

Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets

Year 1 Progress Report & Year 2 Proposal

1 Year 1 Proposal

In order to setup the context for this progress report, this section covers a brief motivation for our work and summarizes the Year 1 Proposal we originally submitted under grant number NNA06CB89H.

Large datasets are being produced at a very fast pace in the astronomy domain. In principle, these datasets are most valuable if and only if they are made available to the entire community, which may have tens to thousands of members. The astronomy community will generally want to perform various analyses on these datasets to be able to extract new science and knowledge that will both justify the investment in the original acquisition of the datasets as well as provide a building block for other scientists and communities to build upon to further the general quest for knowledge.

Grid Computing has emerged as an important new field focusing on large-scale resource sharing and high-performance orientation. The Globus Toolkit, the “de facto standard” in Grid Computing, offers us much of the needed middleware infrastructure that is required to realize large scale distributed systems. We proposed to develop a collection of Web Services-based systems that use grid computing to federate large computing and storage resources for dynamic analysis of large datasets. We proposed to build a Globus Toolkit 4 based prototype named the “AstroPortal” that would support the “stacking” analysis on the Sloan Digital Sky Survey (SDSS). The stacking analysis is the summing of multiple regions of the sky, a function that can help both identify variable sources and detect faint objects. We proposed to deploy the AstroPortal on the TeraGrid distributed infrastructure and apply the stacking function to the SDSS DR4 dataset, which comprises more than 300 million objects dispersed over 1.2 million files, a total of 8 terabytes of data.

We claimed that our work with the AstroPortal would lead to interesting and innovative research work in three main areas: 1. *resource provisioning* (advanced resource reservations, resource allocation, resource de-allocation, and resource migration); 2. *data management* (data location, data replication, and data caching); and 3. *distributed resource management* (coupling data and processing resources together efficiently, and distributed resource management for scalability).

The rest of this document has the following outline. Section 2 covers the progress we have made since the original proposal submission, including the completed milestones, short term goals, and how we have disseminated our results to date. Section 3 discusses the Year 2 Proposal that we plan to complete between 10/2007 and 9/2008. Finally, Section 4 covers the contributions our work will make with the successful completion of the work outlined in this document, as well as a brief conclusion.

2 Year 1 Progress Report

We have made significant progress since our initial proposal. This section will first discuss the completed milestones, followed by the following short-term goals, the deliverables we expect to produce, and the dissemination of our results.

2.1 Completed Milestones

We initially proposed to build the AstroPortal, which would implicitly involve interesting and innovative research work in three main areas: 1) *resource provisioning*, 2) *data management*, 3) *distributed resource management*.

At this point we have developed a Web Services-based system, AstroPortal, that uses grid computing to federate large computing and storage resources for dynamic analysis of large datasets. We have deployed the AstroPortal on the TeraGrid distributed infrastructure and is now online in beta testing by our collaborator’s group Alex Szalay at John Hopkins University.

As for the three main areas that we claimed to address in our work, we have implemented four basic building blocks to address them. 3DcacheGrid, Dynamic Distributed Data cache for Grid applications addresses the *data*

management. CompuStore, a Data Aware Task Scheduler, addresses the *distributed resource management*. DRP, Dynamic Resource Provisioning, addresses *resource provisioning*. Finally, DeeF, Distributed execution environment Framework, is used to integrate all these three basic components into a unified execution environment that can be used to facilitate the ease of implementation of applications.

2.2 Short Term Goals

Performance Evaluation: With the implementation of the AstroPortal and the basic building blocks completed, we will now focus on the performance evaluation (efficiency, effectiveness, scalability, and flexibility) of our implementations. Although we have some preliminary performance results, most of the results are from a relatively basic implementation of the AstroPortal prior to the completion of the various optimizations such as caching, data aware scheduling, and dynamic resource provisioning.

New Science: We expect to work with the astronomy community at large to get the AstroPortal into production so it can be used to advance the astronomy domain as we have outlined in our initial proposal. Our contacts with the astronomy community are 1) Alex Szalay from the Department of Physics and Astronomy at Johns Hopkins University, 2) Jerry C. Yan from the NASA Ames Research Center, and 3) the US National Virtual Observatory (NVO) at <http://sandbox.us-vo.org/grid.cfm>.

