set per parameter, the variables can be listed on the same line, separated by a comma (,), slash (/), or space. For example, when setting the MASK parameter to include all readonly, hidden and system files to be masked any of the following methods would be acceptable.

```
MASK=readonly
MASK=hidden
MASK=system
```

```
MASK=readonly, hidden, system
```

```
MASK=readonly hidden system
```

```
MASK=readonly, hidden/system
```

Appendix C / Error Messages

This section lists the error messages that may be displayed while running the ZIP and UNZIP programs. When the word *Warning* appears in the message, program execution will continue, otherwise, the program will abort to UNIX.

ZIP Error Messages

The following error messages may appear when using ZIP. A brief explanation of each message follows.

ZIP: Can't open XXXX.ZIP for write access!

The named ZIP file is read-only or locked by another application and cannot be modified.

ZIP: No file(s) found.

No matching files were found to list using the View option.

ZIP: Insufficient disk space for updated files: XXXX.ZIP.

The **-b** option was used, and there is not enough space on the original file system containing the ZIP file to receive the updated ZIP file. Try to free up some space on the file system containing the ZIP file, and retry the operation. Also, make sure that the file system specified with the **-b** option is different than the file system containing the ZIP file.

ZIP: Insufficient disk space for ZIP comment.

There is insufficient disk space to hold the comment as entered.

ZIP: Warning! Not enough memory for Shrinking method.

The **-es** or **-ea** or **-eb** options were used to specify that certain files should be Shrunk. However, there is not enough memory available to perform Shrinking. Instead, all files will be Imploded.

ZIP: Warning! Can't delete XXXXX.

The **-m** (Move) option was specified to delete files after the ZIP file was constructed. However, the named file could not be deleted, and is probably read-only.

ZIP: No files specified for deletion!

The **-d** (Delete) option was specified, but no filenames were given to delete. This option does *not* default to ***** if no filenames are given.

ZIP: Nothing to do!

No matching files, or files with the specified attributes, or files after the specified date, etc., were found to compress.

ZIP: Insufficient memory.

Insufficient memory is available to process the ZIP file. Break the ZIP file into smaller ZIP files if possible. Also, the ZIP file might be corrupted: PKWARE, Inc. provides a program called PKZIPFIX which may be used to repair damaged ZIP files. Try using PKZIPFIX to fix the ZIP file.

ZIP: XXXX.ZIP - error in ZIP, use ZIPFIX.

The named ZIP file has a corrupted file index. Use PKWARE, Inc.'s PKZIPFIX on a PC to reconstruct the ZIP file.

ZIP: Can't create: XXXX.

The named file could not be created. Either the target directory is full or the file already exists and is read-only or locked by another application.

ZIP: Disk full, file: XXXX.

A disk full error occurred while writing to the specified file. Try freeing up some disk space on the target file system. Also, see the **-b** option.

ZIP: Can't find: XXXX.ZIP.

The named ZIP file could not be found.

ZIP: Too many files.

There are too many files to compress into one ZIP file. The limit is 4,000 files per ZIP file.

UNZIP Error Messages

The following error messages may appear when using UNZIP. An explanation of each message follows.

UNZIP: Warning! File XYZZY already exists. Overwrite (y/n)?

The file **XYZZY** already exists on the disk. Entering **n** will leave the original file on the disk, and not extract the file from the ZIP file. Entering **y** will extract the file from the ZIP file, and overwrite the file on the disk. Also, see the **-o** and **-n** options.

UNZIP: Warning! I don't know how to handle: XYZZY.

The file **XYZZY** is compressed or encoded in a way that this version of UNZIP is unable to handle. Either a later version of UNZIP is required to properly extract this file, or the ZIP file is possibly corrupted.

UNZIP: Warning! Inconsistent local header for file: XYZZY.

The local header for file **XYZZY** is different than the central header information. Use PKWARE, Inc.'s PKZIPFIX on a PC to reconstruct the ZIP file.

UNZIP: Warning! File fails CRC check.

The CRC-32 check for the file being extracted or tested did not match the stored value for the file. The file is probably corrupted.

UNZIP: Warning! XXXX.ZIP has errors.

The named ZIP file had one or more errors detected in it.

UNZIP: Warning! File has bad table.

The file being tested or extracted has an error in its encoding. The file is probably corrupt.

UNZIP: Warning! No file(s) found.

No files were found to extract, test, or list.

UNZIP: Warning! XXX.ZIP - error in ZIP, use ZIPFIX.

