Visualization, Exploration, and Data Analysis of Complex Astrophysical Data

M. Comparato, U. Becciani, A. Costa, and B. Larsson
INAF, Osservatorio Astrofisico di Catania, Italy; marco.comparato@oact.inaf.it

B. Garilli
INAF, Istituto di Astrofisica Spaziale e Fisica Cosmica, Italy

C. Gheller
CINECA, Casalecchio di Reno, Italy

AND

J. Taylor
Institute for Astronomy, University of Edinburgh, UK

Received 2007 May 4; accepted 2007 July 5; published 2007 September 5

ABSTRACT. In this paper, we show how advanced visualization tools can help the researcher in investigating and extracting information from data. The focus is on VisIVO, a novel open-source graphics application that blends high-performance multidimensional visualization techniques and up-to-date technologies to cooperate with other applications and to access remote, distributed data archives. VisIVO supports the standards defined by the International Virtual Observatory Alliance in order to make it interoperable with VO data repositories. The paper describes the basic technical details and features of the software, and it dedicates a large section to show how VisIVO can be used in several scientific cases.

Online material: color figures

1. INTRODUCTION

The astronomical community has always dedicated special attention to the growth of graphical and visualization tools, driving their evolution or even being directly involved in the development of many of them. At present, the most popular software for astronomers can be subdivided into two main categories: tools for image display and processing and tools for plotting data. Notable among the former are IRAF, by NOAO; ESO-MIDAS, by the European Southern Observatory; SAOImage, by the Smithsonian Astrophysical Observatory; and GAIA, by ESO. Many other tools are available, but we refer the reader to dedicated surveys for a complete list. Gnuplot and SuperMongo are popular applications adopted for 2D data plots. A more sophisticated solution is represented in IDL by ITT Visual Information and is characterized by a large library of functions specifically developed for astrophysics. Again, for a complete list, we refer the reader to specific surveys.

Among the most popular N-body visualization codes used by the community are TIPSY: motivated by the need to quickly display and analyze the results of N-body simulations, it is mainly limited to this type of data; ParaView: produced by Kitware in conjunction with the Advanced Computing Laboratory at Los Alamos National Laboratory (LANL), the goal of the project is to develop scalable parallel processing tools with an emphasis on distributed memory implementations; and IDL, mentioned above, which contains support for N-body data display but is not free software.

A new generation of graphic software tools is now emerging. These tools are designed to overcome the limits and the barriers of traditional software by exploiting the latest technological opportunities. The main challenges and objectives are the following:

1. High performance and multithreading, in order to exploit multicore systems, large memories, and powerful graphic cards and coprocessors. This allows the user to handle large amount of data in real time.

2. Interoperability, allowing different applications, each specialized in doing different things, to interact with each other in a coordinated and effective way according to well-defined protocols. The aim is to provide to the user a complete suite of tools to best analyze his or her data. Huge, monolithic, and often inefficient tools are obsolete.

3. Collaborative work. The tools allow several users to work on the same data at the same time from different places, exchanging experience, information, and expertise.

4. Access to distributed resources, via Web services and/or grid protocols. Often, data can no longer be moved from data centers, as they are too large and complex. The astronomer must have the tools to access them, independently of his geographical location, in a fast and reliable way.
Tools such as VisIVO, Aladin (Bonnarel et al. 2000), and Topcat have been recently developed in the framework of the Virtual Observatory (VO) to achieve all or some of these goals. In this paper, we focus in particular on VisIVO, which stands for Visualization Interface for the Virtual Observatory. VisIVO is being developed as a collaboration between the Italian National Institute for Astrophysics (INAF)—Astrophysical Observatory of Catania and CINECA (the largest Italian academic high-performance computing center) in the framework of the FP6 EU-funded VO-Tech project. The next section gives a short review of the basic functionalities of VisIVO, while § 3 describes PLASTIC, a messaging protocol that allows heterogeneous applications to work together. Parts of §§ 3 and 4 are dedicated to presenting several scientific cases in which the support of graphics and visualization is of primary importance. In these sections we show some of VisIVO’s capabilities in action, demonstrating how they can be effectively used in practical applications.

2. VisIVO

VisIVO is a C++ application specifically designed to deal with multidimensional data. It is free software available both for MS Windows and for GNU/Linux (porting to MacOS is in progress). It can be downloaded from the Web site. The software is built on the top of the Multimod Application Framework (MAF; Viceconti et al. 2004). MAF is an open-source framework for the development of data visualization and analysis applications. It provides high-level components that can be easily combined to develop a vertical application. It is being developed by the visualization group of CINECA and can be downloaded from the Web site. The framework is based on the Visualization ToolKit (VTK; Schroeder et al. 2004) library for the multidimensional visualization and on the wxWidgets library, a portable open-source GUI (graphical user interface) library, for the user interface. It incorporates other open libraries (for example, for data encryption) or drivers for virtual reality devices (3D mouse, gloves, haptics, etc.). VisIVO’s architecture strictly reflects the structure of a typical scientific application built with the MAF, being mainly developed in the highest layers of the framework. The software exploits, wherever possible, the standard visualization services, views, operations, and interface structures provided by the framework and implements all the elements that are specific to the visualization and analysis of astronomical data.

