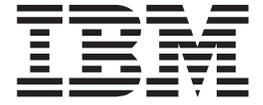
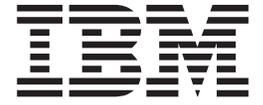


z/OS



UNIX System Services Command Reference: APAR OW52499

z/OS



UNIX System Services Command Reference: APAR OW52499

About This Guide

This document supports APAR OW52499 for UNIX System Services. This document is only available on the z/OS UNIX Systems Services web site at:
<http://ibm.com/servers/eserver/zseries/zos/unix/release/apar.html>

Chapter 1. pax — Interchange portable archives

Format

- **pax** [-cdnqvzE][-f *archive*] [-o *type*] [-s *substitute*] ... [*pattern* ...]
- **pax -r** [-cdiknuvzE] [-f *archive*] [-o *options* ...] [-p *string* ...] [-s *substitute* ...] [-V *volpat*] [*pattern* ...]
- **pax -w** [-diLqtuvXzE] [-b *blocksize*] [[-a] [-f *archive*]] [-o *options* ...] [-ssubstitute ...] [-V *volpat*] [-x *format*] [*pathname* ...]
- **pax -r -w** [-diklLnquvXE] [-p *string* ...] [-s *substitute* ...] [*pathname* ...] *directory*

Note

APAR OW52499 (OS/390 R6 - z/OS R4) changed the pax default behavior for extended USTAR attributes. Before the APAR is applied, **-o saveext** is the default and extended USTAR attributes are saved. After the APAR is applied, **-o noext** is the default and extended USTAR attributes are NOT saved. The APAR also introduces an environment variable (`_OS390_USTAR=Y`) which makes pax save the extended USTAR attributes as long as **-o noext** is not used. For more information on extended USTAR attributes, see “z/OS Extended USTAR Support” on page 12.

File Tag Specific Options:

- **pax -o fromfiletag**
- **pax -o setfiletag**

Description

pax reads, writes, copies, and lists archive files. An *archive file* is a single file containing one or more files and/or directories. Archive files can be HFS files or MVS datasets. A file stored inside an archive is called a *component file*; similarly, a directory stored inside an archive is called a *component directory*. **Rule** : MVS datasets cannot be specified for component files. Included with each component file and directory is recorded information such as owner and group name, permission bits, file attributes, and modification time.

You can therefore use a single archive file to transfer a directory structure from one machine to another, or to back up or restore groups of files and directories.

Archives created by **pax** are interchangeable with those created with the **tar** utility. Both utilities can read and create archives in the default format of the other (USTAR for **pax** and TAR for **tar**, **os390** for both). Additionally, OS390 formatted archives created by pax are interchangeable with those OS390 formatted archives created by the tar utility. Archives are generally named with suffixes such as .pax or .tar (or pax.Z and tar.Z for compressed files), but this is not required.

As shown in Format, **pax** performs one of the four archive functions based on the usage of the **-r** and **-w** options: list, read, write, or copy:

list If you do not specify **-r** or **-w**, you are in list mode. In this mode, **pax** uses the standard output to display the table of contents of an existing archive file. The **-v** (verbose) and **-E** options can be used to show the file attributes (to include file tags and ACLs) and extended attributes of each component.

pax

By default, **pax** displays all component files and directories contained in the archive. One or more patterns may be used to display information on specific components.

read If you specify **-r** but not **-w**, you are in read mode. In this mode, **pax** reads an archive file as input and extracts components from the archive. By default, **pax** selects all components. Patterns may also be used to identify specific components to extract. If the archive contains several components with the same name, **pax** extracts each of them with later components overwriting files created by earlier components with the same name. The **-k**, **-n**, or **-u** options can be used to control the extraction of files when multiple files with the same name exist in the archive or on the file system. **pax** can read input archives in **cpio**, **tar**, and **OS390** format.

When extracted, if a component does not have a fully qualified pathname beginning with the root (/) directory, its path is assumed to be relative to the current working directory. The **-s** or **-i** options can be used to dynamically change the pathnames of extracted components. Ownership, permissions, file attributes (such as file tags and ACLs), and extended attributes of the extracted files are discussed under the **-p** option.

write If you specify **-w** but not **-r**, you are in write mode. In this mode, **pax** creates an archive file that contains the specified pathnames as components. If a pathname is a directory, **pax** writes to the archive file all the files and subdirectories in that directory. If you do not specify any pathname, **pax** reads the standard input to get a list of pathnames to select; the input should give one pathname per line.

The **-d**, **-X**, and **-L** options can be used to restrict pathnames to the current directory or device, or to follow symbolic links.

The **-a** (with **-w**) option can be used to append to an existing archive.

copy If you specify both **-r** and **-w**, you are in copy mode. In this mode, **pax** reads the specified pathnames and copies them to the target directory. In this case, the given directory must already exist and you must be able to write to that directory. If a pathname is a directory, **pax** copies all the files and subdirectories in that directory as well as the directory itself. If you do not specify any pathname, **pax** reads the standard input to get a list of pathnames to copy; the input should give one pathname per line.

The name of the archive file can be specified with the **-f archive** option. If **-f** is not used, **pax** will read from standard input for the list and read (**-r**) functions and will write to standard output for the write (**-w**) function.

pax can read input archives in **cpio**, **tar**, and **os390** format. It can also write these formats; see the **-x** option.

Patterns

Command-line patterns are similar to the wildcard constructs described in the **sh** command. You can use them to select specific components when reading or listing an archive.

Slash character(s) in a pathname must be explicitly matched by using one or more slashes in the pattern; it cannot be matched by the asterisk (*) or question mark (?) special characters or by a bracket expression. For example, the pattern **"*.c"** will only match files in the archive with name that are not preceded by a slash. The pattern **"*/*.c"** will match files in the archive preceded by a single slash.

Tip : Patterns should be quoted to prevent the shell from first expanding them. For example, if the pattern *.h is not quoted, the shell will first resolve it into the list of files in the current directory ending with .h. If there are none, the shell will replace *.h with an empty list and **pax** will then list every component in the archive because no pattern is specified. If one or more .h files are returned by the shell, **pax** will list only those components in the archive matching the .h files found in the current directory.

pax does not support patterns when writing or copying — however, wildcards can be used in specifying the pathname with the write or copy function because the shell will first expand them before passing the results to **pax**.

The **-c** option can be used to select files that do not match the pattern.

Options

The following options can appear on **pax** command lines. Some of them are appropriate to only some forms of the command, as shown in “Format” on page 1.

-a Appends specified files or directories to the end of the contents of an existing archive. If the archive does not already exist, **pax** creates it.

Note: The following types of archives cannot be appended to:

- Compressed archives
- Archives residing in MVS partitioned datasets
- Cannot specify OS390 format to be appended to non-OS390 format archive
- Cannot specify non-OS390 format to be appended to OS390 format archive

-b *blocksize*

Specifies the block size in an output operation. Each output operation writes *blocksize* bytes, where *blocksize* is an integer appropriate to the output device. If b follows the *blocksize* number, the block size is the given number of 512-byte blocks. If k follows the *blocksize* number, the block size is the given number of 1024-byte blocks. The default *blocksize* is 10k for **tar** archives, 5k for **cpio** archives. The block size must be at least 512 bytes for reading.

-c Selects all those files that do *not* match any of the patterns given on the command line; this is the opposite of the usual behavior. If a pattern is not given, then no files will match.

