Flexible Framework for Mining Meteorological Data

Rahul Ramachandran *, John Rushing, Helen Conover, Sara Graves and Ken Keiser

Information Technology and Systems Center
University of Alabama in Huntsville
Huntsville, AL - 35899

1. Introduction

A Framework has been defined as a reusable, "semi-complete" application that can be specialized to produce custom applications [1], or as the skeleton of an application that can be customized by an application developer [2]. Microsoft Foundation Classes [3] and Java AWT [4] are examples of Frameworks that are commonly used by software developers in their applications.

Frameworks have been a focus of software development in the commercial arena because they allow for rapid design and development of robust, specific applications. Application developers in domains such as data analysis and scientific data mining have traditionally lacked such "off-the-shelf" frameworks. As a result, application developers have to build, validate and maintain software systems from scratch, which is expensive and time consuming. The Information Technology and Systems Center at the University of Alabama in Huntsville has focused research and development efforts on providing a Flexible Framework for Data Mining and Analysis. The ultimate goal for this framework is to allow scientists the flexibility to build analysis systems using the framework or use individual components from the framework for their analysis.

2. Algorithm Development and Mining (ADaM) Framework

2.1 Description

ADaM [5] was developed in response to the need to mine large scientific data sets for geophysical phenomena detection and feature extraction. This framework provides knowledge discovery and data mining capabilities for data values, as well as for metadata, and catalogs the information discovered. It contains algorithms for detecting a variety of geophysical phenomena to address the needs of the Earth Science community.

*Corresponding author Address:
Rahul Ramachandran
Information Technology and Systems Center
University of Alabama in Huntsville,
Huntsville, AL 35899
email: rramachandran@itsc.uah.edu

ADaM consists of three basic types of modules: input filters (readers for different data formats), processing modules (general-purpose algorithms and user-defined algorithms) and output filters (writers for different data formats). Since the number of data sets and algorithms for analysis continues to increase and change, the system was designed to be extensible by the use of plug in modules. New modules can be added to the framework easily. Thus, the framework is designed to evolve along with the demand. Within this framework, a mining task is assembled from these modules as a series of steps, with results from each step passed to the next one. Figure 1 illustrates both ADaM's data processing stream, as well as the three basic types of modules: input, processing, and output.

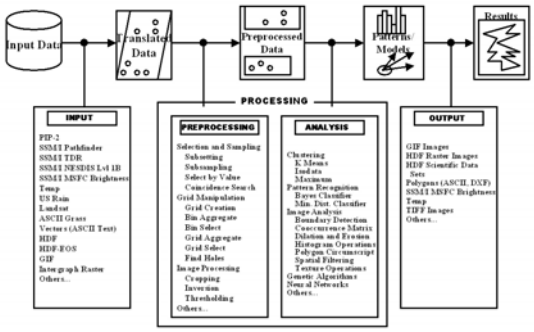

Figure 1: ADaM Processing Architecture

The use of data input filters, specialized for a variety of data types, has been instrumental in simplifying the development of the processing and output operations. Each input filter translates the data into a common internal structure so that the processing operations can all be written for a single data representation. This allows the addition of new operations to the system without having to address input data format problems. Similarly, the addition of a new input filter provides access to the entire suite of processing operations for the data type in question. ADaM currently has many different operations that can be performed on the input data stream. These operations vary from specialized atmospheric science data set specific algorithms to generalized image processing techniques. The last step in the mining process is the selection of the output format. Since the input data has been converted to ADaM's data model, the output modules allow the user the option to select either the input format or a different format for the final data product. In the same manner as the input modules,

the output filters effectively insulate the processing operations from having to support all the possible output formats.

2.2 Unique Design Features

The internal data model in ADaM has been optimized for Earth Science Data. It can contain raster and vector numeric data structures and semantic fields such as Latitude, Longitude, and Time. Because an important aspect of mining is the automated processing of vast data stores, the framework was primarily designed such that one could build a system that would work at data archives in a batch mode. The framework was also configured to allow client/server architecture to permit remote use across the network. This client/server architecture also provides the users configuration capabilities to set up coarse grain parallelization. The framework architecture and design optimize executables built at runtime, depending on the mining plan specified by the user. It also provides the capability to build customizable packages for different applications. Examples of these applications are described below:

