The Montage Architecture for Grid-Enabled Science Processing of Large, Distributed Datasets

Joseph C. Jacob, Daniel S. Katz, and Thomas Prince
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109-8099

G. Bruce Berriman, John C. Good, and Anastasia C. Laity
Infrared Processing and Analysis Center, California Institute of Technology
770 South Wilson Avenue, Pasadena, CA 91125

Ewa Deelman, Gurmeet Singh, and Mei-Hui Su
Information Sciences Institute, University of Southern California
4676 Admiralty Way, Suite 1001, Marina del Rey, CA 90292

Abstract – **Montage is an Earth Science Technology Office (ESTO) Computational Technologies (CT) Round III Grand Challenge project that will deploy a portable, compute-intensive, custom astronomical image mosaicking service for the National Virtual Observatory (NVO). Although Montage is developing a compute- and data-intensive service for the astronomy community, we are also helping to address a problem that spans both Earth and space science: how to efficiently access and process multi-terabyte, distributed datasets. In both communities, the datasets are massive, and are stored in distributed archives that are, in most cases, remote with respect to the available computational resources. Therefore, use of state-of-the-art computational grid technologies is a key element of the Montage portal architecture. This paper describes the aspects of the Montage design that are applicable to both the Earth and space science communities.**

I. Introduction

Montage is an effort to deploy a portable, compute-intensive, custom image mosaicking service for the astronomy community [1, 2, 3]. The Earth and space science communities each are faced with their own unique challenges, but they also share a number of technical requirements and can mutually benefit by tracking some of the information technology developments and lessons learned from both communities. Although Montage is developing a compute- and data-intensive service for the astronomy community, we are also helping to address a problem that spans both Earth and space science: how to efficiently access and process multi-terabyte, distributed datasets. Both communities have recognized the necessity of image re-projection and mosaicking as tools for visualizing medium- and large-scale phenomena and for enabling multi-wavelength science.

Like Earth science datasets, sky survey data are stored in distributed archives that are, in most cases, remote with respect to the available computational resources. Therefore, state-of-the-art computational grid technologies are a key element of the Montage portal architecture. The Montage project is contracted to deploy a science-grade custom mosaic service on the TeraGrid. TeraGrid is a distributed infrastructure, sponsored by the National Science Foundation (NSF,) that is expected to deliver 20 teraflops performance, with 1 petabyte of data storage, and 40 gigabits per second of network connectivity between the multiple sites. A second project at JPL also plans to use Montage to construct large-scale mosaics, in this case on the Information Power Grid (IPG,) NASA's computational grid infrastructure.

Astronomical images are almost universally stored in Flexible Image Transport System (FITS) format. The FITS format encapsulates the image data with a meta-data header containing keyword-value pairs that, at a minimum, describe the image dimensions and how the pixels map to the sky. Montage uses FITS for both the input and output data format. The World Coordinate System (WCS) specifies image coordinate to sky coordinate transformations for a number of different coordinate systems and projections useful in astronomy (some directly analogous to projections popular in the Earth science community).

Two Spitzer Space Telescope Legacy Program teams, GLIMPSE and SWIRE, are actively using Montage to generate science image products, and to support data simulation and quality assurance. Montage is designed to be applicable to a wide range of astronomical data, but is explicitly contracted to support mosaics constructed from images captured by three prominent sky surveys spanning multiple wavelengths, the Two Micron All Sky Survey (2MASS), the Digitized Palomar Observatory Sky Survey (DPOSS), and the Sloan Digital Sky Survey (SDSS). 2MASS has roughly 10 terabytes of images and catalogs, covering nearly the entire sky at 1-arc-second sampling in three near-infrared wavelengths. DPOSS has roughly 3 terabytes of images, covering nearly the entire northern sky in one near-infrared wavelength and two visible wavelengths. The SDSS second data release (DR2) contains roughly 6 terabytes of images and catalogs covering 3,324 square degrees of the Northern sky in five visible wavelengths.