2.3 Deliverables and Dissemination of the Results

We expect to have the following deliverables upon the completion (09/30/2007) of the NASA GSRP Fellowship under Grant Number NNA06CB89H. As a result of the work funded by the NASA GSRP Fellowship, we have produced a series of software systems, papers, documents, presentations, an online portal and documentation, which can all be accessible online at the main AstroPortal web site at <http://people.cs.uchicago.edu/~iraicu/research/AstroPortal/index.htm>. Deliverables that have not been completed yet will also be posted on this web site when they are completed. We clearly denote each item below with one of three categories, depending on what state the particular item in question is in: 1) COMPLETED, 2) IN PROGRESS, and 3) TO DO.

- **Implementations:**

- Basic Building Blocks:
 - *3DcacheGrid: Dynamic Distributed Data cache for Grid applications* (**COMPLETED**)
 - *CompuStore: a Data Aware Task Scheduler* (**COMPLETED**)
 - *DRP: Dynamic Resource Provisioning* (**COMPLETED**)
 - *DeeF: Distributed execution environment Framework* (**COMPLETED**)
- Application: “*AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis*”; the web portal to access and use the AstroPortal can be found at <http://s8.uchicago.edu:8080/AstroPortal/index.jsp>. (**COMPLETED**)

- **Papers and Reports.**

- Ioan Raicu, Ian Foster, Alex Szalay. “*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets*”, IEEE/ACM SuperComputing 2006. (**COMPLETED**)
- Ioan Raicu, Ian Foster, Alex Szalay. “*AstroPortal*”, IEEE/ACM SuperComputing 2006, Argonne National Laboratory Booth, Project Summary Handout, November 2006. (**COMPLETED**)
- Ioan Raicu, Ian Foster, Alex Szalay, Gabriela Turcu. “*AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis*”, TeraGrid Conference 2006, June 2006, Indianapolis, USA. (**COMPLETED**)
- Alex Szalay, Julian Bunn, Jim Gray, Ian Foster, Ioan Raicu. “*The Importance of Data Locality in Distributed Computing Applications*”, NSF Workflow Workshop 2006. (**COMPLETED**)
- Ioan Raicu, Ian Foster, Alex Szalay. “*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets*”, Technical Report, Department of Computer Science, University of Chicago, May 2006. (**COMPLETED**)
- Several more papers and reports outlining design, implementation, and performance of the basic building blocks and the AstroPortal system, as well as new science from the astronomy field that was achievable due to our work. (**IN PROGRESS** and **TO DO**)

- **Documents for NASA:**
 - **Initial Year 1 Proposal:** Ioan Raicu, Ian Foster. "*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets*", NASA GSRP Proposal, Ames Research Center, NASA, February 2006 -- Award funded 10/1/06 - 9/30/07. **(COMPLETED)**
 - **6 Month Progress Report:** Ioan Raicu, Ian Foster. "*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets, Year 1 Progress Report & Year 2 Proposal*", NASA GSRP Proposal, Ames Research Center, NASA, February 2007. **(COMPLETED)**
 - **Final Report: (TO DO)**
- **Presentations:**
 - "*AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis*", **IEEE/ACM SuperComputing 2006**, November 2006. **(COMPLETED)**
 - "*Storage and Compute Resource Management via DYRE, 3DcacheGrid, and CompuStore*", **University of Chicago, Department of Computer Science, Distributed Systems Lab Seminar**, November 2006. **(COMPLETED)**
 - "*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets*", **DSL Workshop 2006**, June 2006. **(COMPLETED)**
 - "*AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis*", **TeraGrid Conference 2006**, June 2006. **(COMPLETED)**
 - "*Harnessing Grid Resources to Enable the Dynamic Analysis of Large Astronomy Datasets*", **University of Chicago, Department of Computer Science, Graduate Seminar**, February 2006. **(COMPLETED)**
 - "*AstroPortal: A Science Portal to Grid Resources*", **University of Chicago, Department of Computer Science, Distributed Systems Lab Seminar**, January 2006. **(COMPLETED)**
- **Online:** Web site making all our work (papers, presentations, source code, etc) publicly available. **(IN PROGRESS)**

We still have some work completed that we have not had the chance to publish yet; furthermore, we have several more milestones (i.e. performance evaluation, new science) to complete which could also be published. We expect to target some of the best conferences and journals in the Grid Computing field, such as Journal of Grid Computing, IEEE/ACM SuperComputing (SC), ACM High Performance Distributed Computing (HPDC), IEEE/ACM Conference on Grid Computing (GRID), etc. Furthermore, I expect to visit Jerry C. Yan at the NASA Ames Research Center to present my findings in person, demo the AstroPortal, and find some synergy between my work and that being done at ARC.

