Accessing and Visualizing Scientific Spatiotemporal Data

Daniel S. Katz[1], Attila Bergou[1], G. Bruce Berriman[1], Gary L. Block[1], Jim Collier[1], David W. Curkendall[1], John Good[1], Laura Husman[1], Joseph C. Jacob[1], Anastasia Laity[1], P. Peggy Li[1], Craig Miller[1], Tom Prince[1], Herb Siegel[1], Roy Williams[3]

[1]Jet Propulsion Laboratory, California Institute of Technology, USA
[2]NASA Infrared Processing and Analysis Center, California Institute of Technology, USA
[3]Center for Advanced Computing Research, California Institute of Technology, USA
Daniel.S.Katz@jpl.nasa.gov

Abstract

This paper discusses work done by JPL's Parallel Applications Technologies Group in helping scientists access and visualize very large data sets through the use of multiple computing resources, such as parallel supercomputers, clusters, and grids. These tools do one or more of the following tasks: visualize local data sets for local users, visualize local data sets for remote users, and access and visualize remote data sets. The tools are used for various types of data, including remotely sensed image data, digital elevation models, astronomical surveys, etc. The paper attempts to pull some common elements out of these tools that may be useful for others who have to work with similarly large data sets.

1. Introduction

The Jet Propulsion Laboratory (JPL) is run by the California Institute of Technology for NASA, as NASA's lead center for robotic space exploration. Today, most of the data taken by JPL's spacecraft, as well as other NASA spacecraft, are sent to Earth for processing and analysis. As the spaceborne instruments improve, more and more data is taken and then returned to Earth.

Many common data sets are now each 5-10 TB, and the number of data sets is also increasing. Desktop tools often have difficulty working with data sets of this size. Simply storing the data and accessing it is difficult. Gaining knowledge from data is still hard, but one very effective method is to visualize the data.

This paper discusses work done by JPL's Parallel Applications Technologies (PAT) Group. The PAT Group's goal is to help JPL and Caltech scientists deal with large data sets. The remainder of this paper will discuss a number of tools that the PAT group has developed, for a few particular situations. The first is working with local data sets that the scientists themselves have generated. The second is providing access and information from data developed by the scientist to others located elsewhere. Finally, the third is accessing and working with data developed by other scientists. Unfortunately, the length of this paper will not permit many details to be provided on the actual tools.

2. Working with local data

Supercomputing is often defined by comparison with desktop systems, where the exact definition frequently changes. One example definition is computing that is 2 orders of magnitude more powerful than that available on a desktop. When data is generated on a supercomputer, it is often similarly larger in size than data that could be generated on a desktop. Not coincidentally, data sets of this size are also difficult to deal with in a desktop system. A common method for working with such data sets is to leave them on the supercomputer on which they were developed, and to return just the visual data to the user's desktop.

An example data set is synthetically-enhanced Martian terrain data. JPL has collected coarsely-sampled elevation data over the surface of Mars. When planning future missions, algorithms such as entry, descent, and landing (EDL) and surface navigation must be tested against simulated terrain. The PAT group has developed a terrain server to supply users with such terrain. We have parallelized an application previously developed at JPL [1], and created a system that allows users to launch jobs and return terrain data generated either by that job or previous jobs. The overall system is called TEDS, the Terrain and Environmental Data Server.

Two types of data are often used to represent terrain for a particular geographic area: a digital elevation model (DEM) that gives a height (z) for each location (x,y), and image data that includes one or more values for each pixel. When the pixels are mapped to locations, the two datasets are used together as a terrain model. One tool that has been developed to visualize terrain models is the Digital Light Table (DLT).

Proceedings of the 16th International Conference on Scientific and Statistical Database Management (SSDBM'04)
1099-3371/04 $ 20.00 © 2004 IEEE

The DLT was originally built to allow scientists to explore a SAR (synthetic aperture radar) mosaic of the Amazon basin [2]. This goal led to a number of requirements, including the ability to pan and zoom through the data in real time, along with the ability to scale to large input sizes, over 20 billion pixels, and to scale to large output sizes, over 7 million pixels.

