PROCEEDINGS OF SCIENCE

# Large-Scale Data Management and Analysis for Astronomical Research

**Cheng-Hsien Tang***, **Min-Feng Wang, Wei-Jen Wang, Meng-Feng Tsai**
*Department of Computer Science and Information Engineering, National Central University, Taiwan*

**Yuji Urata, Chow-Choong Ngeow, Induk Lee**

*Institute of Astronomy, National Central University, Taiwan*

**Kuiyun Huang**

*Academia Sinica Institute of Astronomy and Astrophysics, Taiwan*

The improvement of information technology enables precise scientific observation that demands larger storage and faster data processing techniques than ever before. From the perspective of astronomical research, one of the most important challenges is to extract useful astronomical information efficiently from a huge collection of observed data. Even though the existing distributed computing techniques, such as grid computing and cloud computing, have provided the scientists a better way to access powerful computing resources, the development of big-data management and analysis software is still lagging far behind. The awkward predicament obstructs the connected computing resources from being utilized efficiently. Therefore, it is beneficial to provide an integrated, efficient information management and analysis system for astronomical research.

This research, conducted by the Pan-STARRS research team at Taiwan, focuses on the issues of integrating commercial data warehouse and large-scale grid computing techniques, and develops a system for efficient data management and fast analysis in astronomy-related fields. Our system can be viewed as a data grid system that supports analysis of large data collections. The system consists of two analytical sub-systems and one data presentation and management sub-system. The first one is called the PARallel Hierarchical Agglomerative Clustering System (PARHACS), which uses a distributed message-passing algorithm to efficiently calculate a hierarchical cluster, given a set of astronomical data. The second sub-system is called the SIMilarity Classification System (SIMCS), which uses a decentralized Multiple Classifier System (MCS) framework to support a complex classification procedure using multiple classifiers. The last sub-system is called the ASTROnomical Information Management System (ASTROIMS), which utilizes a multidimensional data-warehouse design to construct a more concise, integrated, and scalable platform for fast data retrieval and management. It is able to perform data maintenance procedures automatically and to reduce maintenance and operation costs easily. In addition, the sub-system provides a user-friendly interface to facilitate a variety of data analytical tasks on line.

*The International Symposium on Grids and Clouds and the Open Grid Forum*
*March 19 - 25, 2011*
*Academia Sinica, Taipei, Taiwan*

*Speaker.

 

PoS(ISGC 2011 & OGF 31)044

## 1. Introduction

International astronomical observation projects such as Pan-STARRS [1] and TAOS [2] have been continuously collecting huge amount of data. Even though various sky-image processing and calibration software have processed the raw image files and produced relatively smaller datasets, the primary database still accumulates huge amount of data with complicated formats that keeps growing in a rapid speed. The huge amount of data has become a critical challenge for many analytical processes in various astronomical applications. Astronomical researchers now face an undesired situation that they may have to spend most of their time learning how to retrieve portions of their research data, repeatedly and regularly. Therefore, it is important to design an information system that can manage the corresponding astronomical data, and provide facilities to support analysis tasks for the astronomical researchers.

To provide better support for efficient data analysis over huge astronomical datasets, we have implemented a system that provides computational power and storage capacity. Our system aims to provide users more advanced analytical functions based on machine learning and clustering techniques. The proposed system consists of three core subsystems that handle different problems. One subsystem provides an integrated and consolidated view of the entire datasets to enable efficient data delivery and retrieval. Two other subsystems perform efficient distributed classification and parallel clustering so that researchers with different research interests can focus on different portions of data based on the criteria of their interests. The prototype of our system is able to benefit astronomical researchers in many data process applications such as research target classification, verification of existed theoretical model, and efficient research data management. We believe this approach can contribute on more aspects of analytical jobs with continuous collaborations with astronomical researchers.

The design of our information management subsystem provides integrated, consolidated views of data storage as well as versatile data access capabilities. It is designed for the researchers and other modules to facilitate research on different interests such as the solar system, variable stars, or galaxy observations. The proposed system obtains numerical data from existing large data collection such as the Pan-STARRS project [1], the International Virtual Observatory Alliance (IVOA) [3] and the Sloan Digital Sky Survey [4]. It adopts the multidimensional data-warehouse concept, and incorporates possible analytical angles to index the underlying dataset. So analysis can be conducted both on more generalized scopes and also on specialized area. The other two core subsystems provide classifications and clustering capabilities for many astronomical analysis tasks. Appropriate classification or clustering allows the researchers to spend their energy on the small portions of interested groups, or to prepare the data for different analysis tasks based on their observed features. This is often necessary when dealing with the datasets from international astronomical projects. Meaningful analytical processes are simply infeasible if they are required to operate on the entire set of huge data collections that are still increasing. On the other hand, the classification and clustering processes themselves may also take very long time, which either is unacceptable or becomes the bottlenecks in the entire analysis process. The design of our two analytical core subsystems is able to adapt to the distributed or parallel computing schemes that can utilize great amount of computation resources for timely results. As a result, they can utilize the computing power from the emerging techniques of cloud or grid systems.

