NAME

elf32_xlatetof, elf32_xlatetom – class-dependent data translation

SYNOPSIS

cc [*flag* ...] *file* ... **–lelf** [*library* ...]

#include <libelf.h>

Elf_Data *elf32_xlatetof(Elf_Data **dst*, **const Elf_Data** **src*, **unsigned** *encode***);**

Elf_Data *elf32_xlatetom(Elf_Data **dst*, **const Elf_Data** **src*, **unsigned** *encode***);**

MT-LEVEL

Unsafe

DESCRIPTION

elf32_xlatetom( ) translates various data structures from their 32-bit class file representations to their memory representations; **elf32_xlatetof( )** provides the inverse. This conversion is particularly important for cross development environments. *src* is a pointer to the source buffer that holds the original data; *dst* is a pointer to a destination buffer that will hold the translated copy. *encode* gives the byte encoding in which the file objects are to be represented and must have one of the encoding values defined for the ELF header's **e_ident[EI_DATA]** entry (see **elf_getident**(3E)). If the data can be translated, the functions return *dst*. Otherwise, they return null because an error occurred, such as incompatible types, destination buffer overflow, etc.

elf_getdata(3E) describes the **Elf_Data** descriptor, which the translation routines use as follows.

d_buf Both the source and destination must have valid buffer pointers.

d_type This member's value specifies the type of the data to which **d_buf** points and the type of data to be created in the destination. The program supplies a **d_type** value in the source; the library sets the destination's **d_type** to the same value. These values are summarized below.

d_size This member holds the total size, in bytes, of the memory occupied by the source data and the size allocated for the destination data. If the destination buffer is not large enough, the routines do not change its original contents. The translation routines reset the destination's **d_size** member to the actual size required, after the translation occurs. The source and destination sizes may differ.

d_version This member holds version number of the objects (desired) in the buffer. The source and destination versions are independent.

Translation routines allow the source and destination buffers to coincide. That is, **dst→d_buf** may equal **src→d_buf**. Other cases where the source and destination buffers overlap give undefined behavior.

Elf_Type	32-Bit Memory Type
ELF_T_ADDR	Elf32_Addr
ELF_T_BYTE	unsigned char
ELF_T_DYN	Elf32_Dyn
ELF_T_EHDR	Elf32_Ehdr
ELF_T_HALF	Elf32_Half
ELT_T_OFF	Elf32_Off
ELF_T_PHDR	Elf32_Phdr
ELF_T_REL	Elf32_Rel
ELF_T_RELA	Elf32_Rela
ELF_T_SHDR	Elf32_Shdr
ELF_T_SWORD	Elf32_Sword
ELF_T_SYM	Elf32_Sym
ELF_T_WORD	Elf32_Word

Translating buffers of type **ELF_T_BYTE** does not change the byte order.

SEE ALSO

elf(3E), **elf32_fsize**(3E), **elf_getdata**(3E), **elf_getident**(3E)