This paper discusses the aspects of the Montage design that are applicable to both the Earth and space science communities. The remainder of the paper is organized as follows. Section II describes how Montage is designed as a modular toolkit. Section III describes techniques that are employed in Montage to dramatically expedite the calculation of mappings from one projection to another. We expect that these techniques could be beneficial for other mosaicking or re-projection applications in both Earth and space science. Section IV describes the architecture of the Montage TeraGrid portal. Performance on the TeraGrid is discussed in Section V. A summary and description of future plans is provided in Section VI.

II. Montage Modular Design

Montage has the broad goal of providing astronomers with software for the computation of custom science grade image mosaics in FITS format. Custom refers to user specification of the parameters describing the mosaic, including WCS projection, coordinate system, mosaic size, image rotation, and spatial sampling. Science grade mosaics preserve the calibration and astrometric (spatial) fidelity of the input images.

TABLE I
The Design Components Of Montage

Component	Description
Mosaic Engine Components	
mImgtbl	Extract geometry information from a set of FITS headers and create a metadata table from it.
mProject	Re-project a FITS image.
mProjExec	A simple executive that runs `mProject` for each image in an image metadata table.
mAdd	Co-add the re-projected images to produce an output mosaic.
Background Rectification Components	
mOverlaps	Analyze an image metadata table to determine which images overlap on the sky.
mDiff	Perform a simple image difference between a pair of overlapping images. This is meant for use on re-projected images where the pixels already line up exactly.
mDiffExec	Run `mDiff` on all the overlap pairs identified by `mOverlaps`.
mFitplane	Fit a plane (excluding outlier pixels) to an image. Meant for use on the difference images generated by `mDiff`.
mFitExec	Run `mFitplane` on all the `mOverlaps` pairs. Creates a table of image-to-image difference parameters.
mBgModel	Modeling/fitting program which uses the image-to-image difference parameter table to interactively determine a set of corrections to apply to each image to achieve a "best" global fit.
mBackground	Remove a background from a single image (a planar correction has proven to be adequate for the images we have dealt with).
mBgExec	Run `mBackground` on all the images in the metadata table

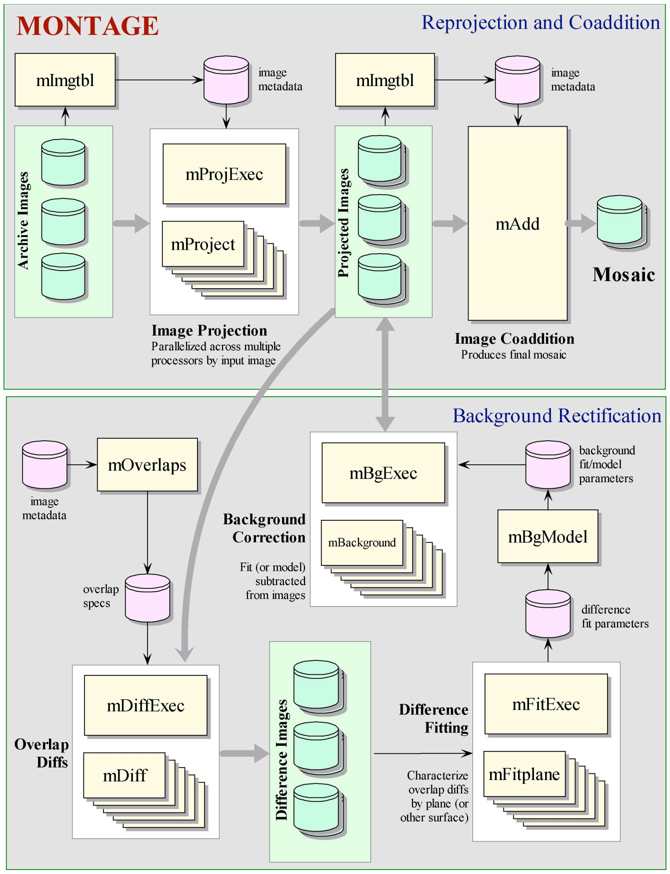


Fig. 1. The high-level design of Montage.