Extensions to the basic MAF infrastructure have been developed in order to match astronomy-specific requirements and to provide the highest performance. Internal data representation is in the form of a “table data” structure, which is composed of a sequence of variables loaded from a data source such as a file or a database. Regardless their original type, variables are all converted to “double” format. While this incurs a penalty in the application’s memory needs, it provides the necessary precision in some of the data processing stages. Once a table is loaded the user can manage and visualize the data. These operations do not increase the memory usage as long as they do not create new tables or new fields: the visualization process is carried out using references to the table data with no data replication. In order to visualize data, the user has to set which of the loaded fields will be used as the coordinate system of a Cartesian reference frame. In this way, the software ensures maximum flexibility in data usage.

2.1. VisIVO for Data Visualization

Data visualization is the main target of VisIVO. The software is designed to simultaneously handle as many properties as possible. Complex tables can be loaded and manipulated; new fields can be derived and finally represented graphically, using points, colors, transparencies, surfaces, glyphs, and volume rendering. The first step of a working session is usually data loading. Data can be read from files; VisIVO supports different kinds of file formats: standard file formats, such as VOTables, FITS, HDF5, ASCII, raw binaries, and the native data format of the popular GADGET simulation code (Springel 2005). The VOTable format is an XML standard for the interchange of astronomical data, defined by the International Virtual Observatory Alliance (IVOA). Data are represented as a set of tables, each table being an unordered set of rows, whose format is specified in the table XML metadata. Row sequences of table cells, each containing either a primitive data type or an array of such primitives. VOTables can also contain links to external files as a separate data source. VisIVO uses the Savot VOTable parser developed by CNS to load and write VOTables. FITS and HDF5 importers are implemented using the published API and libraries. The ASCII table format consists of columns of data spaced with the most common separation characters (space, tab, etc.). Raw files are sequences of variables written as binary dumps of the memory. The binary files can be managed by descriptor files that store the associated information (number of variables, data types, etc.). VisIVO can also interact with CNS VizieR data service (Ochsenbein et al. 2000), retrieving data directly from remote archives (see par. 2.3).

Once data are loaded they can be visualized and analyzed. VisIVO can deal with both structured and unstructured data. The former are represented by fields defined on a regular mesh. The latter are data with no special geometry; they are treated as sets of points. Graphically, unstructured data have a default representation as pixels. The points’ geometrical distribution is set by selecting three of the fields loaded as table data; for example, using the $x$, $y$, $z$ coordinates of the particles from a

---

1 See http://www.starlink.ac.uk/topcat.
2 See http://visivo.cineca.it.
3 See http://openmaf.cineca.it.
4 See http://www.ivoa.net.
5 Centre de Données astronomiques de Strasbourg.
$N$-body simulation, or the right ascension, declination, and $V$ mag (Johnson magnitude $V$) fields for data from stellar catalogs (Fig. 1).

Besides their geometric position, points can be used to display further quantities, using colors and glyphs (3D shapes, like spheres or cubes). Points can be colored as a function of a given scalar field (e.g., their temperature or their spectral index) with a color map that the user can customize. 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 colored according to their magnitude.

If the size of the data set is too large to be managed in memory, VisIVO also allows the user to extract a random subset of points. It is also possible to select the points that lie in a region that the user is specifically interested in (e.g., a galaxy cluster in a cosmological simulation or a globular cluster in a catalog of stars). The selection can be accomplished using either a rectangular sampler or the cluster finder utilities. For the latter, the following cluster identification method is implemented. A field associated to the points (e.g., the point mass) is used to set a threshold. All the points that have a value of the field above the threshold comprise a cluster. Surfaces that divide regions above and below the threshold can be visualized (see Fig. 2). Regions geometrically disjoint (i.e., their threshold surfaces do not overlap) are identified as separate clusters.

Structured mesh-based data do not have a default graphical representation. VisIVO can visualize them using volume rendering and isosurfaces. Volume rendering is a visualization technique in which the field values are represented by different colors and different transparencies. The global effect is a cloud appearance. This method also enables the user to emphasize the inner parts of the volume. Isosurfaces are surfaces of given value calculated from a mesh-based quantity. The isosurface can be defined as the surface that divides regions in which a given field has value above a certain value from regions in which it is below that value.

### 2.2. VisIVO for Data Analysis

VisIVO provides various built-in utilities that allow the user to perform mathematical operations and to analyze 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 law, 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 example, for the gas distribution of a simulated galaxy cluster, the X-ray emission due to thermal bremsstrahlung can be calculated as proportional to the product of the square of the mass density and the square root of the temperature of the gas. If these two quantities are available, the emission can be immediately derived. It is also possible to merge two different table data structures to create a new one. Data in the resulting table can be treated as a single data set. The merging capabilities and the mathematical operations give great flexibility in data analysis and representation. Several built-in functions allow the user to perform a statistical analysis of a point distribution. The “scalar distribution” function calculates the distribution of any quantity loaded in the table data and plots it as a histogram. The “correlation filter” calculates the linear two-point correlation function of a point set. This is defined as the probability $\delta P$ of finding a point in a randomly chosen volume $\delta V_i$ and a point in another volume $\delta V_j$ separated by a distance $r$. The
two-point correlation function of VisIVO is based on the three-dimensional (3D) counterpart of the Peebles & Hauser estimator:

$$\xi_{\text{Peebles}} = \frac{DD(r)}{RR(r)} \left( \frac{N_{\text{rd}}}{N} \right) - 1 \quad (1)$$

