

**NAME**

**dwarf\_formref**, **dwarf\_global\_formref** - retrieve offsets for REFERENCE class attributes

**LIBRARY**

DWARF Access Library (libdwarf, -ldwarf)

**SYNOPSIS**

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

*int*

```
dwarf_formref(Dwarf_Attribute attr, Dwarf_Off *retoffset, Dwarf_Error *err);
```

*int*

```
dwarf_global_formref(Dwarf_Attribute attr, Dwarf_Off *retoffset, Dwarf_Error *err);
```

**DESCRIPTION**

These functions return the offsets associated with a DWARF attribute descriptor.

Function **dwarf\_formref()** returns the compilation unit relative offset of the descriptor referenced by argument *attr* in the location pointed to by argument *retoffset*. Argument *attr* must be a reference that is local to a compilation unit. Permitted forms for argument *attr* are DW\_FORM\_ref1, DW\_FORM\_ref2, DW\_FORM\_ref4, DW\_FORM\_ref8 and DW\_FORM\_ref\_udata.

Function **dwarf\_global\_formref()** returns the section-relative offset of the descriptor referenced by argument *attr* in the location pointed to by argument *retoffset*. Argument *attr* should be a legal **REFERENCE** class form. Permitted forms for argument *attr* are: DW\_FORM\_ref\_addr, DW\_FORM\_ref\_udata, DW\_FORM\_ref1, DW\_FORM\_ref2, DW\_FORM\_ref4, DW\_FORM\_ref8 and DW\_FORM\_sec\_offset. The returned offset is relative to the start of the ".debug\_info" ELF section.

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

**RETURN VALUES**

These functions return DW\_DLV\_OK on success. In case of an error, these functions return DW\_DLV\_ERROR and sets argument *err*.

**ERRORS**

These functions may fail with the following errors:

[DW\_DLE\_ARGUMENT]            Either of arguments *attr* or *retoffset* was NULL.

[DW\_DLE\_ATTR\_FORM\_BAD] The attribute referenced by argument *attr* was not of a permitted kind.

**SEE ALSO**

dwarf(3), dwarf\_attr(3), dwarf\_formblock(3), dwarf\_formflag(3), dwarf\_formsdata(3), dwarf\_formsig8(3), dwarf\_formstring(3), dwarf\_formudata(3), dwarf\_hasattr(3)