Montage constructs an image mosaic in four stages:

1. Re-projection of input images to a common spatial scale, coordinate system, and WCS projection,
2. Modeling of background radiation in images to achieve common flux scales and background levels by minimizing the inter-image differences,
3. Rectification of images to a common flux scale and background level, and
4. Co-addition of re-projected, background-matched images into a final mosaic.

Montage accomplishes these steps in independent modules, written in ANSI C for portability. This “toolkit” approach helps limit testing and maintenance costs, and provides considerable flexibility to users. They can, for example, use Montage simply to re-project sets of images and co-register them on the sky, implement a custom background matching algorithm without impact on the other steps, or define a specific processing flow through custom scripts. Table I gives a brief description of the main Montage modules and Fig. 1 illustrates how they may be used together to produce a mosaic.

Three usage scenarios for Montage are as follows: the modules listed in Table I may be run as stand-alone programs; the executive programs listed in the table (i.e., mProjExec, mDiffExec, mFitExec, and mBgExec) may be used to sequentially process multiple input images; or the grid portal mechanism described in Section IV may be used to process a mosaic in parallel on computational grids. The modular design of Montage permits the same set of core compute modules to be used regardless of the computational environment being used.

III. Techniques For Rapid Re-projection

As described in Section II, the first stage of mosaic construction is to re-project each input image to the spatial scale, coordinate system, and projection of the output mosaic. Traditionally, this is by far the most compute-intensive part of the processing because it is done in two steps; first, input image coordinates are mapped to sky coordinates (i.e., right ascension and declination, analogous to longitude and latitude on the Earth); and second, those sky coordinates are mapped to output image coordinates. All of the mappings from one projection to another are compute-intensive, but some require more costly trigonometric operations than others and a few require even more costly iterative algorithms. The first public release of Montage employed this two-step process to map the flux from input space to output space. Because the time required for this process stood as a serious obstacle to using Montage for large-scale image mosaics of the sky, a novel algorithm that is about 30 times faster was devised for the second code release.

The new approach uses an optimized "plane-to-plane" re-projection algorithm, modeled after a similar algorithm developed by the Spitzer Space Telescope project, for those projection mappings that can be computed without the intermediate step of mapping to the sky. The simplest of these is the mapping from one tangent plane projection to another. To generalize this to arbitrary input and output projections, we approximate the actual projection with a tangent plane projection with a polynomial warp. The fast plane-to-plane projection can then be done rapidly on these tangent plane approximations.

The error introduced by the Spitzer plane-to-plane re-projection is negligible on arbitrary spatial scales in the case where the transformation is between two tangent planes. For other projections, the tangent plane approximation introduces additional errors in astrometry, but early indications are that these errors can be kept below around 1% of a pixel width over a few degrees on the sky for most projections. Exceptions are the Aitoff and similar projections, where this approach is only applicable over a degree or two. The accuracy of this approach is well within acceptable tolerance levels and at a scale that is suitable for most scientific research. In situations where greater accuracy is necessary, the projection should be done using the intermediate step of mapping to the celestial sphere, as in the Montage first code release.

IV. Montage Grid Portal Architecture

The Montage TeraGrid portal has a distributed architecture, as illustrated in Fig. 2. The portal is comprised of the following five main components, each having a client and server: (i) User Portal, (ii) Abstract Workflow Service, (iii) 2MASS Image List Service, (iv) Grid Scheduling and Execution Service, and (v) User Notification Service. These components are described in more detail below.