- **PM-ESIP Custom Order Processing System -** The Passive Microwave Earth Science Information Partner (PM- ESIP) system provides science researchers and users with the capability to interactively customize and receive hydrologic data sets derived from the latest space-based passive microwave instruments. The processing capability of ADaM is utilized to provide on demand customizable products. A web-based interface allows researchers to search, select data sets and different operations, and executes their operations on the backend servers. The researchers are then notified about their orders along with information on how to obtain their customized products once the system has completed processing their defined tasks. (URL: http://pm-esip.msfc.nasa.gov/order_data.html)

- **Tropical Cyclone Detection and Wind Speed Application -** The Advanced Microwave Sounding Unit is a microwave radiometer that can be used to detect temperature at different levels of the atmosphere. Based on gradients in temperature measurements in a given area, it is possible to estimate maximal sustained radial wind speed [6]. This wind speed estimate can be combined with other factors such as ice scattering and moisture in order to detect tropical cyclones. Selected operations from the ADaM framework were extracted and combined to build a stand-alone application to detect Tropical Cyclones and to estimate their wind speeds from AMSU data. This application allows AMSU data to mined in near real time as the data is ingested. (URL: http://pm-esip.msfc.nasa.gov/cyclone/)

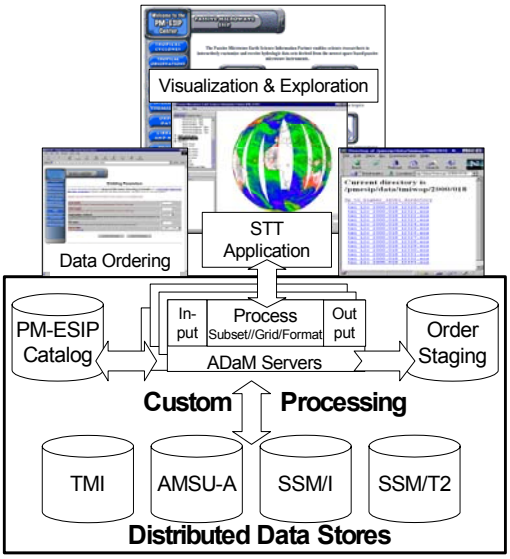

Figure 2: PM-ESIP Custom Order Processing System

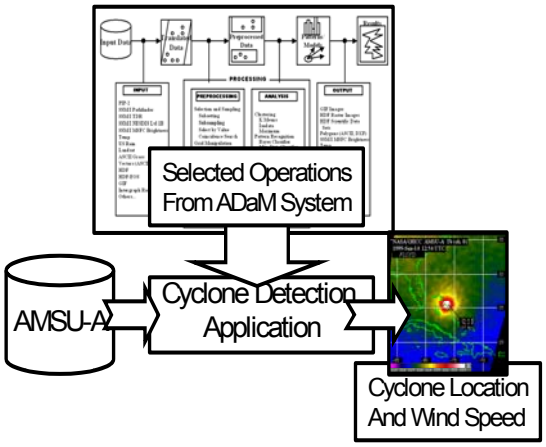

Figure 3: Tropical Cyclone Detection and Wind Speed Application

- **ADaM Lite (General Purpose Mining Utility) -** The flexibility of the framework allows ITSC/UAH to package a compact Data Mining System (ADaM Lite). The main intent is to provide researchers with a turnkey software product with mining capabilities that can be utilized for data analysis. Selected input/output filters, preprocessing operations and the most commonly used pattern recognition and mining algorithms are packaged together with an easy to use interface (see Figure 4). This interface allows the end users to create complex mining plans. The interface also provides the documentation about the different operations and the parameters that

can be adjusted to modify the behavior of the operations. In addition, there are several other utilities incorporated in this interface, such as sample selection capability (important for supervised classification).

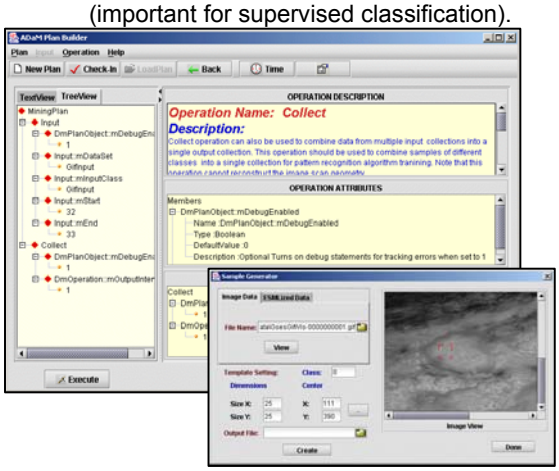

Figure 4: ADaM Lite Interface

2.3 Rationale for an Alternate Framework

During the lifetime of the ADaM system, many things have evolved. The number of operators and data formats supported by the tool has grown, the number of developers working on the product has increased, and ADaM is now being used in ways not originally envisioned for the product. Also, there are many technologies available now that were not available or were not in a stable state when ADaM was designed. In a recent review of the ADaM architecture, the design team has identified several improvements that can be made to the ADaM system. These improvements are significant enough to warrant a fundamental change to the ADaM framework.

