Journal of Coastal Research	SI 50	pg - pg	ICS2007 (Proceedings)	Australia	ISSN

Data Management Challenges in Coastal Applications

S. Tummala and T. Kosar
Department of Computer Science
and Center for Computation & Technology
Louisiana State University,
Baton Rouge, LA, 70803, USA
{sirish, kosar}@cct.lsu.edu

ABSTRACT

Tummala, S. and Kosar, T., 2007. Data Management Challenges in Coastal Applications. Journal of Coastal Research, SI 50 (Proceedings of the 9th International Coastal Symposium), pg – pg. Gold Coast, Australia, ISBN

The goal of this paper is to identify the data management challenges in coastal applications such as hurricane track prediction, storm surge modeling and coastal erosion modeling. The problems in managing the data due to different paradigms such as increase in data and computational requirements, conceptual changes in the computational models involved, and changes due to the evolution of objectives of the models are explained. Potential problems in a complete processing cycle that can be solved using automation are enumerated right from the selection of input data to the archival of output data or feeding the output data into a visualization system. Challenges to the user like having to complete the data management operations manually, to learn the underlying complexity of the resources, and to intervene during data placement failures are explained. Specialized tools for data placement and automation of specific tasks and the workflow mechanisms that are being used currently to automate the entire end-to-end cycle of scientific computation are mentioned as a solution to these problems.

INTRODUCTION

Coastal applications like hurricane track prediction, storm surge modeling, coastal erosion modeling are characterized by computation of large scale coastal data. These applications are already not only computationally intensive but are also becoming increasingly data intensive in nature. For example, The Earth Scan Laboratory of LSU (EARTH SCAN LAB, 2005) is collecting approximately 40 GB of data each day in the form of telemetry data for research purposes and emergency responses but is able to store only one fourth of it due to the storage limitations. With the objectives of the applications changing over a period of time, changes in the requirements and nature of the applications being used are making the data management challenges much evident. In this scenario, we would like to present the data management challenges that are increasingly becoming an issue for achieving the desired performance in these applications. These challenges may be due to some paradigms like increase in the data and computational requirements (increasing grid resolution, data resources, and number of runs), conceptual changes in the computational models involved, and changes due to the evolution of objectives of the models.

PARADIGMS FOR DATA MANAGEMENT CHALLENGES

The increasing data management issues and the resulting overhead of manual work to the end user (i.e., coastal scientist) can be due to the one or more of the following paradigms. In each of these paradigms, the causes leading to the data management challenges are explained.

Increasing Data and Computational Requirements

Increases in the data and computational requirements can be attributed to the factors such as increase in the grid resolution of the study area, increase in the number of data sources to be taken into account for forecasting, and increase in the number of runs of the model per forecast. These are illustrated as follows:

Increasing Grid Resolution: Grid resolution represents the distance between two points where the future atmospheric parameters are forecasted. For each forecast, the model needs to compute the projected parameters at each point in the grid. Hence the grid resolution decides the number of points where the computation of forecast is performed by the system. If the forecast is done on a coarser grid, the number of points on the grid will be low, implying that the computation points for the system to calculate will be lower. If the resolution of the grid is finer, then the system will have to compute the weather parameters at larger number of points. Hence the data requirements of the system will increase and this may cause problems in the placement and management of such large amounts of data. In 2002, the MEAD system generated 600GB of data per storm of from 100 forecasts per each storm by using a resolution of 20km. At present, it is estimated that the hurricane ensemble predictions will be using a grid resolution of 1- 2 km resulting in several tens of Terabytes of data (RAMAMURTHY, 2002).

Increasing Number of Data Sources: The data for climate forecasting applications is collected from different sources like satellite imagery, sensor networks, and LIDAR data from radar appliances. All these constitute into High resolution bathymetric, topographic, and airborne gravimetric data (NOS, 2005). In the coastal climate forecasting applications, the models are run while their scope is being limited to a fixed geographical boundary location. In the due course of time, if the scope of the geographical area under the coverage of the model is increased, then the numbers of data sources that are needed to be taken into account also increase. This will lead to the system having to place and manage larger number of input data. This aspect also causes an overload to the data assimilation components of the weather forecasting models as there is an increase in the numbers of data items and sources to be assimilated. Steven Smith, the director of AccuWeather, states this problem as *"So not only are you talking in the broad sense of distributed computing, where modelers are trying to improve forecasts using the capabilities of distributed computing, we are also trying to figure out how do we handle not megabytes or gigabytes of data, but terabytes of data. And when you are going to 24 different sources, which have potentially completely different operating procedures, this creates a headache* (GOTH, 2005).*"*

Increasing Number of Runs of the Model: In weather forecasting models, a number of runs are made for every forecast so that the most accurate prediction is made. If the numbers of runs of the model are high, then the data placement and management requirements of a system will increase causing a threat to the performance of the system as the system has to maintain and consider larger amounts of data throughout its forecasting cycle before it makes a final forecast. For example, The National Hurricane Center (NHC) in Miami, Florida issues 72 hr tropical cyclone track and intensity forecasts four times per day for all storms in the north Atlantic and eastern north Pacific east of 140°W. The Central Pacific Hurricane Center (CPHC) in Honolulu, Hawaii issues similar forecasts for tropical cyclones in the north Pacific from 140°W to 180°W. WAVEWATCH III (TOLMAN, 1997) is a third generation model used for the simulation of the near shore waves. In the WAVEWATCH III simulation, for each track from the NHC, a separate run may be performed (SURA, 2005). Computing the models for each such track and analyzing the results of all the tracks for comparison and correlation will be required to produce the final forecasting.

