

## GammaLib - Change request #1053

### Remove superfluous return statements

01/04/2014 01:10 PM - Deil Christoph

<b>Status:</b>	Rejected	<b>Start date:</b>	01/04/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>	Deil Christoph	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

#### Description

Jürgen, I'd like to re-propose a change to the Gammalib Coding and design conventions (<http://gammalib.sourceforge.net/coding/index.html>) that I proposed a year ago in #558 and that got lost in discussion on other things without being accepted or clearly rejected.

I propose to change ""Each function and/or method terminates with a return statement." to ""Each function and / or method that returns something terminates with a return statement. Functions that do not return anything do not need an empty return statement at the end because returning to the caller is the default behaviour of any C / C++ / Python function / method."

Here's our discussion on this question from #558:

Christoph: Why require a return statement in functions / methods returning void? This seems completely superfluous (i.e. a waste of screen real estate) to me.

Jürgen: The C++ documentation says "The return statement stops execution and returns to the calling function.", so this has nothing to do with having or having not something to return. I agree it's optional for void functions, but personally I preferred to always have a return statement for clarity. Probably I decided to use this rule also for compatibility assurance, but this is maybe not an issue.

I don't see how the empty return statement at the end adds clarity and I doubt there's users with decade-old compilers that got such a fundamental C / C++ / Python behaviour wrong.

My argument for removing it is the same as before: Removing these superfluous return statements will save 1000s of lines, making more space on my screen for useful code.

Jürgen, I'm happy to make a pull request if you agree.

#### History

##### #1 - 01/05/2014 11:53 PM - Knödseder Jürgen

It's of course a question of style, but I also consider this as a safe guard for programmers: a function should always have a return statement.

Consider the following code:

```
#include<iostream>
int function(void) {
    int i = 0;
}
int main(void) {
    int k = function();
    std::cout << k << std::endl;
}
```

The code is obviously wrong, yet it compiles without any warning. Running the program gives on my machine

, i.e. an arbitrary value.

So if you ask a programmer to **always** add a return statement, he would probably not forget the ones that are needed :-). And it also shows the point where the function is **intended** to end.

## #2 - 01/06/2014 12:02 AM - Deil Christoph

But the first thing everyone learns is to compile with `-Wall`:

```
$ g++ test.cpp
$ g++-mp-4.8 -Wall test.cpp
test.cpp: In function 'int function()':
test.cpp:3:9: warning: unused variable 'i' [-Wunused-variable]
    int i = 0;
        ^
test.cpp:4:1: warning: no return statement in function returning non-void [-Wreturn-type]
}
^
```

Excellent compilers warn about this by default:

```
$ clang++ test.cpp
test.cpp:4:1: warning: control reaches end of non-void function [-Wreturn-type]
}
^
1 warning generated.
```

:-)

GammaLib doesn't have this problem and if anyone ever writes something like this CI will see it in a build with `-Wall -Werror`.

**#3 - 01/07/2014 03:10 PM - Knödseder Jürgen**

*- Status changed from New to Rejected*