Mem. S.A.It. Vol. 80, 441

Memorie della

VisIVO: data exploration of complex data

G. Caniglia[1,4], U. Becciani[1], M. Comparato[1], A. Costa[1], C. Gheller[2], A. Grillo[1], M. Krokos[3], and F. Vitello[4]

[1] Istituto Nazionale di Astrofisica – Osservatorio Astronomico di Catania, Italy
e-mail: gabriella.caniglia@oact.inaf.it
[2] CINECA, Casalecchio di Reno, Italy
[3] School of Creative Technologies, Univ. of Portsmouth, UK
[4] Consorzio Cometa, Consorzio Multiente per la promozione e l'adozione di teconologie di calcolo avanzato, Catania, Italy

Abstract. Advanced visualization tools can help the researcher in investigating and extracting information from complex data. VisIVO and VisIVOServer, are novel open-source graphics applications that use high-performance multidimensional visualization techniques for astrophysical data exploration and analysis. They support the standards defined by the International Virtual Observatory Alliance in order to make them interoperable with VO (Virtual Observatory) data repositories. This paper is an outline of the current functionality supported by VisIVO and VisIVOServer.

Key words. Techniques: image processing - Methods: data analysis - Advanced Visualization - Complex Astrophysics Datasets - Data Exploration - High-Performance Graphics

1. Introduction

An essential part of astrophysical research is the necessity to employ graphics and visualization tools for appropriately displaying images and data plots either from real-world observations or from highly-complex advanced numerical simulations. The latest generation of graphical software gives the astrophysical community powerful instruments for data exploration by using:

1. High performance and multithreading in order to exploit multicore systems, powerful graphic cards and coprocessors. In this way the user can interact with large scale datasets in real time.
2. Interoperability so as to allow for different applications, each specialized for different purpose, to share the same data set.
3. Collaborative workflows so that several users can work at the same time, perhaps at different geographical locations, with the same data exchanging information and experiences.

Tools such as VisIVO, Aladin Bonnarel et al. (2000) and TOPCAT[1] have been developed recently in the framework of the Virtual Observatory (VO), incorporating some of the aforementioned technologies.

Send offprint requests to: G. Caniglia

[1] http://www.star.bris.ac.uk/~mbt/topcat/

2. VisIVO

2.1. Overview

VisIVO[2] is a C++ application specifically designed for exploration of multidimensional data. It is open-source software available for a variety of platforms (MS Windows, GNU/Linux and MacOS). VisIVO is wrapped around the Multimod Application Framework (MAF, Viceconti et al. 2004). MAF is an open source software framework for rapid development of data visualization and analysis applications. It provides high-level components that can be easily combined to develop vertical applications. This framework was developed by the visualization group of CINECA[3] and is based on the visualization toolkit (VTK, Schroeder et al. 2004) library, for the multidimensional visualization and on the wxWidget library, a portable open-source GUI (graphical user interface) library. MAF also incorporates other open libraries or drivers, e.g., for interaction with virtual reality devices (3D mice, gloves, haptics devices, etc.). The latest version of VisIVO supports the most important and popular astronomical data formats. Data can be retrieved directly by connecting to an available VO service (e.g., VizieR, Ochsenbein et al. 2000) then loading into the local computer memory where regions of interest can be identified, manipulated interactively, and visualized. VisIVO can deal with both observational and simulated data, especially focusing on multidimensional datasets (e.g., catalogs, computational meshes, etc.), and can visualize data represented either as points or volumes. Furthermore, it incorporates advanced visualization functionality such as computation of isosurfaces, or even creation and visualization of vectors by combining a selection of scalar values. All derived data can be further saved not only as ascii or binary files but also as VOTables.

[2] http://visivo.oact.inaf.it
[3] http://openmafcineca.it

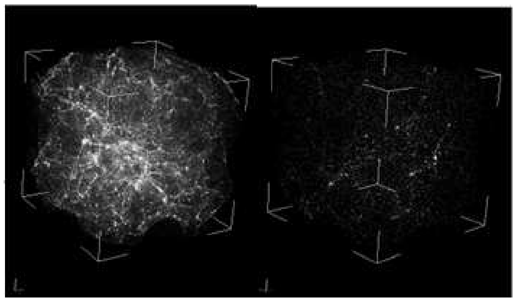

Fig. 1. LOD (level of detail). The left image shows a customized user visualization; the right image also employs on Lod representation.

