NAME

dwarf_finish, **dwarf_object_finish** - free resources associated with a debug descriptor

LIBRARY

DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

#include <libdwarf.h>

int
dwarf_finish(*Dwarf_Debug dbg*, *Dwarf_Error *err*);

int
dwarf_object_finish(*Dwarf_Debug dbg*, *Dwarf_Error *err*);

DESCRIPTION

The **dwarf_finish**() and **dwarf_object_finish**() functions are used to release the resources associated with a debug descriptor allocated by a prior call to dwarf_init(3) and dwarf_object_init(3) respectively.

Argument *dbg* denotes a valid *Dwarf_Debug* instance. Argument *err* will be used to record error information in case of an error.

After a call to **dwarf_finish**() or **dwarf_object_finish**(), the argument *dbg* will be invalid and should not be used further.

For *Dwarf_Debug* descriptors opened using dwarf_init(3), the application would need to explicitly release the *Elf* instance associated with the descriptor by first retrieving the instance using dwarf_get_elf(3) and closing it using elf_end(3).

RETURN VALUES

These functions return DW_DLV_OK if successful. In case of an error, the functions return DW_DLV_ERROR and record additional information in argument *err*.

EXAMPLES

To deallocate a *Dwarf_Debug* instance allocated using dwarf_elf_init(3) use:

    Dwarf_Debug dbg;
    Dwarf_Error de;

    if (dwarf_finish(dbg, &de) != DW_DLV_OK)

        errx(EXIT_FAILURE, "dwarf_finish: %s", dwarf_errmsg(de));

To deallocate a *Dwarf_Debug* instance allocated using dwarf_object_init(3) use:

    Dwarf_Debug dbg;
    Dwarf_Error de;

    if (dwarf_object_finish(dbg, &de) != DW_DLV_OK)
            errx(EXIT_FAILURE, "dwarf_object_finish: %s",
                dwarf_errmsg(de));

To deallocate a *Dwarf_Debug* instance allocated using dwarf_init(3) use:

    Dwarf_Debug dbg;
    Dward_Error de;
    Elf *e;

    if (dwarf_get_elf(dbg, &e, &de) != DW_DLV_OK)
            errx(EXIT_FAILURE, "dwarf_get_elf: %s", dwarf_errmsg(&de));

    if (dwarf_finish(dbg, &de) != DW_DLV_OK)
            errx(EXIT_FAILURE, "dwarf_finish: %s", dwarf_errmsg(de));

    (void) elf_end(e);

SEE ALSO
   dwarf_elf_init(3), dwarf_get_elf(3), dwarf_init(3), dwarf_object_init(3), elf_end(3)