-d Does not traverse directories. A pattern matching a directory extracts only the directory itself. When creating an archive, a directory name stores only the directory itself.

-E Same as verbose (**-v**) output, but additionally displays extended attributes. See “Output” on page 10 for more information. **-o E** is equivalent to **pax -E**.

-f *archive*

Lets you specify the name of the archive file instead of using the standard input for list mode, read mode (**-r** operations), and the standard output for write mode (**-w**). The archive file you specify may be an MVS dataset. For more information, see Appendix K.

pax

Note: Avoid writing to an archive which is in the directory tree or the set of files being archived. Doing so causes **pax** to write the archive to itself and results in unpredictable results during the write or later during a read.

-i Lets you rename files as **pax** works. With extractions, **pax** displays the name of the component it is about to extract and gives you the chance to specify a name for the extracted file. With write operations, **pax** displays the name of the file or directory it is about to record in the archive, and lets you specify a different name to be assigned to the component. If you enter . as the name, **pax** processes the file or directory with no change to the name. If you just press <Enter>, **pax** skips the file (doesn't extract or archive it). **pax** ends if you enter end-of-file.

If you also specify **-s**, **pax** makes the given substitution before displaying the name of the component.

-k Prevents the overwriting of existing files.

-L Follows symbolic links. When you specify this option, **pax** copies the file to which a symbolic link points to the archive. Normally, only the symbolic link is copied.

-l Is applicable only when you are in copy mode—that is, when you are using the **-rw** format to copy files to another directory. If you specify **-l**, **pax** creates links to the original files whenever possible, rather than copying them.

-n Treats the *pattern* arguments as ordinary pathnames. You can use this option only when you specify **-r** but not **-w**. **pax** extracts only the first component with a given pathname, even if the archive contains several components with the same name. **pax** checks the given pathnames against the archive before applying any renaming from the **-i**, or **-s** options. **pax** writes an error message for each specified file that cannot be found in the archive.

-o options

Provides information for modifying the algorithm for writing and extracting files.

The following set of options controls the use of z/OS extended USTAR support for the USTAR and OS390 format to preserve, restore, and display z/OS specific information such as external links, extended attributes, file tag information, ACLs, and other information (long link names, for example) not otherwise supported by the USTAR format. The OS390 format saves those z/OS specific attributes by default. For more information on extended USTAR support, see “z/OS Extended USTAR Support” on page 12.

Note: In addition to the below options, environment variable `_OS390_USTAR=Y` may be used to turn on the z/OS extended USTAR support for USTAR format.

-o A Displays extended ACL (access control list) data.

Tip: Specifying **pax -o A** does not automatically turn on the verbose table of contents format. You must also specify **-v** to display the file permission bit settings associated with the file.

For more information on ACLs, see *z/OS UNIX System Services Planning* and “ACL (Access Control List) pax Support” on page 14.

-o keyword=value[,keyword=value]...

These options are used to convert data from one code set to another while reading or writing an archive. This is functionally equivalent to using the **iconv** utility to convert each file before or after archiving. This option has the format: where *keyword* is either **to** or **from** and *value* is the name of a code set. Two common code set names and their values are:

ISO8859-1
ASCII
IBM-1047
EBCDIC

For example, to convert from ASCII to EBCDIC, use:

```
-ofrom=ISO8859-1,to=IBM -1047
```

From EBCDIC to ASCII, use:

```
-ofrom=IBM-1047,to=ISO8859-1
```

For a more complete list of code sets, refer to *z/OS C/C++ Programming Guide*.

Specifying an unknown keyword results in a warning message from **pax**.

You can omit either the **to** or **from** keyword. If you omit **to**, **pax** assumes that you want to write (or read) a portable archive tape and will convert the data to ISO/IEC 8859-1. If you omit **from**, **pax** assumes that you are converting from the system-specific local code set.

If your input contains a character that is not valid in the source code set, **pax** displays a warning and continues, leaving the character untranslated. If the source code set contains a character that is not in the destination code set, **pax** converts the character to an underscore (_).

Note: If you do not specify **-o**, no code set conversion is done. When making code set conversions, **pax** assumes that all files are text files, since only text files are portable.

-o saveext / noext

For USTAR and OS390 formatted archives, this option controls whether extended USTAR support is enabled (*saveext*) or disabled (*noext*). *noext* is the default behavior for USTAR format when writing an archive. The *saveext* is the default behavior for OS390 format when awriting an archive. The *saveext* is the default behavior when extracting or listing files from the archive. It is also the default to save extended attributes and external links. In order to list some attributes like ACLs or file tags, **-o A** and **-o T** option must be used. This option has no effect for non-USTAR. For more information on extended USTAR support, see “z/OS Extended USTAR Support” on page 12.

Table 1. USTAR Defaults

Pax Action	USTAR Default
writing, copying	-noext

Table 1. USTAR Defaults (continued)

extracting, listing	-saveext
---------------------	----------

saveext

During archive writing, *saveext* causes **pax** to preserve extended USTAR information. During archive listing, *saveext* causes **pax** to display extended USTAR information. During archive reading, *saveext* enables **pax** to restore extended USTAR information. To restore certain information, the user must also have the appropriate privileges and have specified the corresponding options. For example, in order to restore extended attributes, **-px** must be specified and to restore ACLs **-pA** must be specified. The external links and extended attributes are saved by default for USTAR and OS390 format. The file attributes requiring special headers, such as long links, file tags, and ACLs, need the **-o saveext** to be specified for USTAR (OS390 uses **-o saveext** by default). The environment variable `_OS390_USTAR=Y` may also be used to turn on the support. For more information on extended USTAR support, see “z/OS Extended USTAR Support” on page 12.

noext When creating archives, do not preserve extended USTAR information. When reading or listing an archive, ignore any extended USTAR support (such as extended attributes, long links, external links, file tags, and ACLs) encoded within the archive. If an archive contains z/OS special header files, these will be displayed or restored (or both) as regular files. Special header files are described in z/OS Extended USTAR Support. are described in “z/OS Extended USTAR Support” on page 12).

-o fromfiletag

See “File Tag Specific Options” on page 9.

-o setfiletag

See “File Tag Specific Options” on page 9.

-o type

-o type displays additional information when listing the contents of an archive. Only one type may be specified per **-o type** option. However, **-L type** may be specified multiple times. The types that can be displayed are:

o E Show extended attributes when displaying the archive table of contents. Automatically turns on **-v**. This is synonymous with the existing **pax -E** option.

-o T Displays file tag information. Similar to **ls -T** and **chtag** output. Does not automatically turn on verbose (**-v**) in the same way that **ls -T** does not automatically turn on its **-l** (long listing) option. When used without **-v**, only the file tag information and filenames are displayed. For example:

```
/tmp> pax -o T -f asciitagged.pax
m IS08859-1 T=off text_am
t IS08859-1 T=on text_at
- untagged T=off text_au
```

This option can be used with **-v** and/or **-o E** to display additional verbose output. For example:

```
/tmp> pax -T -vf asciitagged.pax
m IS08859-1 T=off -rw-r--r-- 1 SteveS Kings 9 Apr 30 22:31 text_am
t IS08859-1 T=on -rw-r--r-- 1 SteveS Kings 9 Apr 30 22:31 text_at
- untagged T=off -rw-r--r-- 1 SteveS Kings 9 Apr 30 22:06 text_au
```

-p *string*

Specifies which file characteristics to restore. By default, **pax** will only restore the access and modification times of each component file, and the access permissions (mode) as modified by the current umask, that is, they will only be restored entirely when the umask is 000. The file tag information, external links, and links whose target exceed 100 characters are also restored by default. *string* can consist of any combination of the following characters:

- A** Restores ACL data.
- a** Does not preserve file access times.
- e** Preserves the user ID, group ID, file mode, access time, and modification time, and all extended attributes and ACL entries.
- m** Does not preserve file modification times.
- o** Preserves the user ID and group ID.
- p** Preserves the file mode: access permissions (without modification by umask), set-user-ID bit, set-group-ID bit, and sticky bit.
- x** Preserves extended attributes.

