GammaLib - Change request #1055

Review coding rule: "Output arguments should be passed as pointers"

01/04/2014 02:22 PM - Deil Christoph

Status:	New	**Start date:**	01/04/2014
Priority:	Low	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

The GammaLib Coding and design document contains this coding rule:

> Output arguments should be passed as pointers.

This is a common coding rule that can e.g. also be found here:
http://google-styleguide.googlecode.com/svn/trunk/cppguide.xml?showone=Reference_Arguments#Reference_Arguments

I noticed that GModel::write(GXmlElement& xml) and dozens of other GXXX::write() methods violate this rule, because they passes an output argument by reference instead of as a pointer:
http://gammalib.sourceforge.net/doxygen/classGModel.html#ab2692db0659a2b39a3a185fe62e72e27

Does this "only" need to be cleaned up or is there a reason for this (e.g. the Python interface)?

I did not look around to see whether there are other cases where output arguments are passed by non-const references instead of by pointer.

History

#1 - 01/06/2014 12:01 AM - Knödlseder Jürgen

The examples you mention are those where the arguments are input and output. For example, the write methods typically appends something to an existing object, but they do not create the object from scratch. When I was writing this rule I was more thinking about "pure" output arguments, hence arguments where the method fully sets the object.