a. Data Model Issues

At the time ADaM was originally designed, data interoperability technologies were in a state of infancy. In order to provide general-purpose mining operators, it was necessary to define some mechanism for these operators to communicate with one another. The ADaM system addressed this requirement by defining a standard data model for the input and output of mining operators. This data model solved the interoperability problem in the sense that one could implement an operator once based on this model and reuse it over and over again in different applications. Operators could be chained together seamlessly. The same is true for readers and writers of particular data formats. A single translation module is all that is required to make data in that format available for a wide range of applications. However, this approach also has some drawbacks:

- Integrating third party algorithms into ADaM requires some effort. Algorithms must be translated to use ADaM's internal data model both for input and for output. This makes assimilation of new versions of the algorithm problematic.
- Because the ADaM data model must support all of the information needed by all of its operators, it has become somewhat complex. This means that there is a learning curve for new developers.
- The complexity of the data model also has an impact on size and performance. Although care has been taken to minimize memory usage, the model has become too heavy for very simple applications.
- Providing ADaM algorithms for others to use is a challenge. If ADaM operators are to be exported to some other system, they need to be ported to use that system's data model.
- The initial design requirements for ADaM were to tailor the framework for Earth Science datasets, which are typically raster or vector numeric. The current internal data model design cannot easily handle certain kinds of data such as categorical and text.

All of these limitations can be addressed by decoupling the data model from the operators. However, the data interoperability issues must then be resolved by some other mechanism. A solution for this problem is described in section 3.

b. Infrastructure Issues

The initial design of ADaM was focused towards efficiently running mining operations on large sets of data in a batch mode. A significant amount of infrastructure was added to facilitate this use of the system, including a mining daemon for accepting and queuing requests and a database for storing references to data files. While a significant portion of ADaM applications require large batch processing operations, there are many other applications for which this infrastructure is not required and is not appropriate. Scientists have expressed interest in using the system in an interactive mode for exploration. Scientists have also expressed interest in building lightweight standalone mining applications based on a small subset of the ADaM operators. In these cases, it is desirable to decouple the mining operators from the batch processing infrastructure.

3. New Flexible Framework for Data Mining

The researchers at ITSC/UAH are in the process of designing and building a new framework for Data Mining and Image Processing. This new framework will extend ADaM's current capabilities. The new vision is to build a comprehensive virtual repository of mining and image processing operations where

the operations cover a variety of data types (Raster numeric (images), vector numeric (GIS), character / text (documents), categorical (nominal text) and time series). The new framework design will have a lightweight system architecture and will allow several simple data models to work with the operations. The conceptual design of this framework can be seen in Figure 5.

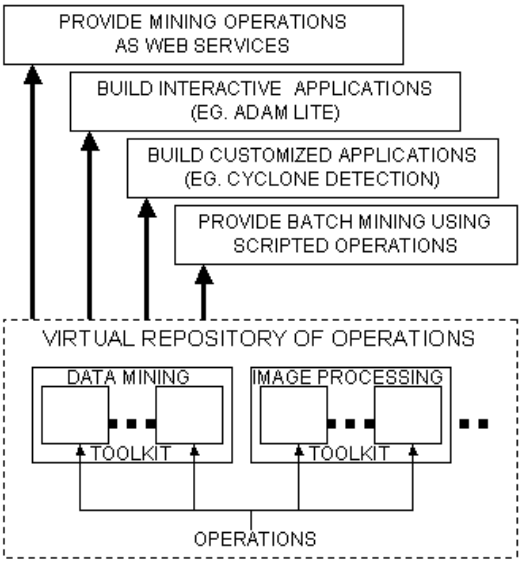

Figure 5: Virtual Repository for Mining and Image Processing

This framework will allow developers to build individual components for each operation where the component will consist of driver program and the mining/image processing operation. It will leverage the extensive library of existing operations from ADaM as well as other systems. It will allow the flexibility of creating specialized and generic standalone applications, and will facilitate use of data mining modules in other processing environments. This framework will also support the implementation of data mining, image processing and operations as web services. The key differences between the new framework and the existing one are the decoupling of the data model and processing infrastructure from the operators themselves.