(Peebles & Hauser 1974), where $N_{\text{rd}}$ is the number of points in an auxiliary random sample, $DD(r)$ is the number of all pairs of points with separation inside the interval $[r - dr/2, r + dr/2]$, and $RR(r)$ is the number of pairs between the data and the random sample with separation in the same interval. The random sample must have a density 2.5 times the density of the real data set. The box is divided into a number $N_{\text{bin}}$ of cubic subintervals. Then a frequency histogram of the pair distances of particles is constructed. The calculation of $RR$ and $DD$ is performed with a Monte Carlo integration. The Fourier transform of the correlation function is represented by the power spectrum, which can be estimated by VisIVO as well. The power spectrum of a set of $N$ massive particles can be calculated as

$$P(k) = \langle |\rho(k)|^2 \rangle, \quad (2)$$

where $P(k)$ is the power spectrum, $k$ is the three-dimensional wavenumber: $k_i = 1/r_i$, with $r_i$ indicating the $i$th component of the spatial position of a point, and $\rho(k)$ is the Fourier transform of the mass density field. The power spectrum provides the same statistical information of the correlation function, but it is much faster to compute. However, in this implementation, periodic boundary conditions are required. Furthermore, in general, the spatial resolution is worse than that of the correlation function. In fact, in order to calculate the power spectrum, a cloud-in-cell algorithm distributes a constant value for each point (its mass) on a regular structured mesh with periodic boundary conditions. The mesh resolution sets the maximum wavenumber that the power spectrum can be calculated on. With this procedure, a mesh-based mass density distribution is reconstructed and a fast Fourier transform (FFT)–based approach can be used to estimate $P(k)$. The last available analysis tool is for Minkowski functionals (MFs). They describe the geometry, the curvature, and the topology of a point set (Platzer & Buchert 1995). In a three-dimensional Euclidean space, these functionals have a direct geometric interpretation. The first functional represents the volume $V$ of a structure, while the second one represents the surface area $A$ and is a measure of the geometry of the distribution. The third functional corresponds to the integral mean curvature $H$ of the structure’s surface. It represents a measure of the distribution topology. The MFs algorithm implemented in VisIVO associates a “covering sphere” of radius $r$ to each data point. The size, shape, and connectivity of the spatial pattern that is composed by the union set of the spheres change with the radius and can be used as a diagnostic parameter. In particular, VisIVO computes the reduced values of the Minkowski functionals, $\Phi(\mu)$ with $\mu = 0, 1, 2$, that are the ratio of the MFs of the actual distribution to the MFs of the same number of disjointed convex bodies. Their values always start from unity: for small radii,
all the covering spheres are disjointed. In the third functional 
φ₃, the first zero provides an estimate of the percolation 
threshold. A spongy structure, such as a Poisson distribution, 
gives lower values for φ₃, while higher values indicate structures 
with few big filaments or tunnels.

2.3. VisIVO and VizieR

In the age of the Virtual Observatory, data collections are 
distributed between various sites. They are accessible to user 
applications via standard technologies such as the Web service 
WSDL/SOAP protocol. One of the services exploiting this Web 
service technology is VizieR, version 2, which is available on 
the CDS servers. Although it is still in beta, the service will 
also be available at ADS, ADAC, and CADC after the final 
release. VizieR is a database that archives, in a homogeneous 
way, thousands of astronomical catalogs gathered over decades 
by the CDS and participating institutions. The new Web service 
interface provides access to the VizieR database of astronomical 
catalogs by adding four new methods to the old interface: 
coneCatalogs, coneResults, ADQLRequest, and getAvailability.

VisIVO, using the Axis C/C++ library, implements an in-
terface to the service. It is able to get the list of available servers 
using the getAvailability method, to get the list of valid pa-
rameter values to pass to the coneCatalogs and coneResults, 
and using the last two methods to get metadata and data (in 
VOTable format) about catalogs depending on the given 
parameters.

In this way, VisIVO is able to directly query the VizieR 
Web service to retrieve data from it and visualize them as if 
they were local data. The interaction with the service is trans-
parent to the user. The user need only fill in specified fields 
with the parameters defining the data that he wants to download. 
The result of this operation is a list of catalogs and, on selecting 
one of them, data can be visualized as if they were in a file or 
saved on the disk in the VOTable format.

2.4. VisIVO Performance

VisIVO’s recommended system requirements are such that 
it can be used on a consumer laptop personal computer. 
VisIVO’s performance is mainly constrained by the system 
RAM size; each loaded file will create a sequence of float 
arrays: the more memory the user has on his system, the more 
data he can load. On a laptop with 1 Gbyte of RAM running 
a GNU/Linux system, VisIVO is able to load and handle 
2 million points interactively, while up to 16 million points 
can be loaded in about 5 s, although the visualization then 
becomes quite slow. On Microsoft Windows XP, VisIVO 
is limited to 8 million points due to Windows’ memory 
management.