-q For read mode only, **pax** assumes that all created files are text files and extracts them to the local text file format. On systems with fixed length records, this might mean appending blanks as padding.

On UNIX and POSIX-compliant systems, **pax** removes all carriage return characters (**\r**) and retains only the newline (**\n**) characters.

-r Reads an archive file from standard input.

-s *substitute*

Modifies pathnames using a substitution command *substitute*. This is similar to the substitution command of the **ed** text editor. The full option has the form:

```
-s#bregex/string/[gp]
```

where *bregex* is a basic regular expression and *string* is a string that **pax** is to insert in place of matches for the regular expression. *string* can contain an ampersand & (standing for the string matching *bregex*), or \1, \2, and so on (with the meanings defined in **regex**), for subexpression matching.

The # is used as the delimiter character separating *bregex* and *string*. You can use any non-null character instead. There cannot be any space between **-s** and the delimiter character.

Normally, **-s** replaces only the first match for *bregex*. A *g* following the *string* replaces all matches in the line.

A *p* following the *string* prints all successful substitutions on the standard error stream. **pax** displays a substitution in the format:

```
oldname >> newname
```

pax

There may be more than one **-s** option on the command line. In this case, **pax** tries the substitutions in the order given. **pax** stops trying to make these substitutions as soon as it makes its first successful substitution. If the null string replaces a filename, **pax** ignores that filename on both input and output.

- t** After reading files being archived, **pax** resets the access time to that prior to **pax**'s access.
- u** Compares component dates to dates of existing files with the same name. When extracting components with **-r** (read mode), **pax** extracts a file only if its modification date is more recent than the modification date on an existing file of the same name. In other words, it doesn't overwrite an existing file if the existing file is newer than the one in the archive.

Similarly, when copying files with **-rw** (copy mode), **pax** does not overwrite an existing file if the existing file is newer than the one being copied.

In a command that uses **-w** but not **-r** (write mode), **-u** checks to see if the file being added has the same name as a file already in the archive. If so, and if the file being added is newer than the one in the archive, **pax** leaves the old file in the archive and appends the new one at the end. In this case, **-u** automatically implies **-a**, which means that **pax** adds new files to the end of the archive.

-V *volpat*

Provides automatic multivolume support. **pax** writes output to files the names of which are formatted with *volpat*. It replaces any occurrence of # in *volpat* with the current volume number. When you invoke **pax** with this option, it asks for the first number in the archive set, and waits for you to type the number and a carriage return before proceeding with the operation. **pax** issues the same sort of message when a write error or read error occurs on the archive; the reasoning is that this kind of error means that **pax** has reached the end of the volume and is to go on to a new one. An interrupt at this point ends **pax**.

- v** Lists pathnames on the standard error stream just before beginning to process the files or directories, but after any **-i**, or **-s** options have had their effect. In list mode (neither **-r** nor **-w** is specified), **pax** displays a "verbose" table of contents; this verbose format shows information about the components in the same format used by the **ls** command. See "Output" on page 10 for more information.

- w** Writes files to the standard output in the specified archive format.

- X** Writes out only those files that are on the same device as their parent directory.

-x *format*

Specifies a file format for an output archive. The *format* argument can be:

- cpio** Standing for the ASCII format used by the **cpio** command.
- cpioB** Standing for the binary format used by **cpio**.
- os390** Standing for the OS390 format which has all the support for saving/restoring extended USTAR support such as special headers, external links, and long links. This format is only supported on OS/390 and z/OS systems.
- tar** Standing for the old format of **tar** files.
- ustar** Standing for the USTAR format used by the **tar** command.

The default *format* is *ustar*. For more information about file formats, see Appendix H.

Tip : In order to preserve information about extended attributes, external links, and link names greater than 100 characters, USTAR format with either `_OS390_USTAR=Y` environment variable or `-o saveext` option must be used. It may also be saved using the `-x os390` option for OS390 format. See the `-o saveext` option for more information about preserving extended attributes.

-z For write or read mode, performs Lempel-Ziv compression. **-z** cannot be used when appending (**-a**) to an existing archive.

For writes, **-z** is functionally equivalent to creating a normal archive and then compressing it using the `compress` utility.

It is recommended that, when creating archive files using the **-f** option, the archive name be suffixed with a `.Z` to identify it as a compressed file and to facilitate it being processed by `uncompress` (if needed).

For reads, **-z** is functionally equivalent to first uncompressing the archive using the `uncompress` utility and then reading it. This option is not required when reading a compressed archive. **pax** will automatically detect that the archive is compressed. It may be useful, however, to use **-z** to confirm that the archive is compressed (you will receive an error message if you specify **-z** on an archive that is not compressed).

File Tag Specific Options

-o fromfiletag

For use with `-o from=,to=`. Use of **-o fromfiletag** indicates that if a component file has a CCSID assigned to it, use that CCSID as the `from= codeset`, thereby, overriding the value specified on `-o from=,to=`.

-o setfiletag

For use with `-o from=,to=`. Using **-o fromfiletag** tags component files that are not already tagged. If a file is untagged (TXTFLAG = OFF, CCSID = 0), then it will be automatically stored with TXTFLAG = ON and with CCSID = to the target codeset. For files which are already **not** untagged, **-o setfiletag** will not change the default behavior. The target codeset and TXTFLAG values will be left as-is. For example, a file tagged as `mixed` will have TXTFLAG = OFF and CCSID \neq 0. UNIX will not automatically force TXTFLAG = ON because it does not want to override the user's reason for making the file `mixed`.

When the currently supported `-o from=,to=` option is used to perform translation, the default behavior for storing the file tag information on writes, reads and copies will be as follows:

-w (write)

- For files that are **not** untagged (TXTFLAG = OFF and CCSID \neq 0), the CCSID preserved in the archive will be set to the CCSID of the `to=codeset` argument. Files that are untagged (TXTFLAG = OFF and CCSID = 0) will not have file tag information stored. The **-o setfiletag** option can be used to force the tagging of files which are not already tagged.
- When a file in the archive is tagged with a different CCSID than the `from=codeset`, an error message will be generated. However,

pax

pax will continue processing. Because this situation indicates a probable corruption of data, upon completion, **pax** will issue a nonzero return code. The **-o fromfiletag** option can be used to avoid this situation. It causes **pax** to use the CCSID of the file rather than the one specified on the **-o from=,to=** option.

