

NAME

dwarf_init, **dwarf_elf_init** - allocate a DWARF debug descriptor

LIBRARY

DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

```
#include <libdwarf.h>
```

int

```
dwarf_init(int fd, int mode, Dwarf_Handler errhand, Dwarf_Ptr errarg, Dwarf_Debug *ret,  
           Dwarf_Error *err);
```

in

```
dwarf_elf_init(Elf *elf, int mode, Dwarf_Handler errhand, Dwarf_Ptr errarg, Dwarf_Debug *ret,  
              Dwarf_Error *err);
```

DESCRIPTION

These functions allocate and return a *Dwarf_Debug* instance for the object denoted by argument *fd* or *elf*. This instance would be used for subsequent access to debugging information in the object by other functions in the DWARF(3) library.

For function **dwarf_init**(), argument *fd* denotes an open file descriptor referencing a compilation object. Function **dwarf_init**() implicitly allocates an *Elf* descriptor for argument *fd*.

For function **dwarf_elf_init**(), argument *elf* denotes a descriptor returned by **elf_begin**(3) or **elf_memory**(3).

Argument *mode* specifies the access mode desired. It should be at least as permissive as the mode with which the file descriptor *fd* or the ELF descriptor *elf* was created with. Legal values for argument *mode* are:

DW_DLC_RDWR Permit reading and writing of DWARF information.

DW_DLC_READ Operate in read-only mode.

DW_DLC_WRITE Permit writing of DWARF information.

Argument *errhand* denotes a function to be called in case of an error. If this argument is NULL then a default error handling scheme is used. See **dwarf**(3) for a description of the error handling scheme used by the DWARF(3) library.

Argument *errarg* is passed to the error handler function denoted by argument *errhand* when it is invoked.

Argument *ret* points to the memory location that will hold a *Dwarf_Debug* reference on a successful call these functions.

Argument *err* references a memory location that would hold a *Dwarf_Error* descriptor in case of an error.

Memory Management

The *Dwarf_Debug* instance returned by these functions should be freed using **dwarf_finish()**.

IMPLEMENTATION NOTES

The current implementation does not support access modes `DW_DLC_RDWR` and `DW_DLC_WRITE`.

RETURN VALUES

These functions return the following values:

[`DW_DLV_OK`] This return value indicates a successful return.

[`DW_DLV_ERROR`] The operation failed.

[`DW_DLV_NO_ENTRY`] The object specified by arguments *fd* or *elf* did not contain debug information.

EXAMPLES

To initialize a *Dwarf_Debug* instance from a open file descriptor referencing an ELF object, and with the default error handler, use:

```
Dwarf_Error err;
Dwarf_Debug dbg;

if (dwarf_init(fd, DW_DLC_READ, NULL, NULL, &dbg, &err) !=
    DW_DLV_OK)
    errx(EXIT_FAILURE, "dwarf_init: %s", dwarf_errmsg(err));
```

SEE ALSO

`dwarf(3)`, `dwarf_errmsg(3)`, `dwarf_finish(3)`, `dwarf_get_elf(3)`, `elf_begin(3)`, `elf_memory(3)`