Table 1 shows the execution time in seconds of typical op-
erations that the user performs with VisIVO. The tests were 
carried out both with GNU/Linux systems and Microsoft Win-
dows XP (the column headers containing the letters “GL” and 
“MW,” respectively) and ATI Mobility Radeon X600 with 
64 Mbyte and ATI Radeon Xpress 1100 graphics cards. The 
notation 16M and 8M identifies tests with 16 and 8 million row 
data sets, respectively. HPS identifies our high-performance dual 
processor system equipped with two AMD Opteron Dual Core 
280 (2.4 GHz), 8 Gbyte of RAM, and a 7200 rpm SATA hard 
disk. LAP1 and LAP2 identify two consumer laptops; LAP1 is 
equipped with an Intel Pentium M 740 (1.73 GHz), LAP2 is 
equipped with an AMD Turion TL50 X2 (1.6 GHz), and both 
have 1 Gbyte of RAM and 5400 rpm hard disks. The “import” 
test consists of loading a binary file containing six fields, the 
“geometry” test consists of defining a 3D point distribution 
from three of the loaded fields, the “display” test consists of 
visualizing the geometry, and the “distribute” test consists of 
distributing a scalar value associated to the 16 million points 
on a regular grid with 128³ cells. The “power spectrum” test 
computes the power spectrum of the point distribution, and the 
“correlation” test computes the correlation function of the point 
distribution. The “subsample” test performs a geometrical sub-
sample of the point distribution, and finally the “extract cluster” 
test performs an extraction of the points within an isosurface 
of one of the scalars associated with the points.

3. APPLICATION INTEROPERABILITY THROUGH 
PLASTIC

The capabilities of VisIVO are extendable through an ap-
plication interoperability protocol called PLASTIC (Platform 
for Astronomy Tool Interconnection; Boch et al. 2006). 
Equally, through PLASTIC, VisIVO’s functionality is made 
available to other applications. The motivation for PLASTIC 
is the desire to leverage the abilities of different desktop ap-

| Test                  | HPS-GL-16M | LAP1-GL-16M | LAP2-GL-16M | LAP1-MW-8M | LAP2-MW-8M |
|-----------------------|-----------|------------|-------------|------------|------------|
| Import                 | 5         | 4          | 16          | 7          | 5          |
| Geometry               | 18        | 20         | 35          | 6          | 7          |
| Display                | 44        | 27         | 30          | 25         | 21         |
| Distribute             | 8         | 9          | 11          | 10         | 5          |
| Power spectrum ..........| 13        | 20         | 16          | 8          | 8          |
| Correlation            | 18        | 30         | 25          | 16         | 14         |
| Subsample              | 4         | 5          | 24          | 2          | 4          |
| Extract cluster ........| 310       | 354        | 420         | 44         | 60         |
applications in a seamless way. Scientific applications such as VisIVO are being continuously enhanced with new features, and while this is of great benefit to users, there are some limitations to this “bigger is better” approach. Inevitably, there will be some overlap between the applications’ feature sets as users demand capabilities from other applications in their own favorite tool, leading to a duplication of effort within the scientific software community. As applications become more powerful their resource footprint usually expands, and their increased complexity may make them more difficult to maintain and to use. The alternative and complementary approach is to encourage collaboration between applications, each a specialist in a particular task. This approach enables the user to assemble a suite of tools according to his personal requirements. Exporting and outsourcing functionality to other applications has been explored by CDS: Aladin (Bonnarel et al. 2000) exposes a publicly documented interface (“VOApp”) that makes it possible for third parties to write plug-ins that expand Aladin’s capabilities and reuse its functionality. However, Aladin’s plug-ins are restricted to being Java applications that can run inside the same JVM, which sometimes leads to resource, packaging, and class compatibility problems. The VisIVO and Aladin developers overcame this constraint by making VisIVO, a C++ application, control Aladin through the latter’s scripting interface. However, the architecture is no longer symmetrical and Aladin is unable to control VisIVO in return. It was recognized that there was a need to generalize the VisIVO/Aladin interoperability to arbitrary applications, and thus the VOTech consortium, of which the VisIVO team is a founding member, created the PLASTIC interoperability protocol. Through PLASTIC, applications can share data and link views. Data exploration using disparate linked views of the data is not a new idea; for example, the Mirage and xmdvtool applications each supports several visualization methods allowing the user to explore data simultaneously using different methods. PLASTIC extends this concept to allow linked views across applications. PLASTIC works through a locally running daemon application called a “Plastic Hub” and is derived from the technology developed for the Astro Runtime (Winstanley et al. 2007). Applications communicate with the Plastic Hub using one of several protocols: different protocols are supported to make it easier for application developers to adopt PLASTIC. PLASTIC does not define a fixed application programming interface (API) of operations that all applications must support. Instead, it employs a simple interapplication messaging system: applications send each other messages requesting certain actions. These are sent via the Plastic Hub, which then routes them to their destinations. The current set of messages includes “load this table” and “select/highlight these data points” but can be extended by application developers as new ways of collaborating arise. The advantage of the hub-based architecture is that individual applications need only understand one of the hub’s communication protocols, and the hub is responsible for any required translation. Furthermore, applications can dynamically discover other applications and their capabilities by interrogating the hub. PLASTIC is platform and language neutral and, at the time of writing, has been incorporated into more than a dozen applications written in Java, C++, Python, Perl, Tcl, and JavaScript/HTML, including Topcat, Aladin, VisIVO, VOSpec, AstroWeka, and Reflex.

