

**NAME**

**dwarf\_get\_macro\_details** - retrieve macro information

**LIBRARY**

DWARF Access Library (libdwarf, -ldwarf)

**SYNOPSIS**

**#include <libdwarf.h>**

*int*

**dwarf\_get\_macro\_details**(*Dwarf\_Debug* dbg, *Dwarf\_Off* offset, *Dwarf\_Unsigned* max\_count, *Dwarf\_Signed* \*entry\_cnt, *Dwarf\_Macro\_Details* \*\*details, *Dwarf\_Error* \*err);

**DESCRIPTION**

Function **dwarf\_get\_macro\_details**() retrieves information about macros associated with a DWARF debug context. Information about macro entries are returned as an array of descriptors of type *Dwarf\_Macro\_Details*, with each *Dwarf\_Macro\_Details* descriptor describing one macro information entry.

Argument *dbg* should reference a DWARF debug context allocated using **dwarf\_init**(3). Argument *offset* is an offset, relative to the ".debug\_macinfo" section, to the start of the desired macro information. Argument *max\_count* specifies the maximum number of macro information entries to be returned, or 0 if all entries are to be returned. Argument *entry\_cnt* should point to a location that will be set to the number of entries actually returned. Argument *details* should point to a location that will be set to a pointer to an array of *Dwarf\_Macro\_Details* descriptors. If argument *err* is not NULL, it will be used to store error information in case of an error.

*Dwarf\_Macro\_Details* descriptors are defined in the header file <libdwarf.h>, and consist of the following fields:

<i>dmd_offset</i>	The section-relative offset within the ".debug_macinfo" section of the macro information entry being described.
<i>dmd_type</i>	The type code of this macro information entry; one of the DW_MACINFO_* constants defined by the DWARF specification.
<i>dmd_lineno</i>	The line number associated with the macro information entry, or 0 if there is no applicable line number.
<i>dmd_fileindex</i>	The source file index for the macro information entry. This field is only meaningful when <i>dmd_type</i> field is set to DW_MACINFO_start_file.
<i>dmd_macro</i>	The contents of this field is a pointer to a NUL-terminated string whose meaning depends on the value of the <i>dmd_type</i> field: <div style="margin-left: 2em;">DW_MACINFO_define      The returned string contains the macro name and value.</div>

DW\_MACINFO\_undef      The string holds the macro name.  
 DW\_MACINFO\_vendor\_ext The *dmd\_macro* field points to a vendor defined string.  
 The field is NULL for other values of *dmd\_type*.

### Memory Management

The memory area used for the array of *Dwarf\_Macro\_Details* descriptors returned in argument *details* is owned by the DWARF Access Library (libdwarf, -ldwarf). The application should not attempt to directly free this pointer. Portable code should instead use **dwarf\_dealloc()** with the allocation type DW\_DLA\_STRING to indicate that the memory may be freed.

### RETURN VALUES

Function **dwarf\_get\_macro\_details()** returns DW\_DLV\_OK when it succeeds. It returns DW\_DLV\_NO\_ENTRY if there is no more macro information at the specified offset *offset*. In case of an error, it returns DW\_DLV\_ERROR and sets the argument *err*.

### EXAMPLES

To loop through all the macro information entries associated with a DWARF debug context:

```
Dwarf_Debug dbg;
Dwarf_Unsigned offset;
Dwarf_Signed cnt;
Dwarf_Macro_Details *md;
Dwarf_Error de;

offset = 0;
while (dwarf_get_macro_details(dbg, offset, 0,
    &cnt, &md, &de) == DW_DLV_OK) {
    for (i = 0; i < cnt; i++) {
        /* Access fields of md[i] ... */
    }
    offset = md[cnt - 1].dmd_offset + 1;
}
```

### ERRORS

Function **dwarf\_get\_macro\_details()** can fail with:

[DW\_DLE\_ARGUMENT] One of the arguments *dbg*, *entry\_cnt* or *details* was NULL.

[DW\_DLE\_NO\_ENTRY] There is no more macro information at the specified offset *offset*.

**SEE ALSO**

dwarf(3), dwarf\_dealloc(3), dwarf\_find\_macro\_value\_start(3), dwarf\_init(3)