

Diverse data to diverse visualization systems end to end

Julian Gallop

CCLRC Rutherford Appleton Laboratory

Abstract

The paper introduces an approach to bridging the gap between diverse data sources and software for visualization and data analysis. The possibility of XML-based languages for data structure description and the introduction of Grid services allows the possibility of partitioning the problem to straightforward components. The work takes place within gViz, a project within the UK e-science Core Programme.

1. Introduction

R&D in Grids and e-science is transforming access to data anywhere. Developments in OGSA and the convergence with Web Services are providing a widely accepted basis for managing distributed data. It is envisaged that the emphasis will shift towards making better use of existing data archives allowing data acquisition to be focused on situations where the data is novel or where there is always a requirement for current, time-stamped data.

However the already well-known problem of numerous data formats and models persists. This is despite many valuable initiatives. Valuable legacy data is slowly being converted. Self describing data interfaces have been available for more than 10 years (netCDF [9] and HDF, the latter now HDF5 [3]). Practitioners within application domains are collaborating and defining new XML-based specifications for their domain (NERC DataGrid [10], VOTABLE [13], Marine XML [7], ESML [4]).

But along with the many data formats go the many data analysis and visualization systems and these are also likely to persist. (In this paper we will for simplicity refer to visualization systems, implying data analysis also) As shown in the figure below there will continue to be a gap between the multiple data formats and models and multiple preferred visualization systems.

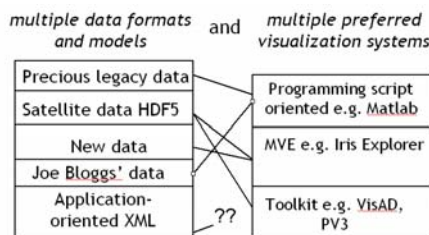


Figure 1

Conventional approaches to this might be to write yet another convertor or for a team of collaborators to decide for simplicity that they will fix the choice of a particular data analysis system within the collaboration.

However the idea of a Grid-enabled Virtual Organisation (VO) encourages teams that form, change and disperse. Such teams are likely to be multidisciplinary and individuals in such a team are likely to have previous experience of diverse visualization systems. The temporary nature of VOs discourages the ad hoc solutions mentioned earlier.

2. Outline of approach

Our proposed approach is to concentrate on finding a solution that provides an end to end framework for a solution to the problem. Thus it does not displace the new efforts referred to in the introduction, but rather suggests a framework in which they might be used. We investigate whether the tasks of analyzing the format of a dataset can be separated from the task of building the format that the data analysis system needs. There appear to be 2 approaches as illustrated in Figure 2.

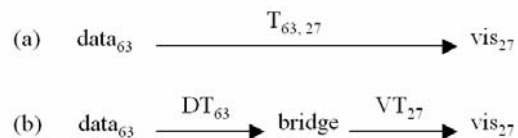


Figure 2

Approach (a) would analyse (for instance) the 63rd data format and the 27th visualization system and automatically create a transformation from one to the other. Approach (b) would identify a single bridging language which would act as an intermediary between the dataset and the visualization system. Approach

(b) in its purest form suffers from the disadvantage that a very large dataset would undergo two transformations instead of one. However this disadvantage would not apply to the metadata and approach (b) is likely to be more transparent to anyone making use of it. We also note that approach (b) allows appropriate expertise to bear on each stage. Combining the best of both approaches appears to be suitable here.

3. Characterisation

Before describing the approach in more detail, it is useful to characterize datasets and visualization systems.

3.1 Characterising data sources

Data sources can be characterized along a number of axes.

- A progression from legacy, de facto data sources to present day initiatives using XML. Legacy sources may be ad hoc or may be publicly available in a well defined form.
- Legacy data sources vary widely in their structure and metadata quality. Nonetheless within a given class of legacy data sources, there is generally a regularity that can be exploited.
- Application dependent/independent
- Data sources can be private by intent; private by default; public but restricted; fully public.
- Data sources can be regular, which generally means that the spatial data can be implied, or unstructured where the spatial data is completely specified.
- Whether the dataset has internal references, such as indexing
- Single dimension/Multidimensional
- Many data sources are available in the form of a relational database.

3.2 Characterising visualization systems

Also, visualization and data analysis systems can be characterized along a number of axes:

- Fixed function or adaptable either by programmable scripts or by visual composable networks
- Some visualization systems can read and deal with volume or multidimensional data
- Some visualization systems can read unstructured data where the geometry has to be specified exactly, not relying on implicit regularity.