The named ZIP file has a corrupted file index. Use PKWARE, Inc.'s PKZIPFIX on a PC to reconstruct the ZIP file.

UNZIP: Warning! Insufficient memory.

Insufficient memory is available to process the ZIP file. If possible, when creating the ZIP file create several smaller files instead of one large ZIP file. Also, the ZIP file might be corrupted: use PKWARE, Inc.'s PKZIPFIX to fix the ZIP file.

UNZIP: Warning! Can't create: XXXXX.

The named file could not be created. The output directory is either invalid or full.

UNZIP: Can't find: XXXX.ZIP.

The named ZIP file(s) could not be found.

UNZIP: Can't open: XXXX

The specified file list could not be opened.

UNZIP: Warning! Can't open XXXX.ZIP

The named file could not be opened. Either a disk error occurred, or the file is locked by another application.

UNZIP: Disk full, file: XXXXX.

There is not enough free room on the destination directory for the file being extracted. The program will abort and exit to UNIX.

UNZIP: Incorrect password for file.

The ZIP file is password protected, and cannot be opened unless the correct password is entered. The password entered is not the correct password.

UNZIP: Skipping encrypted file.

Files that are passworded will be skipped (not extracted) because they are password protected and the correct password was not entered.

Appendix D / Returned Error Codes

Many ZIP and UNZIP users utilize scripts to control the operation of the ZIP and UNZIP commands. You may wish to use scripts to control the handling of error conditions. ZIP and UNZIP will set a normal return code upon termination. Your scripts or other programs can access this status code using the `$?` special variable. The codes used by ZIP and UNZIP are as follows:

-1	Warnings occurred.
0	No errors or warnings.
1	Usage error (not a ZIP file, etc).
2	Invalid command line options.
3	Invalid directive in zip.cfg file.
4	Corrupted ZIP file.
5	File system I/O error.
6	Internal program error.

Appendix E / Self Extracting ZIP Files (for DOS)

Sometimes you may wish to send a ZIP'ed file to a PC based user (running MS-DOS) who does not have PKUNZIP. Or perhaps you want to send such a file to a very non-technical user, and you do not wish to explain the use of PKUNZIP to the user. ZIP for Unix can create "self extracting" files. These are files that, when transferred to DOS, are real `.EXE` executable files. One simply runs the file to cause it to be automatically unzipped.

This is accomplished after you have ZIP'ed the file, using a special program. This program is discussed in this appendix.

`ZIP2EXE_CONV` is used to create a self-extracting ZIP file from a ZIP file. The original ZIP file is left intact. `ZIP2EXE_CONV` uses a DOS program called `sfx.prg` to create the self-extracting ZIP file. `sfx.prg` must be resident in the `/etc` directory when creating a self-extracting file. (This is installed for you when ZIP is installed).

A self-extracting ZIP file is an executable DOS program with an `.EXE` extension. By simply typing in the name of the self-extracting ZIP file at the DOS command prompt, the UNZIP extraction process will be performed.

Self-extracting ZIP files can be read as a normal ZIP file by the ZIP and UNZIP programs. The only difference is, the `.EXE` extension must be specified on the command line or the programs will search for a ZIP file with the same name. At this implementation of ZIP, update of self-extracting files is not allowed. To make modifications to the self-extracting file, create a new file using ZIP and convert to a self-extracting file using `ZIP2EXE_CONV`.

Please note that the PKWare utilities, `PKZIP` and `PKUNZIP`, available under DOS, do not support the self-extracting file formats created by `ZIP2EXE_CONV`. `PKZIP` and `PKUNZIP` cannot be used on these files.

The following example illustrates the process of creating a self-extracting ZIP file.

1. Create the ZIP file using ZIP.
2. Create the self-extracting file by entering:

```
ZIP2EXE_CONV zipfile
```

Where `zipfile` is the name of the ZIP file you want to create the self-extracting file from. The default extension is `.zip` if none is given.

When the process is completed a message similar to the one shown below will be displayed on the screen.

```
zipfile.ZIP => zipfile.EXE
```

DOS users may invoke the self-extracting file created in step 2 using the following options:

```
zipfile.EXE [options] [d:path\] [file...]
```

options Any of the options listed below.
d:path Output drive and/or path.
file Name(s) of files to extract. Wildcards *, ?, are ok. Default is all files.

