The Program Header Table

Program headers are only important in executable and shared object files. The program header table is an array of entries where each entry is a structure describing a segment in the object file or other information needed to create an executable process image. The size of an entry in the table and the number of entries in the table are given in the ELF header (See Figure 2.6). Each entry in the program header table (see Figure 2.7) contains the type, file offset, physical address, virtual address, file size, memory image size, and alignment for a segment in the program. The program header is crucial to creating a process image for the object file. The operating system copies the segment (if it is loadable, i.e., if `p_type` is PT_LOAD) into memory according to the location and size information. The `p_type` field is shown in Figure 2.7 as the first item in the struct.

Figure: The Section Header

```
typedef struct -
    Elf32'Word        sh'name;          //  name of section
    Elf32'Word        sh'type;          //  type of the section
    Elf32'Word        sh'flags;         //  section-specific attributes
    Elf32'Addr        sh'addr;          //  memory location of sectio
    Elf32'Off         sh'offset;        //  file offset to section
    Elf32'Word        sh'size;          //  size of section
    Elf32'Word        sh'link;          //  section type dependent
    Elf32'Word        sh'info;          //  extra information
    Elf32'Word        sh'addralign;     //  address alignment constrai
    Elf32'Word        sh'entsize;       //  size of an entry in section
~ Elf32'Shdr;
```