2.2. Operational scenario

The first operation is loading the data, the supported data types being VOTables, FITS, HDF5, ascii, binary raw data, and the native data format of the popular Gadget simulation code Springel (2005). The results of an operation - such as loading, importing or perhaps a modifying action - are presented to the user as tree-like structure nodes. To display the data associated with particular nodes it is necessary to instantiate views, that is rendering windows for 3D visualization supporting not only standard pan/zoom capability but also advanced navigation functionality (see Figure 1). VisIVO allows the following elements in its views: Points, Vectors, Volume Rendering, Isosurfaces, Glyphs (that is geometrical forms associated with each data point), Stereo Rendering and Histograms. Each view displays data according to a specialized visualization pipeline. It is possible to instantiate several views, which can display different subsets of data. This is a strength of VisIVO as many specialized views, displaying different (or the same) data, can be instantiated, while data can be manipulated and analyzed independently from displayed views. A set of operations are available in order to analyze and modify data loaded in the tree. VisIVO has operations that simply modify the data, operations that perform a statistical analysis on data and operations that do both. All of these generate output nodes that can be displayed according to the output type (e.g. an output that represents statistical 2D data can be displayed in a histogram view). VisIVO is designed to simultaneously handle as many properties as possible, e.g., complex tables can be loaded and manipulated, new fields can be derived and represented graphically, using points, colours, transparency, surfaces, glyphs and volume rendering.

VisIVO can deal with both structured and unstructured data. It does not associate any geometry to the data, this is up to the user who by applying specific operations, can create geometries using one or many of the loaded fields. After the user creates a point distribution using his/her data, points, besides their geometric position, can be used to display further quantities, using colours and glyphs (these are 3D shapes, like spheres or cubes). Points can be coloured as a function of a given scalar field (e.g., their temperature or their spectral index) with a colourmap that is fully user customizable. Each point can also have an associated glyph, whose size can be a function of one (for spheres) or two (for cubes, cylinders, pyramids) fields. A vector quantity can be visualized as well, using either oriented segments or arrows. Vectors can also be coloured according to their magnitude. Structured mesh-based data can be visualized using volume rendering and isosurfacing. VisIVO also provides various built-in utilities that allow the user to perform mathematical operations and to analyze their data. It is possible to apply algebraic and mathematical operators to the loaded data. Basic arithmetic operations (addition, subtraction, multiplication, division) as well as logarithm, power, absolute value and many others are supported. Scalar product, magnitude and norm of vector quantities are available too. In this way, new physical quantities can be calculated for subsequent analysis and visualization.

3. VisIVOServer

VisIVOServer[4] is a spin-off of VisIVO. VisIVO Server can create a 3D view from data tables. It is open-source software available for

[4] http://visivoserver.oact.inaf.it

```
float
5
262144 64 64 64 1 1 1
little
X
Y
Z
scalar0
scalar1
```

Fig. 2. This is an example of a header of the binary internal table.

GNU/Linux. The defining characteristic of VisIVOServer compared with VisIVO is the support for very large-scale datasets. No limits (in principle) are prescribed for data visualization and the forthcoming version will also provide support for FITS and HDF5 among other popular formats. The VisIVOServer application is composed of three distinct software modules as follows:

1. VisIVOImporter;
2. VisIVOFilters;
3. VisIVOViewer.

VisiVOImporter converts input data into an internal binary data format to be used by VisIVO Filters and the VisIVO Viewer. The following data formats can be currently converted: ascii, csv, binary, rawgrids, raw points, fly output, fits VOtable and gadget. The internal binary file is made up from a header together with the actual data values as follows:

1. The header file (e.g. filename.bin.head, see Figure 2) typically contains metadata with the following description:
   – data format;
   – number of fields;
   – number of points for each field. in the same row if the binary file is a volume have the number of cells in each dimension of the mesh size and represent the real unit of each cell;