3.1. PLASTIC Interoperability in Action

The following example illustrates how VisIVO’s PLASTIC interaction works in practice, with several applications exploiting each other’s strengths to explore data taken from the paper by Digby et al. (2003). The AstroGrid Workbench is first used to search for suitable data in the SuperCOSMOS Science Archive (SSA) and Sloan Digital Sky Survey (SDSS). The astronomer begins his analysis by starting Topcat, which connects to the Plastic Hub. The astronomer then uses PLASTIC to send the SuperCOSMOS data directly to Topcat (Fig. 3).

The SuperCOSMOS data do not contain the attributes that the astronomer needs, so he uses Topcat to add synthetic columns to calculate the color indices of the sources and their reduced proper motions. A scatterplot of the $r - i$ color index against the reduced proper motion reveals that the sources fall into three populations: white dwarfs, subdwarfs, and main-sequence stars (Fig. 4).

To fully understand this data set the astronomer starts VisIVO, since it specializes in plotting multidimensional data. VisIVO connects with the Plastic Hub, and shortly afterward the Workbench and Topcat menus update to reflect the fact that VisIVO is running and able to receive VOTables. The astronomer sends the augmented data from Topcat to VisIVO and uses VisIVO to create a 4D plot of $r - i$ color index, $g - r$ color index, reduced proper motion, and magnitude (using the latter to choose the color of the points from a lookup table; Fig. 5).

The white dwarf population can then be selected in Topcat and is automatically highlighted in VisIVO through PLASTIC for further exploration (Fig. 6). Finally, interesting objects can be selected using VisIVO’s picker tool and sent via PLASTIC to Aladin for overlaying over an image so that the astronomer can see their spatial distribution. This workflow is, of course, greatly simplified. In reality, the data would be transferred back and forth between VisIVO and other applications, with inter-

---

8 See http://aladin.u-strasbg.fr/java/nph-aladin.pl?frame = plugins.
9 See http://eurovtech.org.
10 See http://esavo02.esac.int:8080/vospec.
11 See http://www.eso.org/sampo/reflex.
12 See http://www2.astrogrid.org/desktop.
13 See http://surveys.roe.ac.uk/ssa.
14 See http://www.sdss.org.
Fig. 3.—GUIs showing process of sending SUPERCOSMOS data from the Workbench to Topcat. [See the electronic edition of PASP for a color version of this figure.]

esting clusters extracted and spurious data removed. It could even be sent to statistical applications such as AstroWeka\(^\text{15}\) and Eirik\(^\text{16}\) to aid the astronomer in identifying clusters and trends. While some of the above workflow could be accomplished by saving the data set from one application and loading it in the next, PLASTIC makes the operation seamless.

4. SCIENTIFIC CASES

In this section, as an example of the practical usage of VisIVO, we show how the tool can help the researcher in the analysis of two different problems: the classification of galaxies between star-forming and quiescent objects and the detection of shock waves in galaxy clusters.

4.1. Galaxy Classification

One of the most widely used methods relies on the intrinsic color versus absolute magnitude diagram: in this representation, redder objects, assumed to be quiescent, occupy a well-defined locus, also known as the “red sequence.” On the other hand, having spectroscopic information at hand, the equivalent width of the [O\(^\text{II}\)] emission line is a well-known tracer of ongoing star formation activity, while the depth of the break at 4000 Å is an indicator of stellar age (the higher the break, the older the stellar population is). The question that we ask ourselves is whether, using one or the other of these indicators, we select the same population of galaxies and also whether these indicators show some evolution with redshift. Starting from the VVDS epoch 1 survey (Le Fèvre et al. 2005), we have extracted a subsample of galaxies with secure redshift, for which the equivalent width of the [O\(^\text{II}\)] emission line (simply named [O\(^\text{II}\)] from here on) and the depth of the 4000 Å break (here-

\(^\text{15}\) See http://astroweka.sourceforge.net.
\(^\text{16}\) See http://wiki.eurovotech.org/twiki/bin/view/VOTech/EirikDataNavigation.
after D4000) have been measured with good confidence (see Franzetti et al. 2007 for details). Our global initial sample consists of 4640 galaxies, and for each object we have a measurement of $V$ absolute magnitude, intrinsic $U - V$ color, redshift, $[\text{O II}]$, and D4000. As a first step, we import the ASCII file where such information is stored into VisIVO and do a

Fig. 4.—Three populations as seen in Topcat. [See the electronic edition of PASP for a color version of this figure.]
first rough visualization of D4000 versus [O ii] versus z. Such a plot shows that there are ~25 objects for which an anomalously large [O ii] and/or D4000 has been measured. As we are currently interested in the global properties of the sample and not in the outliers, we use the SubSample function to select the bulk of our data, eliminating such outliers. In Figure 7 (left), the 3D representation of the D4000 versus [O ii] versus z for the remaining objects is shown. The color scale indicates the absolute magnitude associated with each object.