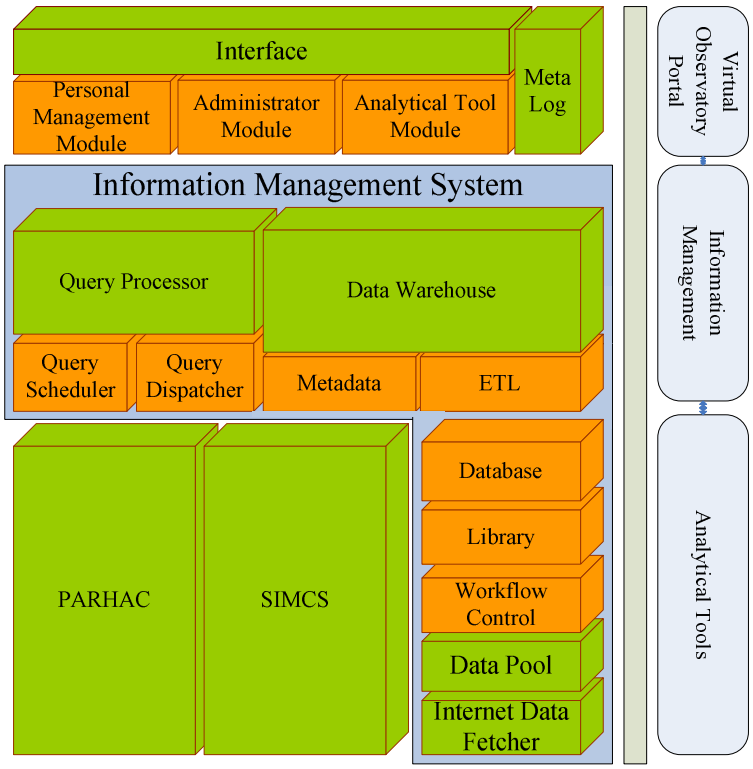


**Figure 1:** The abstraction of the proposed system and its sub-systems.

This paper is organized as follows. In Section 2, we briefly present the user interface and the information management subsystem, namely the ASTROnomical Information Management System (ASTROIMS). In Section 3, we describe the details of the analytical subsystems, the PARallel Hierarchical Agglomerative Clustering System (PARHACS) and the SIMilarity Classification System (SIMCS). In Section 4, we conclude this paper.

## 2. User Interface and Information Management

The proposed system, shown in Figure 1, can be abstracted into three layers. The first layer is the user interface, namely the Virtual Observatory Portal Layer. The Virtual Observatory Portal Layer provides a user interface for data search, data maintenance, and data analysis for different users. The second layer is the ASTROnomical Information Management System (ASTROIMS). It provides the core functionalities for data management and inquiry. The third layer provides analytical tools, including two data analytical subsystems over a set of computer clusters. While the analytical subsystem receives a query from second layer, it automatically collects necessary data from the second layer, performs the calculation tasks based on the user's settings, and then returns the results to the second layer. This section will briefly introduce the user interface and the information management subsystem.

### 2.1 Virtual Observatory Portal

The Virtual Observatory Portal layer is the entry point of the proposed system for all users. It provides a user interface with three different modules. First, the Personal Management Module enables the users to submit their queries for the target data. Second, the Administrator Module provides necessary functions to facilitate the system managers to maintain the system, to update the hardware/software, to change the system configuration, and to collect data logs. Third, the Analytical Tool Module provides advanced analytical tools that allow users to further process data, such as drawing a diagram and fitting data into a model. The module also has a related meta-log query functions defined in the interface. The meta-log stores the metadata of the source data and user-related information.

