NAME

dwarf_get_abbrev_entry - retrieve attribute information from an abbreviation descriptor

LIBRARY

DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

#include <libdwarf.h>

int
dwarf_get_abbrev_entry(*Dwarf_Abbrev abbrev*, *Dwarf_Signed ndx*, *Dwarf_Half *code*, *Dwarf_Signed *form*, *Dwarf_Off *offset*, *Dwarf_Error *err*);

DESCRIPTION

Function **dwarf_get_abbrev_entry**() retrieves attribute information from a DWARF abbreviation descriptor.

Argument *abbrev* should be a valid abbreviation descriptor, as returned by function dwarf_get_abbrev(3).

Argument *ndx* specifies the 0-based index of the attribute. The total count of the attributes contained in the abbreviation entry can be retrieved using the function dwarf_get_abbrev(3).

Argument *code* should point to a location which will hold a returned attribute code.

Argument *form* should point to a location which will hold the returned form of the attribute.

Argument *offset* should point to a location which will hold a returned offset, relative to the ".debug_abbrev" section, for the specified attribute.

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

RETURN VALUES

Function **dwarf_get_abbrev_entry**() returns DW_DLV_OK when it succeeds. It returns DW_DLV_NO_ENTRY if the attribute index specified by argument *ndx* is out of range. In case of an error, it returns DW_DLV_ERROR and sets the argument *err*.

EXAMPLES

To loop through all the attribute entries contained in the abbreviation section, use:

```
Dwarf_Debug dbg;
Dwarf_Abbrev ab;
Dwarf_Off aboff, atoff;
Dwarf_Signed form;
Dwarf_Half attr;
Dwarf_Unsigned length, attr_count;
Dwarf_Error de;
int i, ret;

/* ...allocate ’dbg’ using dwarf_init(3) ... */

while ((ret = dwarf_next_cu_header(dbg, NULL, NULL, &aboff,
    NULL, NULL, &de)) ==  DW_DLV_OK) {
        while ((ret = dwarf_get_abbrev(dbg, aboff, &ab, &length,
            &attr_count, &de)) == DW_DLV_OK) {
                if (length == 1)        /* Last entry. */
                        break;
                aboff += length;
                for (i = 0; (Dwarf_Unsigned) i < attr_count; i++) {
                        if (dwarf_get_abbrev_entry(ab, i,
                            &attr, &form, &atoff, &de) != DW_DLV_OK) {
                                warnx("dwarf_get_abbrev_entry failed:"
                                    " %s", dwarf_errmsg(de));
                                continue;
                        }
                        /* .. use the retrieved information ... */
                }
        }

        if (ret != DW_DLV_OK)
                warnx("dwarf_get_abbrev: %s", dwarf_errmsg(de));
}

if (ret == DW_DLV_ERROR)
        warnx("dwarf_next_cu_header: %s", dwarf_errmsg(de));
```

ERRORS

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

[DW_DLE_ARGUMENT] One of the arguments *abbrev*, *code*, *form* or *offset* was NULL.

[DW_DLE_NO_ENTRY] The attribute index specified by argument *ndx* was out of range.

SEE ALSO

dwarf(3), dwarf_get_abbrev(3)