A usage scenario is as follows. Users on the Internet submit mosaic requests by filling in a simple web form with parameters that describe the mosaic to be constructed, including an object name or location, mosaic size, coordinate system, projection, and spatial sampling. A service at JPL is contacted to generate an abstract workflow, which specifies: the processing jobs to be executed; input, output, and intermediate files to be read or written during the processing; and dependencies between the jobs. A 2MASS image list service at IPAC is contacted to generate a list of the 2MASS images required to fulfill the mosaic request. The abstract workflow is passed to a service at ISI, which runs software called Pegasus (Planning for Execution in Grids) [4, 5, 6]. Pegasus schedules the workflow on the TeraGrid (and possibly other resources), using grid information services to find information about data and software locations. The resulting "concrete workflow" includes information about specific file locations on the grid and specific grid computers to be used for the processing. The workflow is then executed on the remote TeraGrid clusters using Condor DAGMan [7]. The last step in the mosaic processing is to contact a user notification service at IPAC, which currently simply sends an email to the user containing the URL of the Montage output.

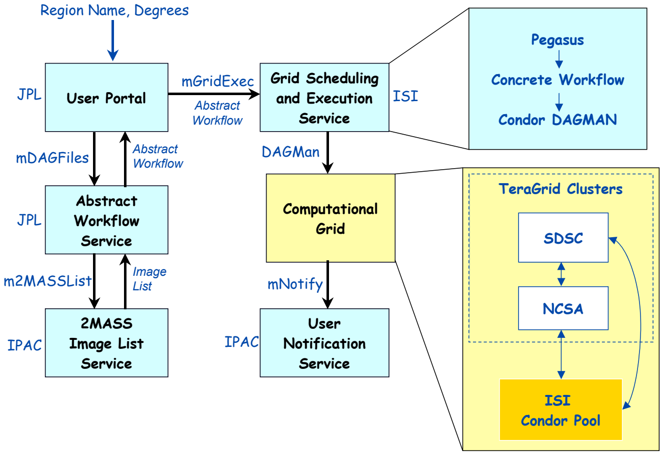
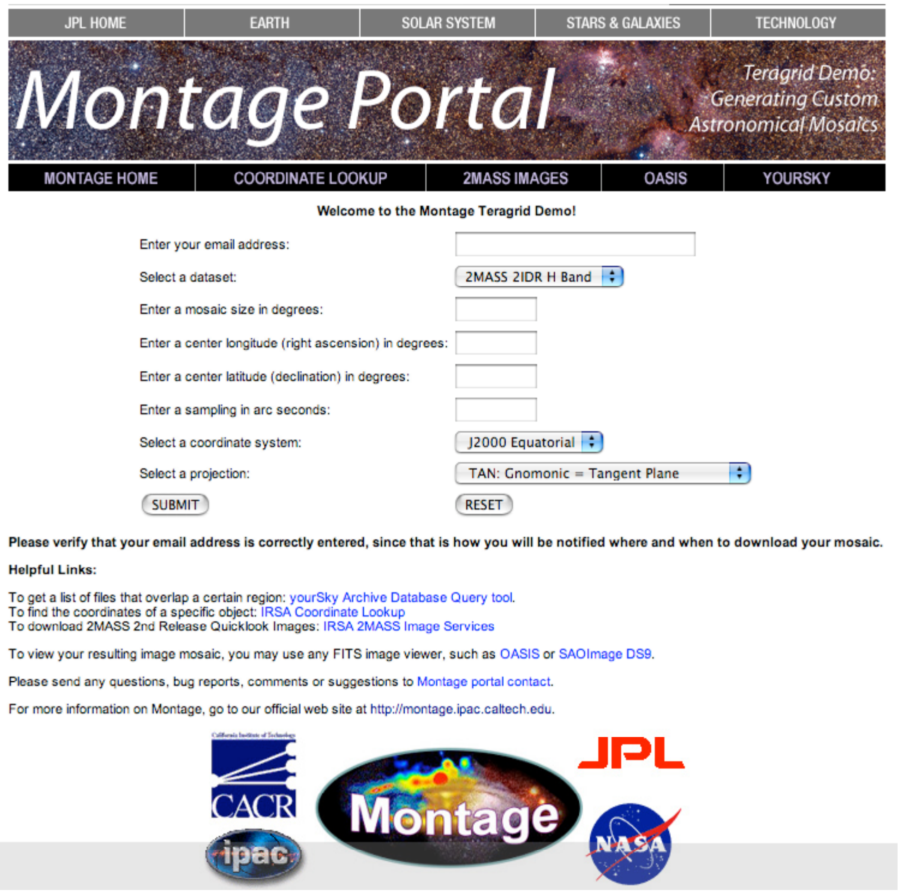


Fig. 2. The distributed architecture of the Montage TeraGrid Portal.