### 2.2 Astronomical Information Management System

The design of the ASTROIMS has two major goals. First, it has to provide an integrated and consolidate view of data storage, so that researchers can retrieve research materials and deposit analysis results on a unified platform. Second, it has to provide versatile data access capabilities for the researchers and other modules to facilitate research on different interests. The ASTROIMS adopts the multidimensional data-warehouse concept, which incorporates possible analytical views to index the underlying dataset. Possible index dimensions usually include observation time, space position, and telescope features. Administrators, researchers, or routine programs can use the interface to retrieve bulks of datasets from the main archive database of the project. Each index dimensions can have different levels of granularities, so that analysis can be conducted on more generalized scopes and on specialized areas as well. In addition to the dimensions that support analytical processes, some other dimensions can be served as the targets of analysis. For example, possible candidate variable stars derived from the TAOS database [2] can be labeled and saved for further investigation.

The ASTROIMS is comprised of three core sub-modules: the query processor, the data warehouse, and the data delivery service. The query processor is responsible for analyzing user' queries. It may split the queries into several smaller queries if necessary, re-schedule the order of queries (for better use of cache), and dispatch them to different machines in the third layer by the data delivery service. After the data delivery service receives the results from the third layer, the system will collect and combine them into a complete result, and then store them in the data warehouse.

The ASTROIMS maintains different kinds of data for different purposes. There are four primary components in a computing node to maintain the cluster and to perform local queries: the cluster control module, the computing module, the data pool, and the internet data fetcher. The cluster control module is to keep the status of other clusters up to date. The computing module provides computing power in a distributed environment along with several built-in analytical functions. When a query is received, the system will check the local data pool for necessary information, and then download the missing data from the internet by the internet data fetcher. Whenever the system gets the entire datasets, it begins to perform the computing tasks. The local results are returned to the query processor and then merged into a complete result.

Figure 2 shows the data flow of the ASTROIMS. Users can choose to use the ASTROIMS which stores all the exist information and do further data analysis; Users can also submit a new

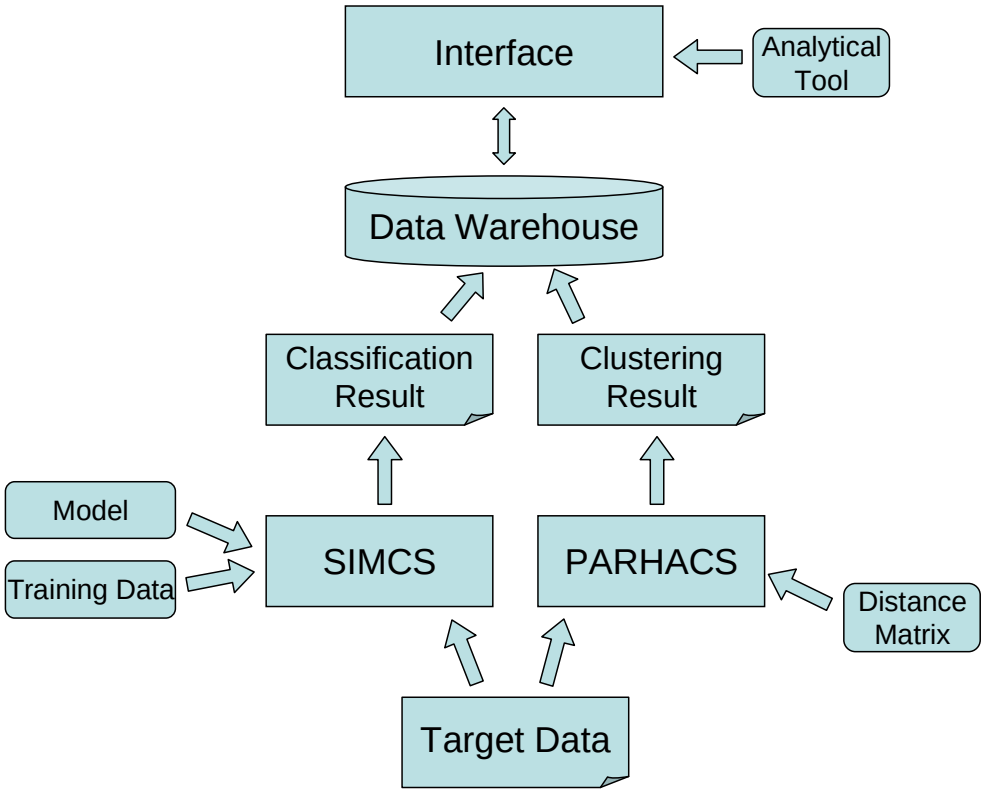


**Figure 2:** Data flow of the Astronomical Information Management System.

classification or clustering task using selected data, and then use the Similarity Classification System or the Parallel Hierarchical Agglomerative Clustering System to analyze the data. When a user assigns a task or specifies new data that come from the internet data fetcher to one of the analytical systems, they will calculate the results and store the results into the data warehouse. Users can get the result and process further analytical tasks through the user interface if the result is ready.