Two effects are immediately visible from Figure 7 (left): the first is the to be expected selection effect on galaxy luminosity: the further we go in redshift, the more luminous are the galaxies that we have in our sample (we have fewer and fewer blue points as redshift increases). The second effect that we see is a tendency for D4000 to decrease with increasing redshift. Before further inspecting this decrease, which could be scientifically very promising, we get rid of the luminosity selection effect by extracting from our global sample a volume-limited subsample; i.e., we include only objects having an absolute magnitude visible throughout the whole redshift range. This is easily done by displaying the absolute magnitude and redshift and cutting the sample at an absolute magnitude $M_V \leq -21$ using the SubSample function (see Fig. 7, right). We save such selections in an ASCII file for further use.

Now that we have a subsample clean from selection effects, we can go back to our original problem, and first of all we want to see whether the effect of D4000 decreasing as redshift increases is still noticeable.

In Figure 8 (left), we show the D4000 versus [O ii] versus z data cube, where for clarity we use four different colors for different D4000 ranges: at higher redshifts, not only does D4000 assume smaller values, but rotating the cube we see that [O ii] has a tendency to be higher (on average) with increasing redshift. These two effects point toward a higher star-
forming rate at higher redshift, and especially for $z \geq 1.1$. If we use the second star formation indicator, i.e., the color-magnitude diagram versus redshift (see Fig. 8, right), we see again that at higher redshift, blue galaxies are more abundant than red galaxies. As a last point, we want to see whether galaxies that would be defined as quiescent in a color-magnitude diagram are also spectroscopically quiescent (i.e., galaxies belonging to the red sequence should not show emission lines)
Fig. 8.—Evolution of abundance of star-forming galaxies with redshift. Left: Spectrophotometric indicators D4000 and [O ii]; right: photometric indicators $U - V$ and $M_c$. In both cases, a trend of increased abundance of galaxies showing low star age (left) and bluer colors (right) at higher redshifts is visible. [See the electronic edition of PASP for a color version of this figure.]

and vice versa, whether galaxies that show an old stellar population are indeed red in intrinsic color.

Displaying again the D4000 versus [O ii] versus redshift data cube (Fig. 9, left), we now use color to indicate the galaxy intrinsic $U - V$: as expected, red galaxies show, on average, a higher stellar age indicator. Still, there are some red galaxies showing a low value for D4000, as well as blue galaxies with an old stellar population (high D4000), and especially the intermediate class of colors (light gray dots) span the whole plane. This high degree of mixing does not seem to depend on redshift. Also in the color-magnitude diagram (Fig. 9, right), this is clearly visible: we have now highlighted in dark gray the galaxies that show a D4000 $\geq 1.5$ (old stellar age) and a low value of [O ii] (no sign of star formation); these galaxies have a tendency to have redder colors but are not necessarily the reddest ones, and not all the red galaxies fall in this category.

To summarize, using VisIVO exclusively, and without any programming knowledge whatsoever, we have been able to inspect...
our sample and (1) see the volume selection and extract a bias-free subsample, (2) find an interesting evolution of galaxy stellar age with redshift, which deserves further statistical inspection (see D. Vergani et al. 2007, in preparation, for a more exhaustive discussion of this topic), and (3) show that selecting red galaxies on the basis of spectroscopic features versus photometric colors does not give the same sample. The degree of contamination in the different cases is thoroughly discussed in Franzetti et al. (2007).

4.2. Detecting Shocks in Galaxy Clusters

Cosmic shock waves are believed to be among the most efficient accelerators of particles in the universe. The final spectrum of the accelerated particles is influenced by the complex interplay between the growth of cosmic structures, the geometry of the shock waves, and the number of shocks that a particle may experience during its life. A consistent and complete description of the dynamics of cosmological shock waves is far from being achieved. However, numerical simulations provide a valuable contribution to the comprehension of this process. We have performed a large number of simulations using the Enzo code, which is an adaptive mesh refinement cosmological code,\(^\text{17}\) developed by Bryan et al. (1995). The Enzo code couples an N-body particle-mesh solver with a PPM Eulerian adaptive method for ideal gas dynamics by Colella & Woodward (1984). For all the runs, we adopted the standard “concordance” model, with density parameters \(\Omega_m = 0.226\), \(\Omega_b = 0.73\), Hubble parameter \(h = 0.71\), and initial spectrum normalization \(\alpha_s = 0.94\). In order to have a large cluster statistics and a wide cosmological volume, we simulated several data cubes, finally assembling an overall volume of \(\sim 100\) \(h^{-1}\) Mpc per side.