The Remote Interactive Visualization and Analysis System (RIVA) [3, 4] is another tool that can visualize terrain data, but unlike the DLT/MSLT, it is designed with the underlying geometric model of a sphere, and thus can accommodate global (planetary) datasets.

There are several unique features that distinguish RIVA from other terrain renderers. The first is multiple data representation. Internally, RIVA represents the data using a spherical model regardless of whether it is a global dataset or a regular gridded dataset. Externally, RIVA can process data in either 2D Cartesian, 3D Cartesian, or 3D Polar coordinates. The dataset can be stored in either Sinusoidal projection to save space or in Cylindrical projection for efficient processing. RIVA is flexible and tries to accommodate different application needs and different data representations. The second feature is Multiple Surface Rendering. RIVA can render multiple terrain surfaces with different resolutions, different data formats, and different coverages, somewhat similarly to the MSLT. RIVA's multiple surfaces can be combined using various blending methods.

The third RIVA feature is Large Dataset Rendering. RIVA allows out-of-core rendering for datasets that exceed the capacity of the physical memory. A lower resolution sample of the original dataset has to be prepared in advance. RIVA loads the lower resolution dataset and renders it until the data pyramiding algorithm identifies that a higher resolution image tile is needed. The full resolution image tile will then be loaded into memory. A memory cache stores the most recent tiles used to reduce disk I/O. RIVA also renders time-varying datasets generated by simulations. Similar to out-of-core rendering, only the data for the first time step reside in the memory. The remaining data is loaded into memory when the animation starts.

The last RIVA feature is High Resolution Animation. RIVA is not only scalable to large input datasets but also scalable to large image outputs. RIVA images are not limited to the framebuffer size or the screen resolution as with other terrain renderers. RIVA can render a large image in multiple passes by partitioning the images into multiple viewports. Theoretically, there is no limit to the image size in RIVA.

3. Sharing Data Visually

A common problem at NASA is that of having a collection of data that has been taken, has possibly manipulated in some way, and is intended to be shared with others. At one time, the method for distributing such data was to put it on a tape and mail it to the end user. This has now generally been replaced by providing ftp or web access for large data sets, or distribution on CD or DVD for small small amounts of data. Our intent is to use the Internet for both small and large files, and specifically, to allow users to obtain just the data they desire. We can do this because we can take advantage of both falling disk prices to put all the data on disk for fast access, and we can use parallel computing that's hidden from the user to provide particular data products derived from the data on disk.

MapUS is an example of a project that demonstrates this idea. MapUS is a Landsat mosaic of the continental United States. This 6-band, 150-GB (compressed), 215,000 by 95,000 pixels image with 1-arc-second resolution was built from 428 individual Landsat patches of the Multi-Resolution Land Characteristics (MRLC) dataset, using custom-designed software based on SGI ImageVision framework and run on JPL supercomputers to accomplish this task as a single step operation, with minimal operator input. As has been discussed previously in this paper, we store the mosaic in a tiled, pyramided format. However, the tiles are not stored as separate files, but rather, a new, custom file-format was developed. [5]

We are also now near the end of development of OnEarth, a new server that will provide access to a mosaic of the land surface of the Earth at 1/2-arc-second resolution. The mosaic will be about 20 times larger than the MapUS mosaic. Past experience with storage systems together with budget as a consideration led to the selection of a Linux storage cluster. Other projects with similar storage requirements joined the development, with the result being a ten-system Linux storage cluster offering 40TB of storage space. The system, named Raid Again Storage using Commodity Hardware And Linux (RASCHAL), has been operational since April 2003.

The OnEarth WMS mosaic project is the first user of this storage system, to-date having restored more than 8500 of the original Landsat 7 scenes, and produced release 1 of the global mosaic. This storage system is being used for the mosaic building project and for hosting the data for the WMS site [6].

