

GammaLib - Action #1231

Feature # 591 (In Progress): Investigate how GammaLib can be made VO compliant.

Design and implement GVOTable class

06/19/2014 02:49 PM - Knödseder Jürgen

Status:	In Progress	Start date:	06/19/2014
Priority:	Normal	Due date:	02/27/2015
Assigned To:		% Done:	80%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
<p>To exchange FITS images or FITS tables using VO protocols, the VOTable format needs to be implemented. For this purpose, a GVOTable class should be implemented that allow provision for conversion between FITS and VOTables. For example</p> <pre>GFitsTable fitstable; ... GVOTable votable(fitstable);</pre> <p>should convert a FITS table into a VO table upon construction. Vice versa</p> <pre>GFitsTable* fitstable = votable.fitstable();</pre> <p>should return a pointer to a FITS table based on the information found in a VOTable. Similar methods should exist for FITS images.</p> <p>The GVOTable class may make use of the GXml class for XML formatting. The class then needs to implement methods for reading and writing, best using the GUrl media independent class:</p> <pre>GUrlFile url("my_file.votable"); votable.read(url); ... votable.write(url);</pre>			

History

#1 - 12/11/2014 10:49 PM - Knödseder Jürgen

- Due date set to 02/27/2015

#2 - 09/21/2015 08:40 PM - Knödseder Jürgen

- Status changed from New to In Progress

- % Done changed from 0 to 10

The GVOTable class should satisfy the following use case (here in Python):

```
import gammalib  
skymap = gammalib.GSkyMap("my_map.fits")  
votable = gammalib.GVOTable(skymap)  
client = gammalib.GVOClient()  
client.send(votable)
```

The GVOTable class should take a sky map object as constructor argument and translate the sky map information into a VO table. Two cases need to be distinguished: a sky map in a WCS projection, an a HealPix skymap. Implementing of the WCS projection has priority as most sky maps are in that format. The GVOTable class should only provide the conversion capabilities, but no capabilities to send the object to the hub. The GVOTable class should have a GXml member that will contain the VO table.

The publishing of the VO table is accomplished by a GVOClient::send method. Eventually, this method could also be named GVOClient::publish which may be more appropriate for the action that is taken. This needs to be decided before implementing the method.

#3 - 09/25/2015 02:03 PM - Louge Thierry

- % Done changed from 10 to 30

GVOTable is now an accessible object in python whitout going through GSkymap.

Standard for GVOTable (<http://www.ivoa.net/documents/REC/DM/STC-20071030.html>) under implementation.

GVOTable needs access to skymap elements like number of pixels in each axis, which can be found in GSkymap. If those informations are not available in GWs, it will be necessary to add an access method to GSkymap to get them.

test that can be used to see the current VOTable:

```
skymap = gammalib.GSkymap("../gammalib/test/data/cena_lobes_parkes.fits")
votable = gammalib.GVOTable(skymap)
makes VOTableExemple.xml on the current directory
```

#4 - 10/06/2016 05:41 PM - Louge Thierry

- % Done changed from 30 to 80

List of corrected bugs during 6th coding sprint:

B1 -> GVOClient bug

Description: GVOClient always launched GVOHub, even when another Hub was running.

Cause: The ping method malfunctioned, the structure of SAMP message was incorrect.

Fix: correction of ping SAMP message structure (empty params required, was not empty)

B2 -> GVOClient bug

Description: When using another HUB than GVOHub, images wont be displayed in external applications.

Cause: bad structure of message image.load.fits

This bad structure is OK for the GVOHub but not functioning with other hubs.

Fix: removed a parasite string in the message structure for table.load.votable

Replaced the hub secret key in the structure values with the client private key value

B3 -> GVOClient bug

Description, fix: same as B2 but concerning table.load.fits

B4 -> GVOHub malfunction

Description: After B2 and B3 solved, GVOHub didn't recognized the concerned messages types anymore.

Fix: Modification of the xml parsing to get the relevant mtypes from the corrected SAMP structure.

Oveview:

The GVOHub messages to the clients were OK.

The messages sent by the clients to the GVOHub were not always well understood by GVOHub (image.load.fits, table.load.votable not OK).

The GVOClient always started a GVOHub even if a hub was already running.

GVOClient didn't sent the correct SAMP messages structures for non-GVOHub (-> GVOHub was not fully standard)

List of new functionalities:

GVOTable can now load an xml file and be shared through SAMP

GVOClient can share VoTables and Fits with VO applications using either GVOHub or any SAMP hub running (result of bug fixing)

```
Testing:
import gammalib
client = gammalib.GVOClient()
client.connect()
fits = gammalib.GFits("<Any available FITS File>")
hdu = fits.at(<HDU considered>)
client.publish(hdu)
votable = gammalib.GVOTable("<Any available VOTable File>")
client.publish(votable)
```

To be done:

We are not sure of GVOTable usefulness.

SAMP doesn't allow to share a stream, but needs a copy of a file no matter the format used (VOTable, FITS...).

If any data can be expressed in FITS and correctly interpreted by VO applications, then GVOTable is of no use.

If cases happen where data can't be put into FITS format, or not understood into FITS format by other apps, then GVOTable may be useful.

For now, GVOTable is usable but limited: it reads an xml file containing a VOTable data and that's it.

Functions already exist in GVOTable to construct a GVOTable xml format from data inside gammalib, but putting it all together should be considered only if GVOTable may be of any use.

#5 - 10/06/2016 11:13 PM - Knödseder Jürgen

- Target version set to 1.2.0

I integrated the code into devel.

I reworked a bit the GVOTable class so that one can now generate a VO table from a FITS table. I checked that the code works by loading an event list, translating the event list into a VO table, and sending the VO table via the hub to Topcat (either via the Topcat Hub or the GammaLib Hub).

I also checked that publishing an image to Aladin still works. Publishing a FITS table to Aladin works also, Aladin seems to convert automatically that table into an image. Even publishing a VO table to Aladin works, also here Aladin converts the VO table into an image.

I think there is a definite need for the GVOTable class, namely for the transmission of tables to Topcat. Trying to send a FITS table to Topcat does not work, but this seems normal since the GVOClient::publish() method sends the FITS table using the image.load.fits command. If one tries this, Topcat seems simply to ignore the notification. Hence to display VO tables in Topcat the GVOTable class is necessary.

#6 - 10/06/2016 11:16 PM - Knödseder Jürgen

Note that I keep the issue open since the transformation of a VO table to a FITS table still needs to be implemented. This implies walking through a filled XML file and filling a table structure.

Also note that FITS vector columns are not yet supported by the class.

#7 - 03/03/2017 10:20 AM - Knödseder Jürgen

- Target version changed from 1.2.0 to 1.3.0

#8 - 06/06/2017 10:26 PM - Knödseder Jürgen

- Target version changed from 1.3.0 to 1.4.0

#9 - 07/31/2017 11:09 PM - Knödseder Jürgen

- Assigned To deleted (Louge Thierry)

- Target version deleted (1.4.0)