Welcome to the Montage Teragrid Demo!

Enter your email address:	
Select a dataset:	2MASS 2IDR H Band
Enter a mosaic size in degrees:	
Enter a center longitude (right ascension) in degrees:	
Enter a center latitude (declination) in degrees:	
Enter a sampling in arc seconds:	
Select a coordinate system:	J2000 Equatorial
Select a projection:	TAN: Gnomonic = Tangent Plane

Please verify that your email address is correctly entered, since that is how you will be notified where and when to download your mosaic.

Helpful Links:

To get a list of files that overlap a certain region: yourSky Archive Database Query tool.
To find the coordinates of a specific object: IRSA Coordinate Lookup
To download 2MASS 2nd Release Quicklook Images: IRSA 2MASS Image Services

To view your resulting image mosaic, you may use any FITS image viewer, such as OASIS or SAOImage DS9.

Please send any questions, bug reports, comments or suggestions to Montage portal contact.

For more information on Montage, go to our official web site at http://montage.ipac.caltech.edu.

Fig. 3. Montage grid portal web form interface.

This design exploits the parallelization inherent in the Montage architecture. The Montage grid portal is flexible enough to run a mosaic job on a number of different cluster and grid computing environments, including Condor pools and TeraGrid clusters. We have demonstrated processing on both a single cluster configuration and on multiple clusters at different sites having no shared disk storage.

A. *User Portal*

Users on the Internet submit mosaic requests by filling in a simple web form with parameters that describe the mosaic to be constructed, including an object name or location, mosaic size, coordinate system, projection, and spatial sampling. Fig. 3 shows a screen capture of the web form interface accessible at http://montage.jpl.nasa.gov/. After request submission, the remainder of the data access and mosaic processing is fully automated with no user intervention.

The server side of the user portal includes a CGI program that receives the user input via the web server, checks that all values are valid, and stores the validated requests to disk for later processing. A separate daemon program with no direct connection to the web server runs continuously to process incoming mosaic requests. The processing for a request is done in two main steps:

1. Call the abstract workflow service client code
2. Call the grid scheduling and execution service client code and pass to it the output from the abstract workflow service client code

B. *Abstract Workflow Service*

The abstract workflow service takes as input a celestial object name or location on the sky and a mosaic size and returns a zip archive file containing the abstract workflow as a directed acyclic graph (DAG) in XML and a number of input files needed at various stages of the Montage mosaic processing. The abstract workflow specifies the jobs and files to be encountered during the mosaic processing, and the dependencies between the jobs. These dependencies are used to determine which jobs can be run in parallel on multiprocessor systems. A pictorial representation of an abstract workflow for a mosaic with three input images is shown in Fig. 4.

C. *2MASS Image List Service*

The 2MASS Image List Service takes as input a celestial object name or location on the sky (which must be specified as a single argument string), and a mosaic size. The 2MASS images that intersect the specified location on the sky are

