

**NAME**

**dwarf\_add\_AT\_string** - create and attach a string class attribute

**LIBRARY**

DWARF Access Library (libdwarf, -ldwarf)

**SYNOPSIS**

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

*Dwarf\_P\_Attribute*

```
dwarf_add_AT_string(Dwarf_P_Debug dbg, Dwarf_P_Die die, Dwarf_Half attr, char *str,  
                    Dwarf_Error *err);
```

**DESCRIPTION**

Function **dwarf\_add\_AT\_string()** creates an attribute descriptor belonging to the 'string' class and attaches it to the debugging information entry referenced by argument *die*. The created attribute descriptor will have DWARF form DW\_FORM\_strp.

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* should specify the attribute code for the created attribute descriptor.

Argument *str* should hold a pointer to a NUL-terminated string which will become the value of the created attribute descriptor.

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\_string()** returns the created attribute descriptor. In case of an error, function **dwarf\_add\_AT\_string()** returns DW\_DLV\_BADADDR and sets the argument *err*.

**ERRORS**

Function **dwarf\_add\_AT\_string()** can fail with:

[DW\_DLE\_ARGUMENT] One of the arguments *dbg*, *die* or *str* 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\_const\_value\_string(3), dwarf\_add\_AT\_name(3), dwarf\_new\_die(3),  
dwarf\_producer\_init(3), dwarf\_producer\_init\_b(3)