3 Year 2 Proposal

As a continuation to our initial proposal, we would like to generalize our work from the AstroPortal even further beyond just the implementation of the basic components (3DcacheGrid, CompuStore, DRP, and DeeF). Although these basic building blocks should allow the implementation of many applications to be built with relatively little effort, we believe it would be valuable to define an abstract model that formally defines each basic component and its interaction with other components. This abstract model should allow us to explore the general problem space much more freely as we will break free of any application specific implementation or feature which might have influenced us when we implemented the basic building blocks and the AstroPortal.

The key observation we make is that as processing cycles become cheaper and data sets double in size every year, the main challenge for a rapid turnaround in the analysis of large datasets is the location of the data relative to the available computational resources; even with high capacity network interconnects, moving the data repeatedly to distant computational resources is becoming the bottleneck. There are large differences in data access speeds among the hierarchical storage systems normally found today in large distributed systems. Furthermore, data analysis workloads can be time varying in both their complexity and frequency, making both the computational and storage resource demands vary frequently.

Abstract Model: The analysis of large datasets normally follows a split/merge methodology, which includes an analysis query to be answered, which gets split down into independent tasks to be computed, after which the results from all the tasks are merged back into a single aggregated result. The hypothesis is that significant performance improvements can be obtained in the analysis of large dataset by leveraging information about data analysis workloads rather than individual data analysis tasks. We define workloads to be a complex query that can be decomposed into simpler tasks, or a set of queries that together answer some broader analysis questions. We believe it is feasible to allocate compute resources and caching storage resource that are relatively remote from the original data location, co-scheduled together to optimize the performance of entire data analysis workloads. Based on the split/merge methodology, we propose AMDASK, an Abstract Model for DATA-centric taSK farms, which defines the abstract model that allows us to study the stated hypothesis. Traditionally, task farms have been defined as a common parallel pattern which drives the computation of independent tasks, where a task is a self contained computation. The data-centric component of the abstract model emphasizes the central role data plays in the task farm model we are proposing, and the fact that the task farm is optimized to take advantage of data cache storage and data locality found in many large datasets and typical application workloads. Together, a data-centric task farm is defined as a common parallel pattern which drives the independent computational tasks taking into consideration the data locality in order to optimize the performance of the analysis of large datasets. This definition implies the integration of data semantics and application behavior in order to address critical challenges in the management of large scale datasets and the efficient execution of application workloads.

We intend to validate the AMDASK model through simulations. We expect the discrete event simulations to show the AMDASK model is both efficient and scalable given a wide range of simulation parameters. Once the model is validated, we will show that the current set of basic building blocks and AstroPortal application fits the model, as well as possibly implementing other applications on top of AMDASK in order to show the model's efficiency, effectiveness, scalability, and flexibility in practice.

Simulations: We will implement the AMDASK model in a discrete event simulation that will allow us to investigate a wider parameter space than we could in a real world implementation and deployment. We expect the simulations to both validate the AMDASK model and help us prove that the model is efficient and scalable given a wide range of simulation parameters (i.e. number of storage and computational resources, communication costs, management overhead, and workloads – including inter-arrival rates, query complexity, and data locality).

The simulations will specifically attempt to model a grid computing environment comprising of computational resources, storage resources, batch schedulers, various communication technologies, various types of applications, and workload models. We will perform careful and extensive empirical performance evaluations in order to create accurate input models to the simulator; the input models include 1) communication costs, 2) data management costs, 3) task scheduling costs, 4) storage access costs, and 5) workload models.

We expect to be able to scale simulations to more computational and storage resources than we could achieve in a real deployed system due to the availability of resources. Furthermore, assuming the input models to be correct, we should be able to accurately measure the end-to-end performance of various applications using a wide range of strategies for the various resource management components.