Shock identification has been performed using a novel method that studies the structure of the baryonic velocity field and evaluates Mach numbers through velocity jumps, developed by F. Vazza et al. (2007, in preparation). Cells that present velocity jumps that can produce a shock wave (Landau & Lifshitz 1959) are tagged as “shocked cells,” and their Mach number is estimated as

\[
M = \left[ 3(1 - (4\Delta v)) \right]^{1/2},
\]

where \(\Delta v\) is the fluid velocity jump across the shock, in the reference frame of the shock itself, \(\Delta v = v_{\text{post}} - v_{\text{pre}}\). This technique, coupled with the unprecedented good resolution of our data in the outermost regions of virialized structures (where shocks very often occur), allows us to study dynamical regimes never seen before. The use of a visualization code such as VisIVO is of great importance because it allows us to detect and follow the behaviors of shock patterns that are characterized by very complex volume-filling properties, which had always been erased by most standard reduction analyses and can be of primary importance to correctly describing the observational impact of these mechanisms. Data from the simulations are saved in raw binary and HDF5 formats, which generates a huge data collection. In the following, for simplicity, we focus on the results of one of the simulations, characterized by 160\(^4\) mesh cells and the same number of N-body particles. The computational box is 40 Mpc, and the presented results are at redshift \(z = 0.0\). However, all the conclusions can be extended to all the outputs of our data collection. VisIVO is used to explore the data in an intuitive and effective way and to focus on the interesting and, sometimes, unexpected aspects. Data have been combined using the “merge tables” function. The “math operations” tools have been used to derive new data sets, such as logarithmic quantities or velocity magnitude. Cosmological simulations produce extremely complex structures, characterized by clumps, filaments, sheets, large voids, etc. In Figure 10, the distribution of particles representing the dark matter is shown in the top panel. The bottom panel presents a comparison between the dark matter and the gas mass distributions, these two quantities being represented by isosurfaces. Dark and light gray isocontours show the gas distribution at different mass density values. For comparison, dark matter isosurfaces (white) are superimposed. Dark matter mass density is calculated by smoothing the particle masses on the computational grid using the VisIVO point distribute function. Gas and dark matter mass distributions are similar, since they are both driven by a common gravitational potential. The temperature distribution, shown in Figure 11 (top) using a volume-rendering technique, follows the overall mass distribution. However, a shock wave that forms during the gravitational collapse events expands rapidly, raising the gas temperature to \(10^6-10^7\) K on volumes up to \(\sim 10^3\) Mpc. In order to have a quantity directly comparable to observations, we have calculated the gas X-ray emission due to thermal bremsstrahlung directly using the VisIVO Math tool. The X-ray distribution is shown using the VisIVO slider utility in Figure 11 (bottom). As expected, the emission is strongly associated with the clusters, since it depends principally on the mass concentration.

Once general features of the data have been analyzed, we switch to the identification and characterization of shocks. Shock fronts and the corresponding Mach numbers are identified by our “velocity jumps”-based procedure. The shock density distribution is presented in Figure 12.

In the top panel, points are colored according to their temperature. Most of the high-Mach (Mach number \(M > 10\)) shocks are in low-temperature regions, therefore far outside the galaxy cluster virial radius. The distribution presents two peaks corresponding to different values of density. The right peak (larger density value) is observed in any kind of simulation, and it corresponds to shocks forming in the outer outskirts of clusters and along filaments, where the matter accretion is still strong. The left peak corresponds to rare voids, which represent the low-density tail of the probability density function in a ΛCDM

\(^{17}\) See http://cosmos.ucsd.edu/enzo.
universe. It has never been detected in any previous numerical simulations. The double-peaked feature of the distribution is confirmed when all the data sets available in our collection are taken into account, as shown in Figure 12 (bottom). This result is determined both by the improved accuracy of our shock reconstruction algorithm and by the simulation code. In fact, some of the previous studies are related to SPH-based simulation, which cannot treat in a proper way the hydrodynamics of underdense regions, which are poorly sampled by the particle-based approach. Figure 13 shows a 4D phase space distribution of a subsample of points extracted from the whole simulation. The VisIVO “randomizer” function allows the user to select a small number of points in order to proceed faster in the analysis, without requiring huge computational resources.

Fig. 10.—Mass distribution of dark and baryonic matter in a $40^3 \, \text{Mpc}^3$ cosmological simulation. Dark matter particles and gas isosurfaces are compared. [See the electronic edition of PASP for a color version of this figure.]

Fig. 11.—Temperature distribution of the simulated gas (left), rendered using a ray-tracing technique. X-ray maps (right) are visualized using an Ortho Slice utility. [See the electronic edition of PASP for a color version of this figure.]
The figure shows the density–temperature–Mach number distribution of sampled points, which are further colored by the Mach number itself (top panel) and the velocity (bottom panel). The top panel shows that high-Mach points (dark spheres on the bottom) are concentrated at low temperature. High-density regions also have high temperatures. However, low-density regions have temperatures in a wide range, from $\sim 0$ to $\sim 10^7$ K. This is due to the fast shock propagation outside collapsing structures (see also Figs. 10 and 11). The bottom panel emphasizes the density–Mach number relation. Still present, even if less clear due to lower statistics, is the low-density Mach number peak. The L-shaped phase space distribution (high Mach–low temperatures, low Mach–high temperatures) is also evident. The velocity seems not to present a specific trend with respect to the other quantities. All these features are also present using the complete data set and in any other random extraction,
proving that there are not spurious effects connected to statistical biases.