Valid options are:

-c [m]	Extract to screen (with more)
-d	Create directories stored in ZIP file
-n	Extract only newer files
-p [a,b,c][1,2,3]	Extract to printer [ASC mode, Bin mode, COM port] [port number]
-o	Overwrite existing files
-x	Extract files (default)
-l	Display software license agreement
-t	Test ZIP file integrity

Appendix F / ZIP for Unix and ZIP for PRIMOS

Computronics markets a product to the Prime Computer 50-series market called ZIP for PRIMOS. The ZIP for Unix product is cross compatible with the PRIMOS version. If you have both products, and you are moving files to or from the Prime, ZIP will be a big help:

- Create a ZIP file on the Prime using the **-A** and **-T** options.
- Transfer this file to your Unix system.
- Use UNZIP on the Unix system, with the **-A** option. You will get the entire 32 character file name transferred, as well as the date/time modified, date/time accessed, etc. As many attributes as possible are retained when performing such a transfer.

- If you utilize the Vmark product Prime/Information on the 50-series, and PI/Open on the Unix system, you can also use ZIP on both systems to migrate your databases. Some additional conversions steps will be needed, but the **-T** and **-A** options on the Prime will simplify the migration to Unix.

- The reverse capabilities are also available. First, ZIP a set of files on your Unix system, with the attributes saved (via **-A**).
- Transfer this ZIP file to the Prime system.
- Now run UNZIP with **-A** and **-T** to get the files converted properly.

Appendix G / Common Problems

We attempt here to answer the most commonly asked ZIP and UNZIP questions. You may find your question answered here and save the trouble of calling Computronics. If your answer is not here, however, never hesitate to call! Help is only a phone call or fax away.

Question:

ZIP is slow. What can I do to speed it up?

Answer:

Are you always working with text files or with binary files? If so, using the **CONVERTTEXT=ALL** or **CONVERTTEXT=NONE** directives will save as much as 30% of the time utilized by a ZIP command. If you don't want to change the system defaults, place the **zip.cfg** file in your current directories. The programs look there before trying to find the file in **/etc**. If you don't want to use the configuration option, the ZIP command options **-aT** and **-an** will provide the same speedup.

Question:

I ZIP a file, move it to a PC, and then use PKUNZIP to unzip the file. The file looks like junk (smiley faces and other special characters)

Answer:

Your file has some binary characters in it, or you have forced text conversion off. Use the configuration file or the **-aT** option, as discussed above, to force text conversion to be performed on the Unix system.

Question:

I use PKZIP to compress a file on a PC, then move it to Unix, and run UNZIP on the file. The text looks strange on Unix, or is uneditable.

Answer:

The PC probably did not mark the file as a text file, but as a binary file. Use the **-vt** option on UNZIP to view the technical settings. If it is binary, force text conversion via **CONVERTTEXT=ALL** or via the **-xT** option.

Question:

How do ZIP and UNZIP decide if text conversion should be performed, in the absence of any special `T`, `n`, or configuration file option?

Answer:

The `ZIP` command will scan the file. If any unprintable characters occur other than a carriage return/line feed, the file will be considered binary. The only exception is that the presence of printer control codes will not change the status of the file if the `-as` option is used. Note that the entire file is scanned to make this determination; the scan will only stop early when it determines that a file is binary.

The `UNZIP` command uses the setting within the ZIP file to determine if text conversion should be performed. You may view this setting via the `-vt` (view technical) option on ZIP or UNZIP.

Question:

If I have ZIP on my Unix system, how do I get the compressed files to and from the IBM PC, Apple Macintosh, or other platform?

Answer:

There are many methods. Computronics will provide a copy of Kermit for Unix, if this helps you. There are many other methods. See the discussion of this topic in section 2 of this manual.

Question:

Can ZIP for UNIX work with PKZIP's revision 2.0?

Answer:

Yes. Make sure you have ZIP for Unix version 3.00 or later to get this new functionality. See the `-e` option on the ZIP command for further details.

(revision 21.0a/20-Sep-91)
(revision 22.1a/12-Mar-92)
(revision 3.00a/18-Apr-94)
(revision 3.00b/12-Dec-94)
(revision 3.01a/28-Dec-94)
(revision 4.05a/12-Apr-96)

Copyright © 1996 Computronics. All rights reserved.

UNIX is a trademark of UNIX System Laboratories.
IBM and IBM PS/2 are trademarks of International Business Machines Corporation.
Crosstalk is a trademark of Microstuff, Inc., Roswell, GA.