- Formats that can be read: most visualization systems possess a native format and can read some other popularly available formats. Several can read pre-XML self-describing models (netCDF [9] and HDF5 [3]). A few can read an XML file directly.

4. An end to end approach

We describe the end to end approach in more detail.

4.1 Some principles

There are a number of principles that can be followed here:

- Adopt XML, since this is widely accepted as a basis of specifications for exchange and the availability of implementations for validation and transforming is growing. Thus a non-XML format is converted to an XML one as soon as possible. In principle, every item in a datum could be tagged with XML. However for any dataset which stresses present day resources, very large data objects are not tagged but are accessed by conventional methods, which are identified and described by XML tags. All other properties in the dataset can be described with XML.
- Adopt a descriptive approach, not prescriptive. A prescriptive approach would specify the format in which the dataset is stored, which is unrealistic. Here the attempt is to use an XML-based language to describe the dataset.
- Decompose the transformation into single purpose components, which can be implemented by a existing tool such as XSLT and potentially could be relocated on different processors.
- Allow for the situation where there are numerous datasets for a given format. This means that a solution that requires describing a specific instance dataset by hand is not acceptable.
- Avoid undue loss of speed. For example additional transfers or conversions of very large datasets should be avoided.

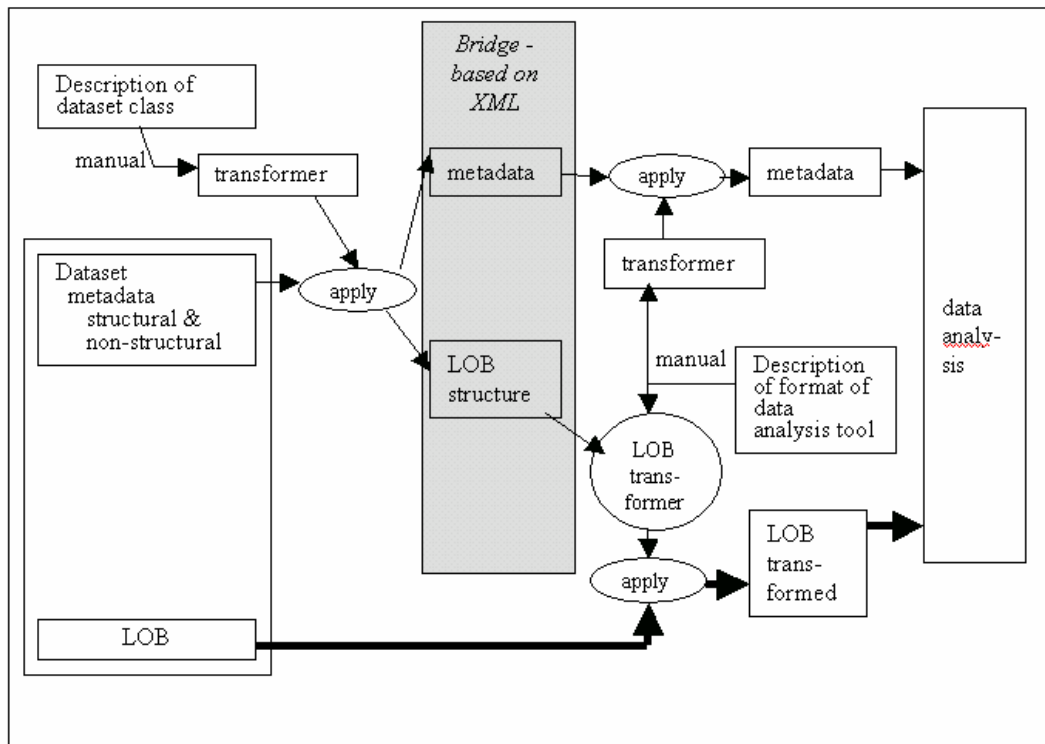


Figure 3

4.2 A framework for an end to end approach

Figure 3 shows a possible framework for an end to end approach. We briefly describe the important aspects of this.

The bold arrows indicate the flow of potentially large data. The acronym LOB (Large Object) is by analogy with the acronym BLOB, but here datasets can be in the form of plain text as well as binary. Note that only one transformation takes place.

Three transformers appear in the figure and their action is indicated by the node “apply”. The one at top left is constructed manually and, for a given class of datasets, is designed to transform metadata in any datasets in that class to a bridge XML-based language. Most frequently at present, the dataset metadata is not expressed in XML, so this transformer could be expressed in the form of a script which can handle file splitting and regular expressions. Where the dataset metadata is expressed in XML as it increasingly will be, the transformer can be expressed as an XSLT script.

