For example, the last example has this declaration:

```c
int * p_updates;
```

This states that the combination `* p_updates` is type `int`. Because the `*` operator is used by applying it to a pointer, the `p_updates` variable itself must be a pointer. We say that `p_updates` points to type `int`. We also say that the type for `p_updates` is pointer-to-int or, more concisely, `int *`. To repeat: `p_updates` is a pointer (an address), and `*p_updates` is an `int` and not a pointer. (See Figure 4.9.)

**Figure 4.9. Pointers store addresses.**

Incidentally, the use of spaces around the `*` operator are optional. Traditionally, C programmers have used this form:

```c
int *ptr;
```