NAME

    dwarf_get_section_bytes - retrieve ELF section byte streams

LIBRARY

    DWARF Access Library (libdwarf, -ldwarf)

SYNOPSIS

    #include <libdwarf.h>

    Dwarf_Ptr
    dwarf_get_section_bytes(*Dwarf_P_Debug dbg*, *Dwarf_Signed dwarf_section*,
      *Dwarf_Signed *elf_section_index*, *Dwarf_Unsigned *length*, *Dwarf_Error *err*);

DESCRIPTION

    Function **dwarf_get_section_bytes**() returns the ELF section byte streams generated by a prior call to
    function dwarf_transform_to_disk_form(3).

    Each call to function **dwarf_get_section_bytes**() will return the byte stream for one ELF section. The
    first call to this function will always return the first ELF section, and the subsequent calls will return the
    rest of sections in the order when they were generated, until the last one. The total number of sections
    generated is returned by the function dwarf_transform_to_disk_form(3).

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

    Argument *dwarf_section* is currently ignored.

    Argument *elf_section_index* should point to a location which will be set to the section index value of the
    returned ELF section.

    Argument *length* should point to a location which will hold the length in bytes of the returned ELF
    section.

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

  Memory Management

    The memory areas used for the returned ELF section byte streams should be freed using the function
    dwarf_producer_finish().

RETURN VALUES

On success, function **dwarf_get_section_bytes**() returns a pointer to a ELF section byte stream. In case of an error, function **dwarf_get_section_bytes**() will return NULL and set the argument *err*.

EXAMPLES

To generate and retrieve ELF section byte streams, use:

```
Dwarf_P_Debug dbg;
Dwarf_Signed count, i, sec_index;
Dwarf_Unsigned len;
Dwarf_Ptr bytes;
Dwarf_Error de;

/* ... Assume that 'dbg' refers to a DWARF producer instance,
 * and that application code has added DWARF debugging
 * information to the producer instance. ...
 */
if ((count = dwarf_transform_to_disk_form(dbg, &de)) ==
    DW_DLV_NOCOUNT) {
        warnx("dwarf_transform_to_disk_form failed: %s",
            dwarf_errmsg(-1));
        return;
}

/* Retrieve section data. */
for (i = 0; i < count; i++) {
        bytes = dwarf_get_section_bytes(dbg, i, &sec_index, &len,
            &de);
        if (bytes == NULL) {
                warnx("dwarf_get_section_bytes failed: %s",
                    dwarf_errmsg(-1));
                continue;
        }
        /* ... use the returned byte stream ... */
}

/* Release resources. */
dwarf_producer_finish(dbg, &de);
```

ERRORS

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

[DW_DLE_ARGUMENT]   One of the arguments *dbg*, *elf_section_index*, or *length* was NULL.

[DW_DLE_NO_ENTRY]   There were no more ELF sections to retrieve, or the function was called before a call to dwarf_transform_to_disk_form(3).

SEE ALSO

dwarf(3), dwarf_producer_finish(3), dwarf_producer_init(3), dwarf_producer_init_b(3), dwarf_reset_section_bytes(3), dwarf_transform_to_disk_form(3)