

NAME

dwarf_add_AT_dataref - create an attribute descriptor for a relocatable address

LIBRARY

DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

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

Dwarf_P_Attribute

```
dwarf_add_AT_dataref(Dwarf_P_Debug dbg, Dwarf_P_Die die, Dwarf_Half attr,  
    Dwarf_Unsigned pc_value, Dwarf_Unsigned sym_index, Dwarf_Error *err);
```

DESCRIPTION

Function **dwarf_add_AT_dataref**() creates an attribute descriptor for a relocatable address and attaches it to the debugging information entry referenced by argument *die*.

If flag DW_DLC_SIZE_64 is set, the address value will be 8 bytes in size and of the DWARF form DW_FORM_data8. Otherwise, the address value will be 4 bytes in size and of the DWARF form DW_FORM_data4.

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

Argument *die* should reference a debugging information entry allocated using `dwarf_new_die(3)`.

Argument *attr* specifies the attribute code of the created attribute descriptor.

Argument *pc_value* specifies the value of the relocatable address.

Argument *sym_index* specifies the ELF symbol index of the symbol to be used for relocation.

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_add_AT_dataref**() returns the created attribute descriptor. In case of an error, function **dwarf_add_AT_dataref**() returns DW_DLV_BADADDR and sets the argument *err*.

ERRORS

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

[DW_DLE_ARGUMENT] Either of the arguments *dbg* or *die* was NULL.

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

SEE ALSO

dwarf(3), dwarf_add_AT_ref_address(3), dwarf_add_AT_reference(3),
dwarf_add_AT_signed_const(3), dwarf_add_AT_unsigned_const(3), dwarf_new_die(3),
dwarf_producer_init(3), dwarf_producer_init_b(3)