

Exemplo e discussão de como definir uma variável global utilizando vários arquivos como no CodeWarrior (main.c e event.c). Esta explicação foi retirada desta [resposta](#) do stack overflow site em 06/07/2020. Mais exemplos e explicações veja o mesmo link.

Using `extern` is only of relevance when the program you're building consists of multiple source files linked together, where some of the variables defined, for example, in source file `file1.c` need to be referenced in other source files, such as `file2.c`.

It is important to [understand the difference between *defining* a variable and *declaring* a variable](#):

- A variable is **declared** when the compiler is informed that a variable exists (and this is its type); it does not allocate the storage for the variable at that point.
- A variable is **defined** when the compiler allocates the storage for the variable.

You may declare a variable multiple times (though once is sufficient); you may only define it once within a given scope. A variable definition is also a declaration, but not all variable declarations are definitions.

Best way to declare and define global variables

The clean, reliable way to declare and define global variables is to use a header file to contain an `extern` *declaration* of the variable.

The header is included by the one source file that defines the variable and by all the source files that reference the variable. For each program, one source file (and only one source file) defines the variable. Similarly, one header file (and only one header file) should declare the variable. The header file is crucial; it enables cross-checking between independent TUs (translation units — think source files) and ensures consistency.

Although there are other ways of doing it, this method is simple and reliable. It is demonstrated by `file3.h`, `file1.c` and `file2.c`:

file3.h

```
extern int global_variable; /* Declaration of the variable */
```

file1.c

```
#include "file3.h" /* Declaration made available here */
#include "prog1.h" /* Function declarations */

/* Variable defined here */
int global_variable = 37; /* Definition checked against declaration */

int increment(void) { return global_variable++; }
```

file2.c

```
#include "file3.h"
#include "prog1.h"
#include <stdio.h>

void use_it(void)
{
    printf("Global variable: %d\n", global_variable++);
}
```

That's the best way to declare and define global variables.