### 2.3 Limitations of the ASTROIMS

The current ASTROIMS is built on a storage sever that can store up to four Terabytes of data, or billions ($10^9$) of astronomical objects. However, the system is on top of a commercial data warehouse system and can be extended to tens of Terabytes of data.

## 3. Analytical Systems

In this section, we will introduce the Parallel Hierarchical Agglomerative Clustering System (PARHACS) and the Similarity Classification System (SIMCS). Both of the systems are used to analyze data.

### 3.1 Parallel Hierarchical Agglomerative Clustering System

The main purpose of using the PARHACS is to reduce the execution time incurred from exe-

cuting clustering algorithms. It parallelizes the hierarchical agglomerative clustering procedure by adapting it into a distributed computing environment, so that it can deal with large-scale data and yet achieve good performance.

The PARHACS helps classify a set of data items into several subsets (a.k.a. clusters) by a distance function. The PARHACS uses a distributed version of agglomerative hierarchical clustering algorithm. Figure 3 shows the four main steps to complete the parallel hierarchical clustering in the PARHACS. First, the PARHACS uses a parallel computing strategy to compute the distance values of all pairs of all data items, which are stored as a distance matrix. Then it reduces the required space of the distance matrix using a threshold. Second, it uses disjoint set operations [5] to find the sets that can be clustered independently. Third, it computes the distance matrix between each disjoint set. Finally, it constructs a complete hierarchical tree using the information from the third step. The details of the PARHACS are described below:

**1.** Computing the distance matrix (in parallel).

Distance Matrix is a matrix that stores the similarities of all pairs of all datasets. The space complexity of a complete distance matrix is $O(n^2)$, given $n$ the size of the input data. This means that the system needs extremely large space to store full information when we are facing even a small size of input data. However, the critical information is usually relatively smaller than the total required space, and thus most information might be useless. To reduce the space cost, we use a threshold to eliminate unnecessary data in the distance matrix: Any pair (elements in the distance matrix) with a larger distance value than the threshold can be deleted. In the case that the removed distance values are required in some computation stages, the system can re-calculate the values on demand.

**2.** Using disjoint set operations to construct disjoint sets (sequentially).

The disjoint set forests algorithm [5] is used to find all the independent sets in a set of data. Because the clustering procedure has to use all the information in the distance matrix, the only way we can distribute the clustering work is to find all the disjoint sets that are related under the given threshold. Here we assume that two data items are in a same group if and only if their distance value is smaller than the threshold. Therefore, we can distribute the clustering tasks easily by assigning different independent sets to different machines for parallel clustering. Note that the task distribution procedure is sequential, while the independent datasets over many machines can be executed in parallel.

**3.** Computing the distance matrix of disjoint sets (in parallel).

Through the distance matrix and the disjoint set operations, the system can find all the disjoint sets under a given threshold. To construct a complete hierarchical tree, the system needs to know the distance between these disjoint sets. Thus, the system uses a strategy that is similar to the first step to calculate the distance matrix of disjoint sets in parallel.

**4.** Complete clustering (in parallel).

After the system obtains all necessary information, it can construct a complete hierarchical tree from the data of interests. The users can use the hierarchical tree to perform further analytical tasks.