-r (read)

- For files that are **not** tagged, the TXTFLAG value will be restored to the value preserved in the archive (ON or OFF), but the CCSID of the target file will be altered to the *to=codeset* CCSID. For example, a file tagged as *mixed* will have TXTFLAG = OFF and CCSID ≠ 0. UNIX will not automatically force TXTFLAG = ON because it does not want to override the user's reason for making the file mixed.

The default behavior for files in the archive that are untagged will not change, and the target file will also be set to untagged. The **-o setfiletag** option can be used to force the tagging of files which do not have filetag information associated with them in the archive.

- If the target file already exists, its filetag information will be ignored.
- When a file in the archive is tagged with a different CCSID than the *from=codeset*, an error message will be generated. However, **pax** will continue processing. Because this situation indicates a probable corruption of data, upon completion, **pax** will issue a non-zero return code. The **-o fromfiletag** option can be used to avoid this situation. It causes **pax** to use the CCSID of the file rather than the one specified on the **-o from=,to=** option.

-wr (copy)

- If the source files is not untagged, then the target file will have its CCSID set to the CCSID of the *to=codeset* CCSID. In the event the target already exists, then its TXTFLAG values are ignored; the source file is used to determine the TXTFLAG setting of the target and will override whatever the TXTFLAG settings are of the target.
- Like **-r** and **-w**, when the CCSID of the source file is different from the *from=codeset* CCSID, a warning message will be generated and upon completion, **pax** will issue a nonzero return code. The **-o fromfiletag** option can be used to avoid this situation. It causes **pax** to use the CCSID of the file rather than the one specified on the **-o from=,to=** option.

Output

When the **-v** or **-E** option is used in list mode, **pax** produces a verbose table of contents for the archive. The output for **-v** is similar to the output from the **ls-l** command with the following exceptions:

- The notation:

pathname == linkname

indicates that *linkname* is a hard link of *pathname*.

- For symbolic and external links, **pax** output always shows a filesize of 0.

Refer to the description of **ls** for an explanation of **ls -l**.

The output from the **-E** option has the same format as **-v** and additionally displays a column showing the extended attributes:

- a** Program runs APF-authorized if linked AC=1
- p** Program is considered program-controlled
- s** Program runs in a shared address space
- l** Program is loaded from the shared library region.

Note: **l** is a lower-case L, not an upper-case i.
 - attribute not set

The format of the **pax -E** output is variable in length and will be extended as necessary to display additional file characteristics that are not supported by **pax -v** (**ls -l**).

Usage Notes

1. On the z/OS system, superuser privileges or read access to the appropriate FACILITY classes are required to create character special files, restore user and group names, and to set certain extended attributes (read access to the corresponding FACILITY classes).
2. The POSIX 1003.1 standard defines formats for **pax**, **tar**, and **cpio** archives that limit the UIDs and GIDs that can be stored to the following maximum values:

Table 2. Max UID/GID Values

Format	Max UID/GID Values
tar, USTAR	16777216
cpio	262143

Values larger than these will not be properly restored for **tar** and **cpio** formatted archives. For USTAR formatted archives, because the user and group names are also stored in the archive, the correct UID and GID will be restored only if the name is defined on the target system.

3. The POSIX 1003.1 standard defines formats for **pax** and **tar** archives that limit the length of the target of a link file to 100 characters or less.

Note: In the case of a hard link, the target is the first occurrence of the hard link which is archived. Subsequent hard links refer to the first instance. Beginning with OS/390 Release 6, **pax** and **tar** provide extended USTAR support that allows these links to be preserved when creating an archive and restored when reading an archive. They may also be saved and restored using the OS390 archive format. See “z/OS Extended USTAR Support” on page 12 for more information.

4. When transferring archives between z/OS and other UNIX systems, note the following:
 - a. File transfers (for example, using OPUT/OGET or ftp put/get) must be done using binary or image format. This is true, even for archives consisting only of text files.
 - b. You may need to convert text files from EBCDIC to ASCII (or some other character set). The **pax -o** option can be used to convert text files while an archive is being created or being restored. You can use the **iconv** utility to convert files before they are stored in the archive or after restoring them from an archive.
5. Automatic conversion (performed by the HFS) on files with file tag information is disabled when:

pax

- reading files during creation of an archive
- during writes while extracting files from an archive

That is, the settings of system and environment variables that turn automatic conversion on and off will have no effect on **pax**'s reading and writing of files.

z/OS Extended USTAR Support

OS390 archive format stores all the extended USTAR attributes by default (-o options do not apply). By default, the IBM z/OS implementation of **pax** and **tar** provide extended support with the USTAR format to preserve and restore z/OS specific file attributes and other information not otherwise supported due to limitations with the USTAR format. The OS390 format also stores these by default. Examples of these include:

- External links
- Extended file attributes (such as program-controlled, APF-authorized, etc.)

This support is only provided for archives using the USTAR format. USTAR is the default format for **pax** when creating an archive. For **tar**, the default format is the original **tar** format. The **-U** option, however, can be used to cause **tar** to use USTAR. When reading an archive, **tar** will automatically recognize the USTAR format— no special option is required. (For more information about the USTAR format, see "tar -- Format of tar archives" in Appendix H.)

The **pax** and **tar** commands also allow the storing/restoring of additional file attributes using explicit options and/or environment variable. The following attributes require special header support as explained below. OS390 format stores/restores these by default. Examples of these additional attributes include:

- Links whose targets exceed 100 characters
- Access Control Lists (ACLs)
- File tag information

The extended USTAR support is provided by using two mechanisms: encoding the information within the USTAR header record and through the creation of special header files. (The same mechanism is used for the OS390 archive format.)

Encoding Information within the USTAR Header Record

External link and extended attribute information is encoded within the standard USTAR header in a manner which is compliant with POSIX standards and should be tolerated by other non-z/OS versions of **pax** and **tar**. Because external links and extended attributes are specific to z/OS, however, they cannot be restored on other platforms.

Special Header Files

Hard links and symbolic links with targets greater than 100 characters cannot be preserved within the standard USTAR format (for a hard link, the target is the first occurrence of the hard link which is archived; subsequent hard links refer to the first instance). In order to preserve links with targets greater than 100 characters, *special header files* are created for each link and stored in the archive. The special headers are stored when one of the following is used: **-o saveext** option, environment variable **_OS390_USTAR=Y**, or **-x os390** option (OS390 format).

Each special header file contains information used by z/OS **pax** and **tar** to restore the link to its original state. Special header files are identified in the archive with type "S" (see "tar -- Format of tar archives" in Appendix H for more information about file types).

Each special header file in the archive will have the same name: **/tmp/OS390_USTAR_SUMMARY_timestamp** where *timestamp* is the creation time (represented in seconds since the epoch) of the first special header file. For example:

```
/tmp/OS390_USTAR_SUMMARY_919026474
```

When a special header file is required to preserve a file and the `_OS390_USTAR=Y` environment variable was used, an informative message along with a reason is displayed indicating that a special header file was created. When `-o saveext` or `-x os390` for **pax** or `-UX` or `-S` for **tar** is used, the informational message is not displayed.

When reading or listing an archive containing special header files and when using the default extended USTAR support, **pax** and **tar** recognize type "S" files as special header files and display or restore the file described by the special header rather than the actual special header file. So, typically, the presence of special header files is not known to the user.

When the archive is complete, if one or more special header files have been created, then a final *special header summary file* is created and added to the archive. This file is identified in the archive with type "T" and has the same name as the special header files. This file summarizes, via script commands and comments, the contents of all previously archived special header files. Its primary purpose is to provide help restoring files included via special header files to those with versions of **pax** or **tar** that do not implement extended USTAR support.