4. Accessing and Visualizing Remote Data

Another general problem is working with data stored in remote archives. Typical problems are understanding what data exists in the archive, what data exists in which file in the archive, and how these files are accessed. Another problem is building visualizations from this remote data. The PAT group has worked on a number of projects related to these problems, but in this paper, we will focus on astronomy applications. In these projects, we attempt to hide the complexity of accessing data from remote archives and building custom visualizations behind a simple interface, which is again that of a web browser.

The yourSky custom astronomical image mosaic server demonstrates this concept. By filling out and submitting the form at http://yourSky.jpl.nasa.gov/, users have custom access on their desktops to all the publicly released data from the member surveys. In this context, "custom access" refers to new technology that enables on-the-fly astronomical image mosaicking to meet user-specified criteria for: region of the sky to be mosaicked, data set to be used, resolution, coordinate system, projection, data type, and image format [7, 8]. All mosaic requests are fulfilled from the original archive data so that the domain experts maintain control and responsibility for their data and data corruption due to resampling is minimized because only one reprojection is done from the raw input data to the end product. Currently, the data archives that are accessible with yourSky are the Digitized Palomar Observatory Sky Survey (DPOSS) [9], that captured the entire northern sky at 1 arc second resolution in three visible wavelengths. and the 2nd incremental beta release of the Two Micron All Sky Survey (2MASS) [10], that captured the entire sky at 1 arc second resolution in three near-infrared wavelengths. The yourSky architecture supports expansion to include other surveys.

One issue with the current version of yourSky is that, while it operates quickly and thus is good for generating "browse" images, it does not preserve the calibration and astrometric fidelity of the data. Another project, Montage, builds on yourSky through the following improvements:

- Preservation of scientific fidelity in the mosaics
- Support for the "Drizzle" algorithm [11]
- Application of physically based background subtraction models
- Improved performance throughput
- Interoperability with grid infrastructure
- Compliance with NVO architecture

Montage consists of two independent but interoperable components: a background rectification engine, responsible for matching background radiation across images, and a coaddition/reprojection engine, responsible for computing the mosaic.

The Montage processing paradigm consists of three main parts: reprojection of images to a common scale and coordinate system; background adjustment of images to a common flux scale and background level; and coaddition of processed images into a final mosaic. The background adjustment process involves fitting the differences between overlapping images and determining the parameters for smooth surfaces to be subtracted from each image to bring them to the common scale. These parameters can either be determined on the fly or done once and saved in a database for any future mosaics done with the same images. We plan to use both approaches, deriving and storing at least one set of fit parameters for the full sky for each image collection, and allowing the user to invoke custom background processing if they think it will provide a better small-scale mosaic.

Currently, Montage has been released as a single processor application, built as a set of executables [12]. The time to run this code on our test system, a PC with a 1.4-Ghz IA32-processor running Linux, is about 100 seconds per 512x1024 pixel 2MASS image, where the image reprojection step takes up nearly all the run time. This step however, is inherently parallelizable, and can be run on however many processors are available to it. In addition, an order of magnitude speedup is possible with optimized algorithms for the most common projection transformations.

We have also demonstrated two prototypes of a parallel Montage. In one case, we scripted the process and data flow, assuming that all processors share a file system. This prototype, called Atlasmaker [13], obtained a speedup of approximately 60 on 64 nodes. The second prototype [14] is more general. It relies on a Montage-specific web server that builds an abstract DAG (directed acyclic graph, a definition of the data and processing dependencies), a general grid software package called Pegasus that builds a concrete DAG (a DAG that is tied to specific computational resources) from the abstract DAG, and another program called Condor DAGman [15] that then runs the concrete DAG on a collection of grid computers. This prototype is, however, currently much slower and only permits a small speedup.

All three versions of Montage share the use of NVO services to determine which input images need to be used and to access those images.

5. Conclusions

This paper has described a number of existing tools developed by the PAT group to make sense of large amounts of data: Digital Light Table (DLT), MSLT, RIVA, MAPUS, OnEarth, yourSky, and Montage. These tools all were developed in collaboration with scientists. They address three visualization modes: visualizing a scientist's data; allowing others to view images generated from a scientist's data; and accessing and visualizing remote data. Many of these applications try to hide the complexities of accessing remote data from archives and running jobs on supercomputers from the users. These applications encompass a set of technologies that can be used for new applications.