a. Data Interoperability

Eliminating the requirement that mining operators use a common data model allows the implementers of those operators to read and write data in whatever format is appropriate, and use internal data structures that are most efficient for their particular circumstances. In data mining applications, it is often necessary for several mining operators to be used together. The output from one operator serves as the input to the others. If the operators do not share a common data model, it is necessary to provide a means for them to share information.

The Earth Science Markup Language (ESML) [7] developed at UAH/ITSC is an ideal mechanism to facilitate data interchange between mining operators. ESML is a language that can describe the structure and semantics of Earth Science data (and other types of data as well). Descriptions are provided using external metadata, so the original data files and applications need not be modified. An ESML-enabled translator will be able to read and write data in any format for which there is an ESML description. This will eliminate the need to write large numbers of data translators. The use of ESML will also facilitate visualization of the data. In order to add a new operator to the framework, all that is required is to provide ESML descriptions of the inputs and outputs of the operator.

b. Processing Infrastructure and Applications

The ADaM data mining system is used both for batch processing and for interactive exploration. In batch processing mode, large data sets are processed for a particular purpose such has identifying large scale storms or classifying clouds. In this case, the data sets of interest and the operators required are already known. In interactive mode, scientists are evaluating different operators and different parameters for those operators. They want quick feedback, and ways to visualize the results of the operators. Since these usage modes are so different, it makes sense to support them with different mechanisms.

In the new framework the infrastructure for data management and batch processing will no longer be coupled with the mining operators. Application developers will be free to use whatever databases, queuing systems, schedulers, scripting languages or other tools that are appropriate for their applications. Many high quality data management tools are now available for free, and it is no longer necessary or desirable to bundle these components with mining operators. Instead of enforcing a single processing architecture, the framework will provide tools and templates that will allow the rapid construction of new and flexible applications using the library or mining operators. Types of applications supported will include:

- Web Services: Mining operators will be web enabled so that data can be sent to a mining node and results sent back to the user's desktop. Multiple operators can be chained together to create an application.

- Interactive Applications: These applications will provide graphical user interfaces to specify which mining operators to run and how to run them. They will also provide tools for visualizing the inputs to and outputs of mining operators.

- Batch Applications: These applications will apply user defined mining processes on large data sets. Most batch applications will be controlled by scripts.

- Custom Executables: Custom executables can be developed for performance critical applications. Mining operators will be tailored to the specific needs of the application and linked into a single executable.

4. Summary

The current ADaM framework is an excellent design for its requirements. It has been used in many different forms and applications for research, analysis and data processing. Several of these applications have been described in this paper. As ITSC/UAH expands its mining and image processing capabilities, it is looking at a new loosely coupled framework of operations. This design will provide all the capability and flexibility of the ADaM framework without the complex overhead.

5. References

[1] Fayad. E. Mohamed and Douglas C. Schmidt, "Object-Oriented Application Framework", Communications of the ACM, October 1997, Vol.40, No. 10

[2] Johnson E. Ralph, "Framworks = (Components and Patterns)" Communications of the ACM, October 1997, Vol.40, No. 10

[3] Microsoft Foundation Classes (MFC), http://msdn.microsoft.com

[4] Java Abstract Window Toolkit (AWT): http://java.sun.com

[5] Keiser, K., J. Rushing, H. Conover and S. Graves, 1999. Data Mining System Toolkit for Earth Science Data. *Earth Observation and Geo-Spatial Web and Internet Workshop (EOGEO)-1999*, Washington, Feb 9-11.

[6] Spencer, R.W., and W.D. Braswell, 2001: Tropical Cyclone Monitoring with AMSU-A: Estimation of Maximum Sustained Wind Speeds. *Mon. Wea. Rev., June 2001, Vol. 129, pp.1518-32.*

[7] Ramachandran, R. M. Alshayeb, B. Beaumont, H. Conover, S. Graves, X. Li, S. Movva, A. McDowell and M. Smith, 2001: Earth Science Markup Language: A Solution for Generic Access to Heterogeneous Data Sets. NASA Earth Science Technology Conference 2001,