So, to allow users of non-z/OS systems to read the *special header summary file*, it is encoded in the ASCII ISO8859-1 codeset. To view the special header file in EBCDIC code page IBM-1047, first convert the file using the **iconv** command. For example:

```
iconv -f ISO8859-1 -t IBM-1047 /tmp/OS390_USTAR_SUMMARY_919026474 >
summary_in_ebcdic
```

If extended USTAR support is disabled during reading or listing an archive by using the **pax -o noext** or the **tar -O** option, or if the archive is processed by either an earlier version of z/OS **pax** or **tar** that does not implement extended USTAR support or a non-z/OS system version of **pax** or **tar**, then the special header files will not be recognized and will be processed as unknown type regular files. During extraction, because all files have the same name, each extracted special header file will overlay the previous one with the special header summary file being the final one restored.

For an example of the *special header summary file*, see POSIX.

File Tags and the Use of `-o noext`

Because special headers are required to store file tag information, the storing and restoring of file tag information is disabled if the user specifies the `-o noext` option. The `-o noext` option is the default for writing an archive. To store information in the special headers, the `-o saveext` or `_OS390_USTAR=Y` environment variable must

pax

be used. When `-o noext` is used, each file is treated as if it were untagged. That is, if `-o noext` is specified, the stored or extracted file will be set to untagged regardless of its previous file tag setting.

`-o noext` disables all attributes stored with special headers, so this option cannot be used to selectively disable the storing or restoring of text flag information. You will have to use `chtag` to do that.

`-o noext` will have no affect on automatic conversion of files. If you use `pax` to read, write or copy files, automatic conversion will be disabled whether `-o noext` is specified or not.

For more information on automatic conversion and file tagging, see *z/OS UNIX System Services Planning*.

ACL (Access Control List) pax Support

Archive Writing or Creating

ACL data is stored in USTAR formatted archives using special headers when one of the following is used: `-o saveext` option or `_OS390_USTAR=Y` environment variable. OS390 format (`-x os390` option) automatically stores all special header information to include ACLs.

You can use `pax -o noext` to disable the creation of special headers. This prevents `pax` from storing ACL data and other non-standard information such as file tag data and long link names. However, there is no option to disable storing of ACL data only.

Archive Reading or Restoring

By default, ACL data will not be restored when reading or restoring files from an archive. However, you can use `pax -p A` to restore ACL data. You can also use `pax -p e` (which restores all file attributes) to restore ACL data.

Archive Copying

If you need to preserve ACLs when copying files to an archive, you must use `pax -p A` or `pax -p e`.

Archive Listing (Table of Contents)

For verbose output (`pax -v`), a `+` is added to the end of the file permission bits for all files with extended ACL entries. For more information about access control lists, see *z/OS UNIX System Services Planning*.

Examples

Archive Listing (Table of Contents):

If `file2` and `dir1` have extended ACL entries:

```
> pax -vf acldata.pax
-rwx----- 1 STIERT SHUT      294912 Nov  9 09:57 file1
-rwx-----+ 1 STIERT SHUT      294912 Nov  9 09:57 file2
drwxr-xr-x+ 2 STIERT SHUT         8192 Mar 20 2000 dir1/
```

Writing (creating) an archive:

1. The following creates an archive file named **/tmp/files.pax** from all the files in the current working directory. The **-v** option is used to display each file as it is being added:

```
pax -wvf /tmp/files.pax *
```

or

```
pax -wvf /tmp/files.pax .
```

The only difference between these two forms is that in the latter example (using **.**), names recorded in the archive will be preceded by a **"/"**.

2. Either of these commands creates a compressed version of the archive created in Example 1:

```
pax -wzvf /tmp/files.pax.Z *
```

or

```
pax -wzvf /tmp/files.pax.Z .
```

The above is equivalent to the following two commands:

```
pax -wvf /tmp/files.pax *
compress /tmp/files.pax
```

3. The following creates an archive **/tmp/dironly.pax** containing only the files and directory names in the current directories (it does not include the contents of subdirectories):

```
pax -wdvf /tmp/dironly.pax. *
```

4. This example creates the archive **/tmp/cfiles.pax** containing all c files in the current directory:

```
pax -wvf /tmp/cfiles.pax *.c
```

5. This example creates the archive **/tmp/allcfiles.pax** containing all c files in the current directory and all sub-directories:

```
pax -wvf /tmp/allcfiles.pax $(find . -name "*.c")
```

6. This example creates the archive **/tmp/ascii_src.pax** using all .c and .h files in the current directory converted into ASCII:

```
pax -wv -o to=IS08859-1 -f /tmp/ascii_src.pax *. [ch]
```

7. The following creates the compressed archive **/u/smith/oldfiles.pax.Z** containing all files on the system that have not been accessed within the last 10 days:

```
pax -wzvf /u/smith/oldfiles.pax.Z $(find / -type f -atime +10)
```

8. The following creates the archive **/tmp/basename.pax** containing all files in the directory **sub1** stored in the archive with **"sub1/"** removed from each component name. Note that the pound character **#** is being used as the delimiter for the **-s** option:

```
pax -wv -s#sub1/## -f /tmp/basename.pax sub1/*
```

Reading an archive:

1. This example extracts all the components of the archive **source.pax**. The **-v** option is used to display each file or directory as it is extracted.

```
pax -rvf source.pax
```

2. To extract all files in **source.pax** and translate them from ASCII to EBCDIC:

```
pax -ofrom=IS08859-1,to=IBM-1047 -rf source.pax
```

3. To extract all files in the archive **source.pax** ending with **.h**:

pax

```
pax -rf source.pax `pax -f source.pax | grep h$`
```

This example uses command substitution to first read the archive and generate a list of all files in the archive that end with */*.

4. This example extracts files into a directory that is different from the directory they are stored in within the archive. Assume the names of all files stored in the archive **source.pax** begin with the root directory (*/*). To extract them into **/newroot/**, use the following command:

```
pax -rvf source.pax -s##/newroot/#
```

The **-v** option is used to show the names of the files as they are extracted and is not required.

5. In the following examples, archive **acldata.pax** contains *file1*, *file2*, and *dir1*. *file1* has no ACL data, *file2* has an access ACL, and *dir1* has a file default ACL, a directory default ACL and an access ACL. If you only specify option **-f**, your output will be:

```
> pax -f acldata.pax
file1
file2
dir1
```

If you also specify **-o A**, ACL information will be displayed:

```
> pax -o A -f acldata.pax
file1
file2
user:WELLIE2:rw-
group:SYS1:rwX
dir1
```

Finally, if you add the verbose option, **-v**, you will see the file permission bit settings that are associated with the file:

```
> pax -o A -vf acldata.pax
-rwx----- 1 STIERT SHUT      294912 Nov  9 09:57 file1
-rwx-----+ 1 STIERT SHUT      294912 Nov  9 09:57 file2
user:WELLIE2:rw-
group:SYS1:rwX
drwxr-xr-x+ 2 STIERT SHUT          8192 Mar 20  2000 dir1/
user:RRAND:rwX
user:WELLIE2:rw-
group:SHUT:rwX
fdefault:user:RRAND:rwX
fdefault:group:SHUT:r-x
default:user:ANGIEH:rwX
default:group:SYS1:r--
```

Tip : Specifying **pax -o A** does not automatically turn on the verbose table of contents format. You must also specify **-v** to display the file permission bit settings associated with the file. To check if a file has an ACL when for example, *file2* and *dir1* have ACLs :

```
> pax -vf acldata.pax
-rwx----- 1 STIERT SHUT      294912 Nov  9 09:57 file1
-rwx-----+ 1 STIERT SHUT      294912 Nov  9 09:57 file2
drwxr-xr-x+ 2 STIERT SHUT          8192 Mar 20  2000 dir1/
```

For more information about access control lists, see

To store a file with an ACL using the OS390 archive format:

```
> pax -o os390 -wf acldata.pax fileAcIs
```

Files

`/tmp/OS390_USTAR_SUMMARY_`

timestamp is an z/OS extended USTAR special header file. See “z/OS Extended USTAR Support” on page 12 for more information.