Applications: After showing that the defined basic building blocks (3DcacheGrid, CompuStore, DRP, and DeeF) and the AstroPortal fit the general abstract model, we intend to further pursue the identification and implementation of other applications to use the basic components based on the AMDASK model in order to prove both the effectiveness and the flexibility of the abstract model in practice. For each particular application, we also expect to quantify the efficiency and expected scalability based on the dataset sizes and typical workloads.

We have identified two such applications. The first is application is very similar to the “stacking” analysis and uses the same SDSS image dataset. This application is named “montage”, which performs the stitching of many images in a contiguous portion of the sky to produce a single unified image. Another application we identified is from the astro-physics domain which would utilize simulation data (as opposed to image data) from the Flash dataset. The applications that we have identified to fit the AMDASK model are volume rendering and vector visualization. The dataset is composed of 32 million files (1000 time steps times 32K files) taking up about 15TB of storage resources. The dataset contains both temporal and spatial locality, which should offer ample optimization opportunities in the data management component. More information can be found on the ASC / Alliances Center for Astrophysical Thermonuclear Flashes at their website at <http://www.flash.uchicago.edu/website/home/>.

4 Contributions & Conclusions

We see the dynamic analysis of large datasets to be a very important due to the ever growing datasets that need to be accessed by larger and larger communities. Attempting to address the storage and computational problems separately essentially forcing much data movement between computational resources will not scale to tomorrow's peta-scale datasets and will likely yield significant underutilization of the raw resources. We defined the abstract model for data-centric task farms, AMDASK, in order to address the integration of the storage and computational issues found in a class of applications which can be decomposed down into independent computational tasks which need to work on large datasets. We validated the abstract model by both simulations and by implementing the basic components needed to handle the data resource management, compute resource management, scheduling, and remote execution environment; we measured both the efficiency and scalability of the abstract model and our implementation. We then explored various applications from the astronomy and astro-physics domain that allowed us to use the AMDASK model and the implemented basic components to show off the flexibility and effectiveness of the abstract model on real world applications.

There are various fundamental research questions and directions we hope to address through our work presented in this proposal. They center on two main areas, data and compute resource management, and how they relate to particular workloads of data analysis of large datasets.

For the data resource management, we believe that data management architectures is very important to ensure that the data management implementations scale to the required dataset sizes in the number of files, objects, and dataset disk space usage while at the same time, ensuring that data element information can be retrieved fast and efficiently. Another important topic is replication strategies in order to meet a desired QoS and data availability. Finally, the data placement and caching strategies must be investigated to identify their appropriateness for workloads, datasets, data locality, and access patterns typically found in the interactive analysis of large datasets. We believe that we will address these data resource management fundamental research questions with the 3DcacheGrid engine and the simulations of the AMDASK model.

In the realm of compute resource management, there are several important issues. Dynamic resource provisioning architectures and implementations must be carefully designed in order to offer the right abstraction while at the same time offer practical and measurable advantages over static resource provisioning. Another important issue is the scheduling of computational tasks close to the data. Essentially, we need to investigate various strategies for workload management in which we can quantify the cost of moving the work vs. moving the data. We believe we will address these compute resource management research questions with DRP, CompuStore, and the simulations.

Furthermore, beside the contributions we have made to the Grid Computing domain, we believe the AstroPortal system to offer contributions to the Astronomy domain as well. The AstroPortal prototype supports the "stacking" operation, the summing of image cutouts from different parts of the sky. This function can help to statistically detect objects to faint otherwise. Astronomical image collections usually cover an area of sky several times (in different wavebands, different times, etc). On the other hand, there are large differences in the sensitivities of different observations: objects detected in one band are often too faint to be seen in another survey. In such cases we still would like to see whether these objects can be detected, even in a statistical fashion. There has been a growing interest to re-project each image to a common set of pixel planes, then stacking images. The stacking improves the signal to noise, and after co-adding a large number of images, there will be a detectable signal to measure the average brightness/shape etc of these objects. While this has been done for years manually for a small number of pointing fields, performing this task on wide areas of sky in a systematic way has not yet been done. It is also expected that the detection of much fainter sources (e.g., unusual objects such as transients) can be obtained from stacked images than can be detected in any individual image. The AstroPortal gives the astronomy community a new tool to advance their research and opens doors to new opportunities on a much larger scale than ever was possible before!