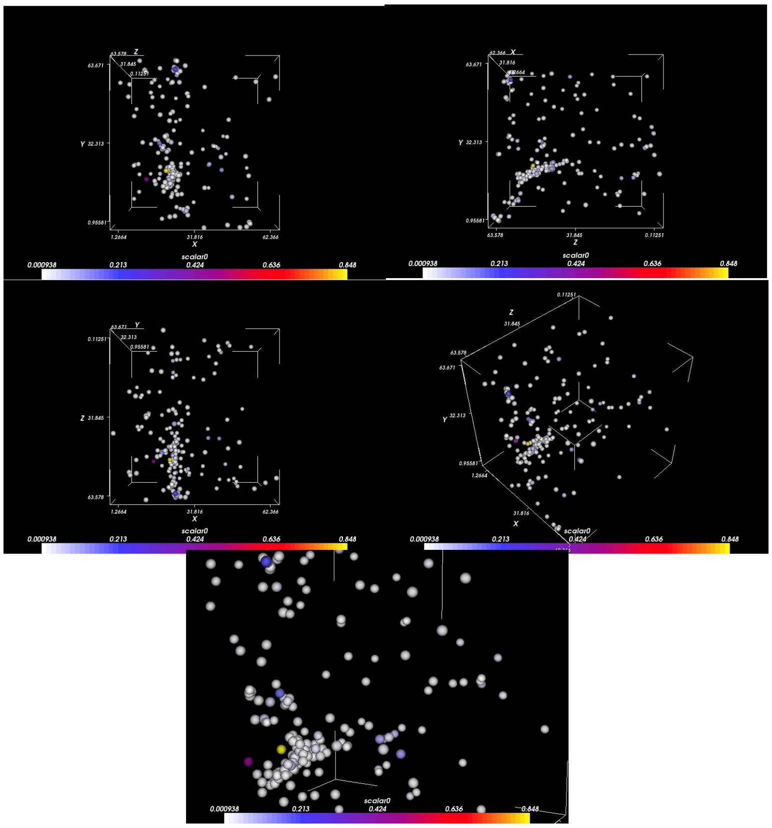

Fig. 3. This is an example of the VisIVOServer output. The first four images are the default-produced images, the last one is the point of view that the user has selected. The user has also made use of a lookuptable and sphere glyphs.

   – endianism;
   – a sequence of the field names.
2. The data values part (e.g.filename.bin) is just a sequence of values (e.g., all X values followed by all Y values).

VisIVOFilters typically convert data from an original data table (internal data format table) into a new one. Currently the following are available:

– Randomizer: Create a random subset from the original data table.
– Math Operations: Create a new field in a data table as the result of a mathematical operation between existing fields.
– Merge Tables: Create a new table from two or more existing data tables.
– Select Rows: Create e new table using limits on one or more fields of a data table.
– Select Columns: Create a new table using on one or more fields of a data table.
– Append Tables: Create a new table appending data of two existing tables.
– Extraction Tables: Create a sphere or cube sub-sample of the original file.
– Scalar distribution: Create internal binary table that have all the values to create an histogram.
– Visual image: Create a table with all the field to load in VisIVOViewer.

VisIVOViewer create views from the input data file. The input data file must fit within the available RAM. The current version of VisIVOViwer displays points in a box considering a three dimension coordinate system that the user can specify or that the program can select automatically. There is also the possibility to use lookuptable glyphs (cubes, spheres, cones or pixels) and scale this by height and/or radius. VisIVOViewer by default generates four image files with a fixed camera position and zooming factor. A fifth image is also produced based on user-defined line command options (see Figure 3).

4. Conclusions

The two application described in this paper, VisIVO and VisIVOServer, are useful for data analysis and visualization, especially when working with large-scale, complex astrophysical dataset. They can be closely integrated, but are also complementary and independent of each other. The user can use for example VisIVOServer to create a preview of his/her data or an appropriately-defined subsample, then employ VisIVO to visualize and interact with such a subsample. Currently, a novel software architecture is under development for VisIVO and VisIVOServer, allowing easier incorporation of enhanced functionality for future developments.

References

Bonnarel F., et al. 2000, A&AS, 143, 33

Comparato, M.,U., Becciani, Costa, A., Larsson, B., Garilli, B., Gheller, C., & Taylor, J. 2007 ASP, 119, 898-913

Ochsenbein, F., Bauer,P., & Marcout, J. 2000 A&AS, 143, 23

Schroeder, W.,Martin, K, & Lorensen, B. 2004, The Visualization Toolkit:An Object-Oriented Approach to 3D Graphics (3rd ed.; Clifton Park:Kitware)

Springel, 2005, User guide for Gadeget-2

Viceconti M., et al. 2004, in IEEE Proc. Eighth International Conference of Information Visualisation (Washington:IEEE), 15, 143, 33