Localization

pax uses the following localization environment variables:

- **LANG**
- **LC_ALL**
- **LC_COLLATE**
- **LC_CTYPE**
- **LC_MESSAGES**
- **LC_TIME**
- **LC_SYNTAX**
- **NLSPATH**

See Appendix F for more information.

Exit Values

- | | |
|----------|--|
| 0 | Successful completion |
| 1 | Failure due to any of the following: <ul style="list-style-type: none">• Incorrect option• Incorrect command-line arguments• Out of memory• Compression error• Failure on extraction• Failure on creation |

If **pax** cannot extract a particular file when reading, or find a particular file when writing, it generates an error message and continues to process other files but returns a status of 1. If any other sort of error occurs, **pax** ends immediately without attempting further processing.

If you see the following message after a write operation:

```
If you want to go on, type device/filename when ready
```

it indicates that your directory or device containing the archive file is full. To continue, enter the name of a new directory; to end **pax**, type <Ctrl-C>.

If you see that message after a read operation, it means that **pax** could not find the archive file you specified, or that it was damaged. In this case, type <Ctrl-C> to end the operation and then restart **pax** with the correct archive name.

Portability

POSIX.2, X/Open Portability Guide.

The **-L**, **-q**, **-V**, **-E**, **-p x** and **-z** options are extensions of the POSIX standard.

Related Information

compress, cpio, ls, tar, uncompress

See Appendix C for more information about **regexp**.

See the **cpio** and **pax** file format descriptions in Appendix H.

Chapter 2. tar — Manipulate the tar archive files to copy or back up a file

Format

```
tar -c[#sbvwlzOUXS] -f file [-V volpat] [tarfile] [blocksize] [-C pathname]
tar -r[#sbvwlzOUXS] -f file [-V volpat] [tarfile] [blocksize]
tar -t[#sbvzEOUXS] -f file [-L type] [-V volpat] [tarfile] [blocksize] [-C
pathname]
tar -x[#AsbvwpmozOUXS] -f file [-V volpat] [tarfile] [blocksize]
```

Note

APAR OW52499 (OS/390 R6 - z/OS R4) changed the tar default behavior for extended USTAR attributes. Before the APAR is applied, for **tar -U**, the **-X** is the default (extended USTAR attributes are saved). After the APAR is applied, for **tar -U**, the **-O** is the default (extended USTAR attributes are NOT saved). The APAR also introduces an environment variable (`_OS390_USTAR=Y`) which makes **tar -U** save the extended USTAR attributes as long as **-O** is not used. For more information on extended USTAR attributes, see pax utility under “z/OS Extended USTAR Support” on page 12.

Description

tar reads, writes and lists archive files. An *archive file* is a single file containing one or more files and/or directories. Archive files can be HFS files or MVS datasets. A file stored inside an archive is called a *component file*; similarly, a directory stored inside an archive is called a *component directory*.

Rule : MVS datasets cannot be specified for component files.

Included with each component file and directory is recorded information such as owner and group name, permission bits, file attributes, and modification time. You can therefore use a single archive file to transfer a directory structure from one machine to another, or to back up or restore groups of files and directories.

Archives created by **tar** are interchangeable with those created with the **pax** utility. Both utilities can read and create archives in the default format of the other (USTAR for **pax** and TAR for **tar**). To save extended USTAR attributes, the USTAR format (**-U**) must be used with **-X** option. Also the OS390 format may be used using the **-S** option. In general, the USTAR format with **-X** option and OS390 format records the most information and is recommended. Archives are generally named with suffixes such as .pax or .tar (or pax.Z and tar.Z for compressed files), but this is not required.

Table 3. Recommended USTAR format options

Intent	Option
To save only standard attributes	tar -U
To save all attributes to be restored on z/OS system	tar -S
To save all attributes to be restored on z/OS and non-z/OS systems	tar -UX

tar

Tip : In order to preserve information such as extended attributes, external links, ACLs, file tag information, and links whose targets exceed 100 characters, either the USTAR format (-U) and -X option or the OS390 format using the -S option must be used. See the -U option for selecting the USTAR format. The -O and -X options and “z/OS Extended USTAR Support” on page 12 contain information about enabling and disabling USTAR support.

You cannot use **tar** unless you specify **-f**.

Options

The four forms of the command shown in the syntax represent the main functions of **tar** as follows:

-c Creates an archive. This command writes each named file into a newly created archive. Directories recursively include all components. Under the USTAR (-U) option, **tar** records directories and other special files in the tape archive; otherwise, it ignores such files. If **-** appears in place of any filename, **tar** reads the standard input for a list of files one per line. This allows other commands to generate lists of files for **tar** to archive.

Tip : In order to preserve information about extended attributes and external links, the USTAR format (-U) must be used. Additionally, to preserve ACLs, file tag information, and link names greater than 100 characters, the USTAR format (-U) and -X option must be used. The OS390 archive format may also be used with the -S option to store all the file attributes.

-r Writes the named files to the end of the archive. It is possible to have more than one copy of a file in a tape archive using this method. To use this form of the command with a tape, it must be possible to backspace the tape.

Rules : You cannot specify both the **-r** and the **-z** option at the same time. Also, you cannot specify OS390 format to be appended to non-OS390 format archive or specify non-OS390 format to be appended to OS390 format archive.

-t Displays a table of contents. This displays the names of all the files in the archive, one per line. If you specify one or more files on the command line, **tar** prints only those filenames. The verbose (-v) and (for USTAR formatted archives or OS390 formatted archives) the **-E** options can be used to show the attributes and extended attributes of each component.

-x Extracts files from an archive. **tar** extracts each named file to a file of the same name. If you did not specify any files on the command line, all files in the archive are extracted. This extraction restores all file system attributes as controlled by other options.

You must specify one of the preceding basic options as the first character of an option string. You can add other characters to the option string. Unlike with other commands, you must give options as a single string; for example, you might specify **-tv**, but you cannot separate them, as in “**-t -**.” You can omit the leading dash **-** if you want. Other possible options in the option string are:

A Restores ACL information when used with **-x** option.

b Sets the number of 512-byte blocks used for tape archive read/write operations to *blocksize*. The *blocksize* argument must be specified, and *blocksize* can be specified only when **b** is in the option string. When reading from the tape archive, **tar** automatically determines the blocking

factor by trying to read the largest permitted blocking factor and using the actual number read to be the *blocksize*. For UNIX compatibility, the largest valid block size is 20 blocks; in USTAR mode, it is 60 blocks.

