

NAME

archive_clear_error, **archive_compression**, **archive_compression_name**, **archive_copy_error**, **archive_errno**, **archive_error_string**, **archive_file_count**, **archive_format**, **archive_format_name**, **archive_set_error** — libarchive utility functions

SYNOPSIS

```
#include <archive.h>

void
archive_clear_error(struct archive *);

int
archive_compression(struct archive *);

const char *
archive_compression_name(struct archive *);

void
archive_copy_error(struct archive *, struct archive *);

int
archive_errno(struct archive *);

const char *
archive_error_string(struct archive *);

int
archive_file_count(struct archive *);

int
archive_format(struct archive *);

const char *
archive_format_name(struct archive *);

void
archive_set_error(struct archive *, int error_code, const char *fmt, ...);
```

DESCRIPTION

These functions provide access to various information about the struct archive object used in the libarchive(3) library.

archive_clear_error()

Clears any error information left over from a previous call. Not generally used in client code.

archive_compression()

Returns a numeric code indicating the current compression. This value is set by **archive_read_open()**.

archive_compression_name()

Returns a text description of the current compression suitable for display.

archive_copy_error()

Copies error information from one archive to another.

archive_errno()

Returns a numeric error code (see **errno(2)**) indicating the reason for the most recent error return.

archive_error_string()

Returns a textual error message suitable for display. The error message here is usually more specific than that obtained from passing the result of **archive_errno()** to **strerror(3)**.

archive_file_count()

Returns a count of the number of files processed by this archive object. The count is incremented by calls to `archive_write_header` or `archive_read_next_header`.

archive_format()

Returns a numeric code indicating the format of the current archive entry. This value is set by a successful call to `archive_read_next_header()`. Note that it is common for this value to change from entry to entry. For example, a tar archive might have several entries that utilize GNU tar extensions and several entries that do not. These entries will have different format codes.

archive_format_name()

A textual description of the format of the current entry.

archive_set_error()

Sets the numeric error code and error description that will be returned by `archive_errno()` and `archive_error_string()`. This function should be used within I/O callbacks to set system-specific error codes and error descriptions. This function accepts a printf-like format string and arguments. However, you should be careful to use only the following printf format specifiers: “%c”, “%d”, “%jd”, “%jo”, “%ju”, “%jx”, “%ld”, “%lo”, “%lu”, “%lx”, “%o”, “%u”, “%s”, “%x”, “%%”. Field-width specifiers and other printf features are not uniformly supported and should not be used.

SEE ALSO

`archive_read(3)`, `archive_write(3)`, `libarchive(3)`, `printf(3)`

HISTORY

The `libarchive` library first appeared in FreeBSD 5.3.

AUTHORS

The `libarchive` library was written by Tim Kientzle (kientzle@acm.org).