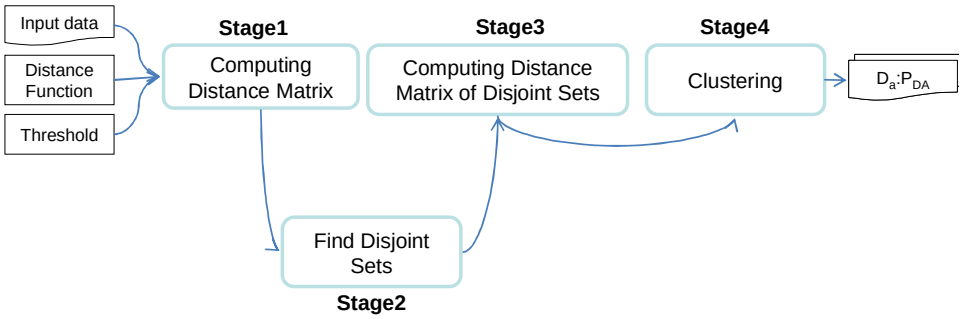


**Figure 3:** The flow chart of the parallel hierarchical clustering algorithm.

### 3.2 Similarity Classification System

The SIMCS is a design framework of multiple classifier system (MCS) [6] that carries out complex classification tasks with large-scale database on support vector machine (SVM) [7] [8] [9]. SVM is an easy and fancy way to deal with complex classification procedures. However, the efficiency of the single SVM drops very fast when the data size becomes larger and more complex [10]. Therefore, we provide a multiple classifier system to divide the data into numerous chunks, and then classify each chunk in parallel with multiple SVMs.

When a user submits a task to the SIMCS, he or she can either choose to train a new model, or to classify the target data using existing models. There are many known ways to train and to select proper classification models such as local accuracy estimation [11] or cross validation [12]. In addition, users can choose to select exactly one classifier model that has the highest accuracy, or simply to combine multiple classifier models that have higher accuracy than a given threshold into one complete model [13]. Then, the system can classify the target data using the combined model.

The SIMCS can utilize the internet data fetcher to obtain new data from the Internet and classify them into different groups by pre-selected models, and store or update the results into the data warehouse of the ASTROIMS for future use.

### 3.3 Limitations of the Analytical Systems

We have used the PARHACS and the SIMCS to process about $10^6$ to $10^7$ datasets of main belt asteroids. The PARHACS spent about two hours using a cluster of 64 virtual machines to get the result; the SIMCS spent about half an hour using a cluster of 11 virtual machines to find an acceptable solution (about 90% prediction accuracy).

## 4. Conclusions

In this paper, we have proposed an integrated information system that supports large-scale data management and analysis for astronomical research. With the proposed three main sub-systems, PARHACS, SIMCS, and ASTROIMS, we can provide the users necessary algorithms, analytical tools, and combination frameworks with a user interface that can deal with different kinds of complex analysis tasks. The proposed system is able to perform automatic data update or maintenance, as well as to reduce the maintenance and operation cost. In the future, we would like to improve

the user interface, and use an existing cloud IaaS (Infrastructure as a Service) system to manage hardware resources.

## Acknowledgment

This research was sponsored by National Science Council, Taiwan under the grants NSC 99-2221-E-008-044 and NSC 99-2218-E-008-012, and the Software Research Center, National Central University..

## References

[1] "The Panoramic Survey Telescope and Rapid Response System," *http://pan-STARRS.ifa.hawaii.edu/*, 2011.

[2] "The Taiwan American Occultation Survey," *http://taos.asiaa.sinica.edu.tw/*, 2011.

[3] "The International Virtual Observatory Alliance," *http://www.ivoa.net/*, 2011.

[4] "The Sloan Digital Sky Survey," *http://www.sdss.org/*, 2011.

[5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein: "Introduction to algorithms, second edition," *The MIT Press,* 2001.

[6] Romesh Ranawana and Vasile Palade, "Multi-Classifier Systems: Review and a roadmap for developers," *International Journal of Hybrid Intelligent Systems,* pp. 35-61, 2006.

[7] Harris Drucker, Chris J.C. Burges, Linda Kaufman, Alex Smola and Vladimir Vapnik, "Support Vector Regression Machines," *Advances in Neural Information Processing Systems,* pp. 155-161 , 1997.

[8] Suykens J.A.K., Vandewalle J., "Least squares support vector machine classifiers," *Neural Processing Letters,* vol. 9, no. 3, pp. 293-300 , 1999.

[9] Nello Cristianini and John Shawe-Taylor, "An Introduction to Support Vector Machines and other kernel-based learning methods," *Cambridge University Press,* pp. 155-161 , 2000.

[10] Don Hush and Clint Scovel, "Polynomial-Time Decomposition Algorithms for Support Vector Machines," *International Journal of Machine Learning,* pp. 51-71, 2003.

[11] Kevin Woods and W. Philip Kegelmeyer Jr. and Kevin Bowyer, "Combination of Multiple Classifiers Using Local Accuracy Estimates," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19, pp. 405-410 , 1997.

[12] Schaffer C, "Selecting a classification method by cross-validation," *Machine Learning,* vol. 13, pp. 135-143, 1993.

[13] E. Kim and J. Ko, "Dynamic classifier integration method," *Journal of Multiple Classifier Systems,* pp. 97-107, 2005.