-C *pathname*

Is an unusual option because it is specified in the middle of your file list. When **tar** encounters a **-C** *pathname* option while archiving files, it changes the working directory (for **tar** only) to *pathname* and treats all following entries in your file list (including another **-C**) as being relative to *pathname*.

-E

Although still supported for compatibility with previous versions of **tar**, this option has been replaced by **-L E**.

f

You must specify **f**. The **f** option uses the file *tapefile* for the tape archive rather than using the default. The *tapefile* argument must be specified, and *tapefile* can be specified only when **f** is in the option string. The *tapefile* argument must precede the *blocksize* argument if both are present. If *tapefile* is the character `-`, the standard input is used for reading archives, and the standard output is used for writing archives.

#s

#s is not supported by. The default archive filename used by **tar** is **/dev/mt/0m**. This option is the least general way to override this default. For a more general method, see the **f** option. The filename generated by this option has the form **/dev/mt/#s**. The **#** can be any digit between 0 and 7, inclusive, to select the tape unit. The density selector *s* can be 1 (low), m (medium), or h (high).

I

Complains if all links are not resolved when adding files to the tape archive.

-L *type*

-L displays additional information when listing the contents of an archive. Only one type may be specified per **-L** option. However, **-L** may be specified multiple times. The types that can be displayed are:

A Displays extended ACL (Access Control List) data.

Note: Specifying **tar -L A** does not automatically turn on the verbose table of contents format. You must also specify **-v** to display the `chmod` settings associated with the file.

For more information on ACLs, see *z/OS UNIX System Services Planning* and “ACL (Access Control List) tar Support” on page 24.

E Same as verbose (**-v**) output, but additionally displays extended attributes. See “Output” on page 23 for more information. **-L E** is equivalent to the **tar -E**

T Displays file tag information. Does not automatically turn on the verbose **-v** option but can be used with **-v** or any other combination of table of contents display options. See “Output” on page 23 for more information.

m

Does not restore a file’s modification time stamp when extracting it from an archive. The default behavior is to restore the time stamp from information contained in the archive.

o

When writing files to an archive, does not record owner and modes of directories in the archive. If this is specified when extracting from an existing **ar** archive, **tar** does not restore any owner and group information in the archive. The default is to record this information when creating a **tar** archive, and to restore it when extracting from the archive.

-O

For USTAR formatted archive, this option turns off the extended USTAR

tar

support. **-O** is the default and user needs to use **-X** option to turn on extended USTAR support for USTAR archive.

For more information, see “z/OS Extended USTAR Support” on page 12.

-X For USTAR formatted archives, **-X** enables extended USTAR support. This option has no affect for non-USTAR formatted archives. **tar -X** functions in the following manner:

- During archive writing, **-X** causes **tar** to preserve extended USTAR information.
- During archive listing, **-X** causes **tar** to display extended USTAR information. This is the default; **-O** may be used to disable extended USTAR support.
- During archive reading, **-X** enables **tar** to restore extended USTAR information. This is the default; **-O** may be used to disable extended USTAR support.

The environment variable `_OS390_USTAR=Y`, also turns on the extended USTAR information

Tip : To restore certain information, the user must also have the appropriate privileges and have specified the corresponding options. For example, you must specify **-p** to restore extended attributes and to restore ACLs.

For more information on extended attributes, see “z/OS Extended USTAR Support” on page 12.

p When extracting, restores the three high-order file permission bits, exactly as in the archive. They indicate the set-user-ID, set-group-ID, and sticky bit. For USTAR formatted archives, **p** also restores, if present, extended attributes and **-A** restores ACLs.

Tip : If **-O** is specified, it overrides **-p** for extended attributes- they will not be restored. **tar** restores the modes exactly as stored in the archive and ignores the UMASK.To use **-p** on UNIX systems, you must have appropriate privileges; **tar** restores the modes restored exactly as in the archive and ignores the UMASK.

U When creating a new tape archive with the **-c** option, forces **tar** to use the USTAR format. The default format used when creating a new archive is the original UNIX **tar** format. When you do not specify **-c**, **tar** can deduce whether the tape archive is in USTAR format by reading it, so you can use **U** to suppress a warning about USTAR format.

In order to save external links, extended attributes, and file tag information, and ACLs, the extended USTAR format must be used. To turn on the extended USTAR format, the **-U** and **-X** options must be specified. The OS390 format may also be used (**-S** option) to save all the file attributes by default.

v Displays each filename, along with the appropriate action key letter as it processes the archive. With the **-t** form of the command, this option gives more detail about each archive member being listed and shows information about the members in the same format used by the **ls -l** command. You can also use the **-L type** option which provides the ability to display additional information such as extended attributes and file tag information. See “Output” on page 23 for more information.

-V volpat

Provides automatic multivolume support. **tar** writes output to files—the

names of which are formatted with *volpat*. Any occurrence of # in *volpat* is replaced by the current volume number. When you invoke **tar** with this option, it prompts for the first number in the archive set, and waits for you to type the number and a carriage return before proceeding with the operation. **tar** issues the same sort of message when a write error or read error occurs on the archive; this kind of error means that **tar** has reached the end of the volume and should go on to a new one.

- w** Is used to confirm each operation, such as replacing or extracting. **tar** displays the operation and the file involved. You can then confirm whether you want the operation to take place. Typing in an answer that begins with “y” tells **tar** to do the operation; anything else tells **tar** to go on to the next operation.
- z** Reads or writes, or both reads and writes, the tape archive by first passing through a compression algorithm compatible with that of **compress**.

Note: You cannot specify both the **-r** option and the **-z** option at the same time.

Output

When the **-v** or **-L E** (or **-E**) option is used with **-t** (table of contents), **tar** produces a verbose table of contents for the archive. The **-L T** option can also be used to additionally, or without the verbose output, display file tag information. The output for **-v** is similar to the output from the **ls -l** command with following exceptions:

- The following notation is used to represent hard, symbolic, and external links:

```
hlink external link to origfile
```

indicates that hlink is a hard link of origfile.

```
slink symbolic link to origfile
```

indicates that slink is a symbolic link to origfile.

```
elink external link to ORIG.FILE
```

indicates that elink is an external link to ORIG.FILE.

- For symbolic and external links, **pax** output always shows a filesize of 0

Refer to the description of **ls** for an explanation of the **ls -v**.

The output from the **-L E** (or **-E**) option has the same format as **-v** and additionally displays a column showing the extended attributes:

```
a      Program runs APF-authorized if linked AC=1
p      Program is considered program-controlled
s      Program runs in a shared address space
l      Program is loaded from the shared library region
```

Note: l is a lower-case L, not an upper-case i.

```
-      attribute not set
```

The format of the **tar -L E** (or **-E**) output is variable in length and will be extended as necessary to display additional file characteristics that are not supported by **tar -v (ls -l)**.

tar

The format of the **tar -L T** output is similar to the output from **chtag -p**. If specified with **-v** or **-L E**, the output will be displayed on the same line of and before the **-v** output. When used without **-v**, only the file tag information and filenames are displayed. For example:

```
/tmp> tar -L T -tf asciitagged.tar
m IS08859-1 T=off text_am
t IS08859-1 T=on text_at
- untagged T=off text_au
```

This option can be used with **-v** and/or **-o E** to display additional verbose output. For example:

```
/tmp> tar -L T -tvf asciitagged.tar
m IS08859-1 T=off -rw-r--r-- 1 SteveS Kings 9 Apr 30 22:31 text_am
t IS08859-1 T=on -rw r--r-- 1 SteveS Kings 9 Apr 30 22:31 text_at
- untagged T=off -rw-r--r-- 1 SteveS Kings 9 Apr 30 22:06 text_au
```

ACL (Access Control List) tar Support

Archive Writing or Creating

ACL data is stored in USTAR formatted archives, when **-X** option is used. The OS390 format (**-S** option) also stores the ACL information.

tar -O can be used to disable the creation of special headers. This prevents **tar** from storing ACL data and other non-standard information such as file tag data and long link names. However, there is no option to disable storing of ACL data only.