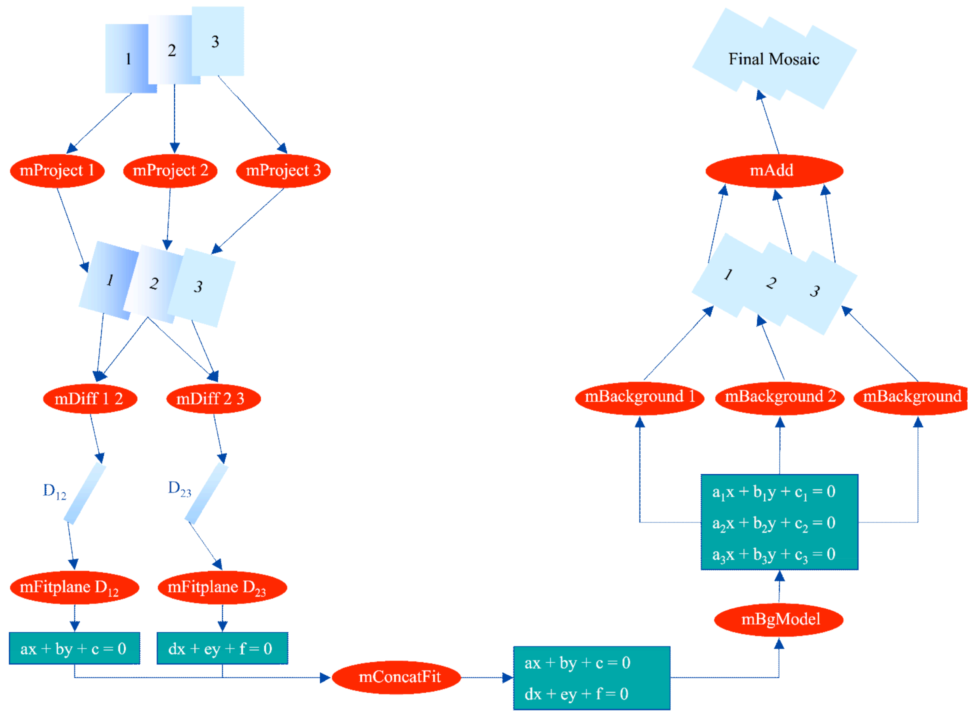


Fig. 4. Example abstract workflow.

returned in a table, with columns that include the filenames and other attributes associated with the images.

D. *Grid Scheduling And Execution Service*

The Grid Scheduling and Execution Service takes as input the zip archive generated by the Abstract Workflow Service, which contains the abstract workflow, and all of the input files needed to construct the mosaic. The service authenticates users, schedules the job on the grid using a program called Pegasus, and then executes the job using Condor DAGMan.

Users are authenticated on the TeraGrid using their Grid security credentials. The user first needs to save their proxy credential in the MyProxy server. MyProxy is a credential repository for the Grid that allows a trusted server (such as our Grid Scheduling and Execution Service) to access grid credentials on the users behalf. This allows these credentials to be retrieved by the portal using the user's username and password.

Once authentication is completed, Pegasus schedules the Montage workflow onto the TeraGrid or other clusters managed by PBS and Condor. Pegasus is a workflow management system designed to map abstract workflows onto the grid resources to produce concrete (executable) workflows. Pegasus consults various Grid information services, such as the Globus Monitoring and Discovery Service (MDS) [8], the Globus Replica Location Service (RLS) [9], the Metadata Catalog Service (MCS) [10], and the Transformation Catalog to determine what grid resources and data are available. If any of the data products described in the abstract workflow have already been computed and registered in the RLS, Pegasus removes the jobs that generate them from the workflow. In this way, the RLS can effectively be used in a data cache mechanism to prune the workflow. The executable workflow generated by Pegasus specifies the grid computers to be used, the data movement for staging data in and out of the computation, and the data products to be registered in the RLS and MCS, as illustrated in Fig. 5.

The executable workflow is submitted to Condor DAGMan for execution. DAGMan is a scheduler that submits jobs to Condor in an order specified by the concrete workflow. Condor queues the jobs for execution on the TeraGrid. Upon completion, the final mosaic is delivered to a user-specified location and the User Notification Service, described below, is contacted.