This transformer has two outcomes, both of which can be classed as metadata and both are expressed in the bridge language. Here we do not aim to construct or prefer a particular view of metadata. For our purposes, we distinguish

the metadata, which defines the data model and format of the dataset (structural metadata) from all other metadata (non-structural) which is essential to the analyst but treated separately.

The structural metadata, which we label here as the LOB structure, is expressed in a form which enables the LOB transformer to be built. It must include: the means of access to the dataset, which includes its URI (which may result in accessing multiple remote files); and the operations (whether plain text or binary) necessary to parse and skip the individual data records. There is also the potential to make use of OGSA/DAI [14] in this context.

The LOB transformer, is responsible for transforming the potentially large data into the form acceptable for the visualization tool. This is generated from:

- (a) the LOB structure, outlined above and
- (b) the description of the format accepted by the visualization tool.

The process of generating this transformer needs to recognize cases where no or minimal transformation need take place.

The transformer at near top right is also constructed manually. For a given visualization or data analysis tool, it is designed to transform metadata expressed in the XML-based bridge

language to the most appropriate form required by the tool. This can be written using XSLT.

The bridge language, shown within the box with a grey background, needs to possess sufficient expressive power. Current investigations are being made using the XDF XML-based data description language [5]. It contains provision for structural metadata, which, present work suggests, needs to be extended to express the requirements of the LOB structure.

4.3 Further points

The framework can be used to handle other issues.

Subset extraction: existing initiatives such as OPeNDAP/DODS [11] allow subsets of remote datasets to be extracted. This capability would be required in the more general framework here. The subset definition would be used to generate a modified LOB structure

More complex structures involving internal references: Datasets often possess an internal structure. Examples of this include geographic datasets which may define shapes in terms of boundary segments and finite element data which may contain property data defined in terms of geometric constructs such as cells, edges or nodes. Once expressed as XML, transformations using XQuery should be considered instead of XSLT.

Component allocation: Since the framework defines several stages of transformation, any could be allocated to different parts of the Grid. For example, expensive dataset conversion could take place on a specialist service which contains generous temporary workspace.

Multiple destinations: Where it is known that large datasets will be delivered to multiple destinations, alternative optimization tactics may be required.

5. Conclusions

An approach for enabling visualization systems to access diverse data sources is being investigated and a framework for this is presented.

There are similarities of approach with the Data Format Description Language [2]. This is an initiative within the Global Grid Forum and it aims to provide a description language for data, whether in binary or plain text, that is not otherwise expressed in XML. It is a possible candidate for a bridging language within the framework described in this paper, but would also need complementing with a means of expressing metadata.

Performance problems are being reduced to a minimum by leaving the potentially large tables in their original form until the destinations are known.

6. Acknowledgements

The author wishes to acknowledge support from the UK e-science core programme through the gViz project. He also wishes to acknowledge the support and encouragement of partners in gViz.

7. References

- [1] AVS/Express - http://www.avs.com/software/soft_t/avsxps.html
- [2] DFDL: All papers and notes on the Data Format Description Language within GGF are available at <http://forge.gridforum.org/projects/dfd-1-wg>
- [3] The HDF5 Home Page, <http://hdf.ncsa.uiuc.edu/HDF5/>
- [4] Earth Science Markup Language (ESML) - <http://esml.itsc.uah.edu/index.jsp>
- [5] "The eXtensible Data Format", http://xml.gsfc.nasa.gov/XDF/XDF_home.html
- [6] Iris Explorer, http://www.nag.co.uk/Welcome_IEC.html
- [7] "Marine XML", <http://ioc.unesco.org/oceanteacher/resourcekit/Module2/Internet/XML/marine.html>
- [8] Matthews B.M., Sufi S., van Dam K.K. "CLRC Scientific Metadata Model", <http://epubs.cclrc.ac.uk/work-details?w=29024>
- [9] netCDF (Network Common Data Form), <http://my.unidata.ucar.edu/content/software/netcdf/index.html>
- [10] O'Neill K.D., "The Metadata Model of the NERC DataGrid", UK All Hands Meeting 2003, <http://www.e-science.clrc.ac.uk/documents/projects/nercdatagrid/AHM-2003-KON.pdf>
- [11] "OPeNDAP", <http://opendap.org/>
- [12] VisAD - <http://www.ssec.wisc.edu/~billh/visad.html>
- [13] "VOTable Definition Version 1.1", International Virtual Observatory Alliance 2004, <http://www.ivoa.net/Documents/latest/VOT.html>
- [14] "Open Grid Services Architecture Data Access and Integration (OGSA-DAI)", <http://www.ogsadai.org.uk/>