Archive Reading or Restoring

By default, ACL data will not be restored when reading or restoring files from an archive. However, for USTAR and OS390 formatted archives, you can use **tar -A** to restore ACL data.

Archive Listing (Table of Contents)

For verbose output (**tar -v**), **+** is added to the end of the file permission bits for all files with extended ACLs. For example, *file2* and *dir1* have extended ACL entries:

```
> tar -tvf acldata.tar
-rwx----- 1 STIERT SHUT 294912 Nov 9 09:57 file1
-rwx-----+ 1 STIERT SHUT 294912 Nov 9 09:57 file2
drwxr-xr-x+ 2 STIERT SHUT 8192 Mar 20 2000 dir1/
```

For more information about access control lists, see *z/OS UNIX System Services Planning*.

Usage Notes

1. Use the **pax** command if you need to use multibyte patterns when searching for filenames.
2. The POSIX 1003.1 standard defines formats for **pax** and **tar** archives that limit the length of the target of a link file to 100 characters or less.

Note: In the case of a hard link, the target is the first occurrence of the hard link which is archived. Subsequent hard links refer to the first instance. Beginning with OS/390 Release 6, **pax** and **tar** provide extended USTAR support and the OS390 format that allows these links to be preserved when

creating an archive and restored when reading an archive. See “z/OS Extended USTAR Support” on page 12 for more information.

3. On the z/OS system, superuser privileges or read access to the appropriate FACILITY classes are required to create character special files, to restore user and group names, and to set certain extended attributes.
4. Pathnames in the tape archive are normally restricted to a maximum length of 100 bytes. However, in USTAR (-U) and OS390(-O) format, pathnames can be up to 255 bytes long.
5. When transferring archives between z/OS and other UNIX systems, note the following:
 - a. File transfers (for example, using OPUT/OGET or ftp put/get) must be done using binary or image format. This is true, even for archives consisting only of text files.
 - b. You may need to convert text files from EBCDIC to ASCII (or some other character set). You can use the **iconv** utility to convert files before or after archiving. When text files are being created or extracted, you can use the **pax -o** option to convert them.
6. Automatic conversion (performed by the HFS) on files with file tag information is disabled when:
 - reading files during creation of an archive
 - during writes while extracting files from an archive

That is, the settings of system and environment variables that turn automatic conversion on and off will have no affect on **tar**'s reading and writing of files. **pax** supports file tag options which support conversion of files based on their file tag settings.

Examples

1. The following command takes a directory and places it in an archive in compressed format:


```
tar -cvzf archive directory
```
2. To identify all files that have been changed in the last week (7 days), and to archive them to the **/tmp/posix/testpgm** file, enter:


```
find /tmp/posix/testpgm -type f -mtime -7 | tar -cvf testpgm.tar -
```

-type -f tells **find** to select only files. This avoids duplicate input to **tar**.
3. In the following examples, archive **acidata.tar** contains file1, file2, and dir1. *file1* has no ACL data, *file2* has an access ACL, and *dir1* contains a file default, a directory default, and an access ACL. If you only specify option **-f**, your output will be:


```
> tar -f acidata.tar
file1
file2
dir1
```

If you also specify **-L A**, ACL information will be displayed:

```
> tar -L A -f acidata.tar
file1
file2
user:WELLIE2:rw-
group:SYS1:rwx
```

Finally, if you add the verbose option, **-v**, you will see the chmod settings associated with the file:

tar

```
> tar -L A -vf acldata.tar
-rwx----- 1 STIERT  SHUT      294912 Nov  9 09:57 file1
-rwx-----+ 1 STIERT  SHUT      294912 Nov  9 09:57 file2
user:WELLIE2:rw-
group:SYS1:rw-
drwxr-xr-x+ 2 STIERT  SHUT      8192 Mar 20 2000 dir1/
user:RRAND:rw-
user:WELLIE2:rw-
group:SHUT:rw-
fdefault:user:RRAND:rw-
fdefault:group:SHUT:r-x
default:user:ANGIEH:rw-
default:group:SYS1:r--
```

Localization

tar uses the following localization environment variables:

- **LANG**
- **LC_ALL**
- **LC_MESSAGES**
- **NLSPATH**

See Appendix F for more information.

Exit Values

- 0** Successful completion
- 1** Failure due to any of the following:
 - Incorrect option
 - Incorrect command-line arguments
 - Out of memory
 - Compression error
 - Failure on extraction
 - Failure on creation

Portability

4.2BSD

The **-U** option is an extension to provide POSIX USTAR format compatibility. The **-p** option is a common extension on BSD UNIX systems that is not available on UNIX System V systems. The **-O**, **-X**, and **-S** options are also extensions of POSIX standard.

Related Information

cpio, **pax**

Also see the **pax** file format description in Appendix H for more information.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM	OS/390	zSeries
AIX	Parallel Sysplex	
BookManager	RACF	
C/MVS	Resource Link	
C++/MVS	RISC System/6000	
C/370	SOM	
DFSMS/MVS	SOMobjects	
ESCON	SP	
IBM	SP2	
IBMLink	S/390	
Infoprint	Sysplex Timer	
Language Environment	System/370	
Library Reader	TalkLink	
MVS/ESA	VTAM	
Open Class	z/OS	
OS/2	z/OS.e	

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others:

ANSI American National Standards Institute

BSD
IEEE
Lotus
POSIX

University of California
Institute of Electrical and Electronics Engineers
Lotus Development Corporation
Institute of Electrical and Electronics Engineers

Acknowledgments

InterOpen Shell and Utilities is a source code product providing POSIX.2 (Shell and Utilities) functions to the z/OS UNIX services offered with MVS. InterOpen/POSIX Shell and Utilities is developed and licensed by Mortice Kern Systems (MKS) Inc. of Waterloo, Ontario, Canada.

Readers' Comments — We'd Like to Hear from You

z/OS

UNIX System Services Command Reference: APAR OW52499

Publication No. SA22-7802-04

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>				

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>				
Complete	<input type="checkbox"/>				
Easy to find	<input type="checkbox"/>				
Easy to understand	<input type="checkbox"/>				
Well organized	<input type="checkbox"/>				
Applicable to your tasks	<input type="checkbox"/>				

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



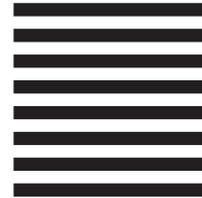
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY
12601-5400



Fold and Tape

Please do not staple

Fold and Tape



Program Number: 5694-A01, 5655-G52

Printed in U.S.A.

SA22-7802-04