E. *User Notification Service*

The last step in the grid processing is to notify the user with the URL where the mosaic may be downloaded. This notification is done by a remote user notification service at IPAC so that a new notification mechanism can be used later without having to modify the Grid Scheduling and Execution Service. Currently the user notification is done with a simple email, but a later version will use the Request Object Management Environment (ROME), being developed separately for the National Virtual Observatory. ROME will

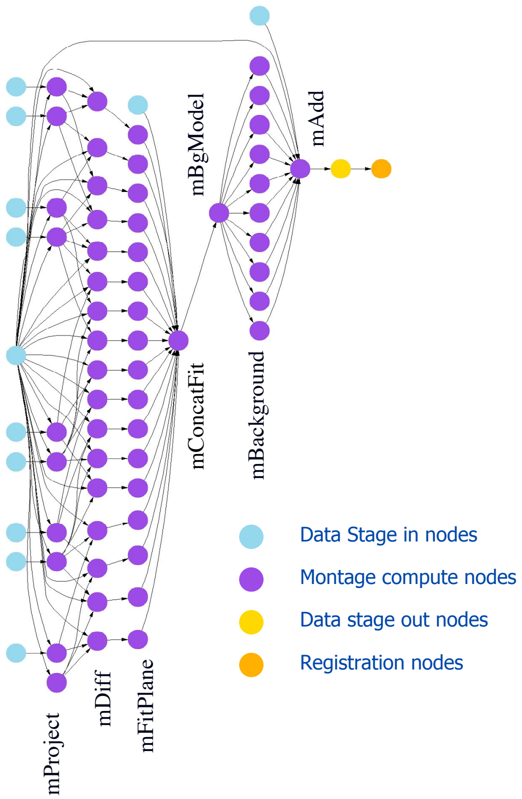


Fig. 5. Example concrete (executable) workflow for a 10 input file job on a single cluster. In addition to the computation nodes, the concrete workflow includes nodes for staging data into and out of the computation and for registering the data products for later retrieval..

extend our portal with more sophisticated job monitoring, query, and notification capabilities.

V. Performance

We have run the Pegasus-enabled Montage on a variety of resources: Condor pools, LSF- and PBS-managed clusters, and the TeraGrid (through PBS). Table II gives the runtimes of the individual workflow components to summarize the results of running a 2-degree M16 mosaic on the NCSA TeraGrid cluster. These performance figures are for the first release of Montage, which does not include the algorithmic optimizations described in Section III. The total runtime of the workflow was 107 minutes and the workflow contained 1,515 individual jobs.

To this point, our main goal was to demonstrate feasibility of running the Montage workflow in an automated fashion on the TeraGrid with some amount of performance improvement over the sequential version. Currently, Pegasus schedules the workflow as a set of small jobs. As seen in the table, some of these jobs run only a few seconds, which is suboptimal because scheduling too many little jobs suffers from large overheads. In fact, if this processing was run on a single TeraGrid processor, it would have taken 445 minutes, so we are not taking very much advantage of the TeraGrid's parallelism. However, initially structuring the workflow in this way allows us to expose the highest degree of parallelism.

We will improve this performance by optimizing both the Montage algorithms and the grid scheduling techniques. We expect about a 30 times speedup without sacrificing accuracy by using the algorithmic techniques described in Section III. We will address TeraGrid performance in three ways: making Pegasus aggregate nodes in the workflow in a way that would reduce the overheads for given target systems; encouraging the Condor developers to reduce the per-job overhead; and examining alternate methods for distributing the work on the grid. Each option has advantages and disadvantages that will be weighed as we go forward.

VI. Conclusion

Montage is a project to design and develop high science quality astronomical image mosaicking software. The software will be made accessible to the science community using two mechanisms: (i) a toolkit that can be directly downloaded and run manually on a local computer, and (ii) a fully automated grid portal with a simple web-form interface. A number of characteristics of the Montage design are applicable to both the Earth and space science communities, including fast image re-projection techniques and grid portal mechanisms. Montage incorporates a tangent plane approximation and fast plane-to-plane mapping technique to optimize the compute-intensive re-projection calculations.