6. Acknowledgments

Most of the work discussed in this paper has been carried out at the Jet Propulsion Laboratory, California Institute of Technology with funding from NASA and the U.S. Air Force. NASA sponsors include the Earth Science Technology Office's (ESTO's) Computational Technologies (CT) project, the Digital Earth project, and the Geospatial Interoperability Office, and the Office of Space Science. The Montage project has been jointly carried out with NASA's Infrared Processing and Analysis Center (IPAC), located at Caltech.

7. References

[1] R. W. Gaskell, L. E. Husman, J. B. Collier, and R. L. Chen; "Synthetic Environments for Simulated Missions," *Proceedings of the IEEE Aerospace Conference*, 2001,

[2] P. Siqueira, S. Hensley, S. Shaffer, L. Hess, G. McGarragh, B. Chapman, and A. Freeman, "A Continental Scale Mosaic of the Amazon Basin Using JERS-1 SAR," *IEEE Trans. GeoSci. Rem. Sens.*, v. 38(6), pp. 2638-2644, 2000.

[3] P. P. Li, W. H. Duquette, and D. W. Curkendall, "RIVA: A Versatile Parallel Rendering System for Interactive Scientific Visualization," *IEEE Transactions on Visualization and Computer Graphics*, v. 2(3), pp. 186-201, 1996.

[4] P. P. Li, "Supercomputing Visualization for Earth Science Datasets", *Proceedings of the NASA Earth Science Technology Office (ESTO) Annual Conference*, 2002.

[5] L. Plesea and J. Jacob, "Building Large Scale Mosaics From Landsat Data," *Proceedings of the 8th ACM Symposium on Advances in Geographic Information Systems*, 2000.

[6] G. Percival and L. Plesea, "Web Map Services (WMS) Global Mosaic," *Proceedings of the 3rd International Syposium on Digital Earth: Information Resources for Global Sustainability*, 2003.

[7] J. C. Jacob, R. J. Brunner, D. Curkendall, S. G. Djorgovski, J. C. Good, L. Husman, G. Kremenek, and A. Mahabal, "yourSky: Rapid Desktop Access to Custom Astronomical Image Mosaics," *Proceedings of SPIE Astronomical Telescopes and Instrumentation: Virtual Observatories Conference*, 2002.

[8] The yourSky Custom Mosaic Server, http://yourSky.jpl.nasa.gov/.

[9] S. G. Djorgovski, R. R. Gal, S. C. Odewahn, R. R. de Carvalho, R. Brunner, G. Longo, and R. Scaramella, "The Palomar Digital Sky Survey (DPOSS)," To appear in: *Wide Field Surveys in Cosmology*, eds. S. Colombi and Y. Mellier.

[10] S. G. Kleinmann, "2MASS - The 2 Micron All Sky Survey," *Robotic Telescopes in the 1990s, (ASP Conf. Ser.)*, 203, 1992.

[11] A.S. Fruchter, and R.N. Hook. "Linear Reconstruction of the Hubble Deep Field," http://www.stsci.edu/~fruchter/dither/drizzle.html

[12] Montage web site: http://montage.ipac.caltech.edu/.

[13] R. Williams, L. Brieger, M. Feldman, J. Jacob, G. Kremenek, and R. Moore, "Atlasmaker: Grid-enabled Multiwavelength Imaging," *Proceedings of Astronomical Data Analysis Software & Systems*, 2003.

[14] G. Singh and E. Deelman, "Montage on the Grid," US Virtual Observatory Technical Report, http://bill.cacr.caltech.edu/cfdocs/usvo-pubs/list.cfm

[15] D. Thain, T. Tannenbaum, and M. Livny, "Condor and the Grid," *Grid Computing: Making The Global Infrastructure a Reality*, (F. Berman, A. J. G. Hey, G. Fox, editors,) John Wiley, 2003.