In Figure 14, we show the geometric distribution of the shocks, visualized as 3D surfaces with Mach \( \geq 2.5 \) and selected according to the mass density values. The top panel shows the shocks on the high-density peak of the Figure 12 distribution. The bottom panel shows the shocks in low-density regions. The presence of such vast surfaces of high-Mach shock in underdense regions suggests that these regions must be carefully taken into account in order to properly evaluate the net amount of energy that ends up in particle reacceleration (Brunetti 2003). A further analysis of the data is ongoing. Quantitative data analysis tools are necessary at this point to estimate the results that have been revealed and emphasized using VisIVO. The “visual-based” method has proved to be extremely effective in getting an immediate and effective approach to a complex and huge data set, selecting and extracting interesting features that would otherwise be hardly detectable.

5. SUMMARY

In the previous sections we have shown how an advanced visualization tool such as VisIVO can be used for helping the researcher in analyzing complex data. Visualization cannot provide quantitative results, but it allows the user to have an immediate and intuitive approach to the data. The various 3D rendering techniques supported by VisIVO, together with the possibility of visualizing complementary quantities with colors, glyphs, and vectors, allows the user to discriminate between data features at a glance, pointing out special characteristics and focusing on interesting regions. The software also implements a limited but effective set of statistical tools, which can be used to make quick estimates of the properties of a distribution. Mathematics functions let the user derive new fields starting from the original ones. VisIVO is being developed to follow the IVOA recommendations and standards, so that it is interoperable with the Virtual Observatory framework. Furthermore, it supports the PLASTIC protocol to allow the user to use the software together with other tools, such as Aladin or Topcat, which have data analysis capacities complementary to those of VisIVO. In this way the researcher can have a complete and customized cooperating set of tools that makes his or her research activity more and more efficient and focused on scientific issues.

New features of VisIVO will focus on exploiting new hardware architectures that are rapidly appearing in many desktop machines, such as 64 bit and multicore systems, subject to making effective use of such capabilities in the tool. The opportunity to use VisIVO in data centers and on dedicated visualization servers will drive the new releases of the code. We will also investigate the possibility of interacting with remote data services that support the SNAP protocol, which is an emerging protocol for retrieving data from numerical simulations. Finally, VisIVO will be developed to integrate with the VO’s Theoretical Data Archive framework. VisIVO will display a subset of the whole data file, which will generally be very large, and will allow the user to select a spherical or
rectangular region and retrieve, through a remote service, the extracted subsample.

We gratefully acknowledge useful discussions with F. Pasian and R. Smareglia from INAF Astronomical Observatory of Trieste, and we would like to thank F. Genova, F. Bonnarel, and T. Boch of CDS–Observatoire Astronomique de Strasbourg and R. Mann of the Institute for Astronomy, University of Edinburgh, for the useful help they gave us, Franco Vazza (Department of Astronomy, INAF-IRA Bologna) for the suggestions and the contributions given for the analysis of cosmological simulations part, and L. Santagatì of INAF–Catania Astrophysical Observatory for the useful revision of the text. Numerical simulations have been performed using the IBM-SP5 system at CINECA, Bologna, with CPU time assigned under an INAF-CINECA agreement (2006/2007).

REFERENCES

Boch, T., Comparato, M., Taylor, J., Taylor, M., & Winstanley, N. 2006, PLASTIC: A Protocol for Desktop Application Interoperability, IVOA Note 01 (Strasbourg: IVOA), http://ivoa.net/Documents/latest/PlasticDesktopInterop.html
Bonnarel, F., et al. 2000, A&AS, 143, 33
Brunetti, G. 2003, in ASP Conf. Ser. 301, Matter and Energy in Clusters of Galaxies, ed. S. Bowyer & C.-Y. Hwang (San Francisco: ASP), 349
Bryan, G. L., Norman, M. L., Stone, J. M., Cen, R. Y., & Ostriker, J. P. 1995, Comput. Phys. Commun., 89, 149
Colella, P., & Woodward, P. 1984, J. Comput. Phys., 54, 174
Digby, A. P., Hambly, N. C., Cooke, J. A., Reid, I. N., & Cannon, R. D. 2003, MNRAS, 344, 583
ESA. 1997, The Hipparcos and Tycho Catalogues (ESA SP-1200; Noordwijk: ESA)
Franzetti, P., et al. 2007, A&AS, 465, 711
Landau, L. D., & Lifshitz, E. M. 1959, Fluid Dynamics (Oxford: Pergamon)

Le Fèvre, O., et al. 2005, A&A, 439, 845
Ochsenbein, F., Bauer, P., & Marcout, J. 2000, A&AS, 143, 23
Peebles, P. J. E., & Hauser, M. G. 1974, ApJS, 28, 19
Platzoder, M., & Buchert, T. 1995, in Proc. First SFB Workshop on Astro-Particle Physics, Applications of Minkowski-Functionals to the Statistical Analysis of Dark Matter Models, ed. A. Weiss et al. (Muenchen: Technische Univ.), 251
Schroeder, W., Martin, K., & Lorensen, B. 2004, The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics (3rd ed.; Clifton Park: Kitware)
Springel, V. 2005, MNRAS, 364, 1105
Viceconti, M., et al. 2004, in IEEE Proc. Eighth International Conference on Information Visualisation (Washington: IEEE), 15
Winstanley, N., et al. 2007, in Proc. Astronomical Data Analysis Software and Systems XVI, Astro Runtime: Client-Side Middleware for the VO, in press.