A Montage mosaic job can be described in terms of an abstract workflow so that a planning tool such as Pegasus can derive an executable workflow that can be run in a grid environment. The execution of the workflow is performed by the workflow manager DAGMan and the associated Condor-G. This design exploits the parallelization inherent in the Montage architecture. The Montage grid portal is flexible enough to run a mosaic job on a number of different cluster and grid computing environments, including Condor

TABLE II
TeraGrid Performance Of Montage

Number of Jobs	**Job Name**	**Average Run-Time**
1	mAdd	94.00 seconds
180	mBackground	2.64 seconds
1	mBgModel	11 seconds
1	mConcatFit	9 seconds
482	mDiff	2.89 seconds
483	mFitplane	2.55 seconds
180	mProject	130.52 seconds
183	Transfer of data in	Between 5-30 seconds each
1	Transfer of mosaic out	18: 03 minutes

pools and TeraGrid clusters. We have demonstrated processing on both a single cluster configuration and on multiple clusters at different sites having no shared disk storage.

Our current and future work includes optimizing the grid scheduling to better account for *a priori* knowledge about the size of the computation required for different parts of the Montage processing. This information will be used to aggregate appropriate parts of the computation in order to lessen the impact of the overhead of scheduling smaller chunks of computation on grid computers. Also, the portal will be integrated with ROME for improved job monitoring, query, and notification capabilities.

Acknowledgments

Montage is funded by NASA's Earth Science Technology Office, Computational Technologies Project, under Cooperative Agreement Number NCC5-626 between NASA and the California Institute of Technology. Pegasus is supported by NSF under grants ITR-0086044 (GriPhyN) and ITR AST0122449 (NVO).

Part of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

References

1. B. Berriman, D. Curkendall, J. C. Good, L. Husman, J. C. Jacob, J. M. Mazzarella, R. Moore, T. A. Prince, and R. E. Williams, *Architecture for Access to Compute Intensive Image Mosaic and Cross-Identification Services in the NVO*, SPIE Astronomical Telescopes and Instrumentation: Virtual Observatories Conference, August 2002.
2. G. B. Berriman, D. Curkendall, J. Good, J. Jacob, D. S. Katz, T. Prince, R. Williams, *Montage: An On-Demand Image Mosaic Service for the NVO*, Astronomical Data Analysis Software and Systems (ADASS) XII, October 2002, Astronomical Society of the Pacific Conference Series, eds. H. Payne, R. Jedrzejewski, and R. Hook.
3. B. Berriman, A. Bergou, E. Deelman, J. Good, J. Jacob, D. Katz, C. Kesselman, A. Laity, G. Singh, M.-H. Su, and R. Williams, *Montage: A Grid-Enabled Image Mosaic Service for the NVO*, Astronomical Data Analysis Software & Systems (ADASS) XIII, October 2003.
4. E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, S. Patil, M.-H. Su, K. Vahi, M. Livny, *Pegasus: Mapping Scientific Workflows onto the Grid*, Across Grids Conference 2004, Nicosia, Cyprus.
5. Y. Gil, E. Deelman, J. Blythe, C. Kesselman, and H. Tangmurarunkit, *Artificial Intelligence and Grids: Workflow Planning and Beyond*, IEEE Intelligent Systems, January 2004.
6. E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, K. Blackburn, A. Lazzarini, A. Arbree, R. Cavanaugh, and S. Koranda, *Mapping Abstract Complex Workflows onto Grid Environments*, Journal of Grid Computing, vol. 1, no. 1, 2003, pp. 25-39.
7. Condor and DAGMan, http://www.cs.wisc.edu/condor/.
8. K. Czajkowski, et al., *Grid Information Services for Distributed Resource Sharing*, Proceedings of 10th IEEE International Symposium on High Performance Distributed Computing, 2001.
9. A. Chervenak, et al., *Giggle: A Framework for Constructing Scalable Replica Location Services*, Proceedings of Supercomputing (SC) 2002.
10. G. Singh, et al., *A Metadata Catalog Service for Data Intensive Applications*, Proceedings of Supercomputing (SC) 2003.