

NAME

dwarf_def_macro - add a macro definition to a DWARF producer instance

LIBRARY

DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

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

int

```
dwarf_def_macro(Dwarf_P_Debug dbg, Dwarf_Unsigned lineno, char *name, char *value,  
                Dwarf_Error *err);
```

DESCRIPTION

Function **dwarf_def_macro()** adds a macro definition to a DWARF producer instance.

Argument *dbg* should reference a DWARF producer instance allocated using **dwarf_producer_init(3)** or **dwarf_producer_init_b(3)**.

Argument *lineno* specifies the line number of the source line where the macro is defined. A line number of zero is used for macros that are defined before any source file is read.

Argument *name* should point to a NUL-terminated string containing the name of the macro. For function-like macros this parameter should also include parentheses and parameter names if any.

Argument *value* should point to a NUL-terminated string containing the value of the macro. If the macro does not have a value, argument *value* should be set to NULL.

If argument *err* is not NULL, it will be used to store error information in case of an error.

RETURN VALUES

On success, function **dwarf_def_macro()** returns DW_DLV_OK. In case of an error, function **dwarf_def_macro()** returns DW_DLV_ERROR and sets the argument *err*.

EXAMPLES

To record the fact that a macro named `_STDIO_H_` was defined at line 20 of the current macro file, use:

```
Dwarf_P_Debug dbg;  
Dwarf_Error de;
```

```
/* ... Assume 'dbg' refers to a DWARF producer instance... */  
if (dwarf_def_macro(dbg, 20, "_STDIO_H_", NULL, &de) != DW_DLX_OK)  
    errx(EXIT_FAILURE, "dwarf_def_macro failed: %s",  
        dwarf_errmsg(-1));
```

ERRORS

Function **dwarf_def_macro()** can fail with:

[DW_DLE_ARGUMENT] Either arguments *dbg* or *name* was NULL.

[DW_DLE_MEMORY] An out of memory condition was encountered during the execution of the function.

SEE ALSO

dwarf(3), dwarf_end_macro_file(3), dwarf_producer_init(3), dwarf_producer_init_b(3),
dwarf_start_macro_file(3), dwarf_undef_macro(3), dwarf_vendor_ext(3)