

NAME

`archive_write_set_format_cpio`, `archive_write_set_format_pax`,
`archive_write_set_format_pax_restricted`, `archive_write_set_format_shar`,
`archive_write_set_format_shar_dump`, `archive_write_set_format_ustar` — func-
tions for creating archives

LIBRARY

Streaming Archive Library (libarchive, -larchive)

SYNOPSIS

```
#include <archive.h>

int
archive_write_set_format_cpio(struct archive *);

int
archive_write_set_format_pax(struct archive *);

int
archive_write_set_format_pax_restricted(struct archive *);

int
archive_write_set_format_shar(struct archive *);

int
archive_write_set_format_shar_dump(struct archive *);

int
archive_write_set_format_ustar(struct archive *);
```

DESCRIPTION

These functions set the format that will be used for the archive.

The library can write POSIX octet-oriented cpio format archives, POSIX-standard “pax interchange” format archives, traditional “shar” archives, enhanced “dump” shar archives that store a variety of file attributes and handle binary files, and POSIX-standard “ustar” archives. The pax interchange format is a backwards-compatible tar format that adds key/value attributes to each entry and supports arbitrary filenames, linknames, uids, sizes, etc. “Restricted pax interchange format” is the library default; this is the same as pax format, but suppresses the pax extended header for most normal files. In most cases, this will result in ordinary ustar archives.

RETURN VALUES

These functions return **ARCHIVE_OK** on success, or **ARCHIVE_FATAL**.

ERRORS

Detailed error codes and textual descriptions are available from the `archive_errno()` and `archive_error_string()` functions.

SEE ALSO

`tar(1)`, `libarchive(3)`, `archive_write(3)`, `archive_write_set_options(3)`, `cpio(5)`,
`mtree(5)`, `tar(5)`