

GammaLib - Action #2340

Prevent de-allocation of client instances

02/15/2018 11:52 AM - Knödseder Jürgen

Status:	Closed	Start date:	02/15/2018
Priority:	Normal	Due date:	
Assigned To:	Knödseder Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.5.1		

Description

If the following code is called with a reference to a column that actually exists already in the table, the column will be destroyed, leading specifically to a

libc++abi.dylib: Pure virtual function called!

exception in Python (this is specifically a Python problem which handles pointers to objects):

```
GFitsTableCol* GFitsTable::set(const int& colnum, const GFitsTableCol& column)
{
    // Free existing column
    if (m_columns[colnum] != NULL) delete m_columns[colnum];

    // Clone column
    m_columns[colnum] = column.clone();

    // Return pointer to column
    return m_columns[colnum];
}
```

This can be prevented by the following code

```
GFitsTableCol* GFitsTable::set(const int& colnum, const GFitsTableCol& column)
{
    // Free existing column only if it differs from current column. This
    // prevents deallocating the instance that is passed as argument which
    // would be
    if ((m_columns[colnum] != NULL) && (m_columns[colnum] != &column)) {
        delete m_columns[colnum];
    }

    // Clone column
    m_columns[colnum] = column.clone();

    // Return pointer to column
    return m_columns[colnum];
}
```

All GammaLib methods should be scanned for this potential problem and corrected.

History

#1 - 02/15/2018 06:00 PM - Knödseder Jürgen

- Status changed from New to Closed

- % Done changed from 0 to 100

The GammaLib code was scanned for comparable issues and corrected similar to the GFitsTable::set method. The code was merged into devel and bugfix-1.5.1.