Conceptual Changes in the Computational Models Involved

The intricate complexities of the model that is being implemented may cause problems to the data placement and management. Sometimes, changes in the conceptual details of the model will result in extra data placement and management overhead for the system. These changes in the conceptual details may be the changes in the existing model such that more data is computed, assimilated, staged and transferred at a time. This will lead to more amounts of the data that is to be handled and placed by the system during the forecasting process. For example in the ENSEMBLE method for MEPS system of forecasting the hurricane track, a set of multiple predictions are utilized at the same time. These multiple predictions are generated from reasonably different initial conditions and/or with various credible versions of models. In the year 2001, the total number of storms was more than 40. The MEPS system made over 100 forecasts for each hurricane case from which the resulting data volume was 600 gigabytes per storm from grid spacing of 20 km (RAMAMURTHY, 2002).

Evolution of Objectives of the Models

The coastal and weather forecasting applications till now have been applied to a specific purpose such as hurricane track prediction, storm surge modeling, and coastal erosion. But in the recent times, the need for overall climate modeling has been identified (NSF, 2002). To satisfy this need, various countries and government agencies are joining hands in building global observation systems of earth. One such initiative is Global Earth Observation System of Systems (GEOSS) (GEOSS, 2003). This long term project strives to provide observation facilities for the entire earth climate in all the dimensions possible. Another such application is NASA's Earth System Enterprise Plan (NASA ESS, 2003). The reason for mentioning these two projects is to convey the grandeur of the applications that come into reality in these projects. The applications that underlie in the implementation of the project need to handle several terabytes and even petabytes of data each time an activity is forecasted. This clearly explains the anticipated need for better data placement in the event of such applications.

PROBLEMS DURING AN END-TO-END PROCESSING CYCLE

Considering a single end-to-end processing cycle in an application run by the user, the data management issues that are arising due to the above paradigms are explained as follows:

Data Pre-processing

Data Collection: It can be either by selection of input data or from assimilation of a real time recorded data or by regeneration of input data from some other processing cycle. Data specific to particular time and geographical and climatic conditions are used for performing simulations in coastal applications. With the amount of data collected as inputs increasing, there is a need for meaningful selection of data from the ever growing archives.

The selection of input data from large amount of existing data sources can be aided by the use of metadata catalogs. Metadata that provides meaningful information about the semantics of the data can help the user in making the selection of only a required set of data. This is being facilitated by providing a querying mechanism to the metadata catalog. One such solution proposed by SCOOP (UAH IT, 2006) is using a metadata catalogue. MDS (KESSELMAN et al., 2001) and SRB (BARU et al., 1998) are two technologies being used to provide a centralized information service to find out the data which may be widely distributed over heterogeneous storage resources.

The selected input data may be available centralized at one location or decentralized at different locations. In fetching all the data from different locations, the timely performance of the system is affected. For example, in the WAVEWATCH III modeling of hurricane track, the input data of GFS winds and Sea Surface Temperatures (SST) (REYNOLDS, 1988) may be collected from different sources to be fed into the system. In this aspect better data placement and assimilation systems are needed to avoid the inefficiency.

When the entire data may be available at centralized locations, the transport mechanism used to fetch the data may pose the problems. Some problems may be due to the performance of the data source. If the number of users of the data is more, then the availability of data will be affected. Failures in data transfer must be rectified immediately such that there is maximum fault tolerance in the system. For achieving the fault tolerance by resuming the data transfers after failure, logging of the data transfer activity into a persistent storage (DEELMAN et al., 2006) is considered to be effective.

Input data is usually fed into the system in the form of files for processing. For better forecasting results, in the hurricane track prediction, real time data is assimilated into suitable format (BOGDEN et al., 2005). This real time data is the data pertaining to different parameters at the different locations. With the advent of distributed sensor networks, there will be high increase in the number of data sources to be assimilated. Also the nature of the data sources will be dynamic which means that the system must be able to deal with on and off stream real time data from such sensor networks. The storage and management of such transient or dynamic stream of data must be incorporated for facing this problem. DDDAS is one such effort where real time wireless sensors from both water and wind are used to feed data into coastal modeling applications in real time (NSF DDDAS, 2006).

Data Transformation: Coastal modeling applications involve transforming the collected input data into a form that is optimized to be input for processing. This would bring in additional processing tasks which may involve converting from one data format into another, extracting the needed data from collected raw data files, and rearranging the data to optimize the data access (MICHALAKES et al., 2004). When dealing with the preprocessing of large data sets, large amounts of processing, storage and caching resources are consumed very easily which might lead to the inefficient use of resources with time.

Staging-in the Input Data

In coastal modeling applications, the data may be preprocessed at the computation site or at a different site prior to performing the actual processing. If the pre processing is done at a different site than the actual computation site, the required input data has to be staged-in to the computation site prior to actual processing. This will bring in the issues of the data transfer discussed earlier. If the pre processing is done at the same site as actual computation, the issues of cleaning up of the remaining intermediate and persistent data that may not be needed for computation will also arise. As mentioned earlier, due to diverse and geographically distributed nature of the resources, the data may not only have to be collected from diverse sources but also have to be deployed on diverse computation centers which would add to the complexity and manual work to be dealt by the user. For example, in WAVEWATCH III simulation (TOLMAN, 1997), in the absence of a portal that enables automatic fetching of the data and initialization of the model, it becomes the task of the user to ensure that the staging is done effectively.

Actual Data Processing

Due to the computational intensiveness of the coastal applications, during computation, many processes of the application may simultaneously perform reads or writes on one or more of the local disk resources. During high performance computation of coastal data like in hurricane simulation, large amounts of data will be read from and written into the disk resources at the computation site. In such a scenario I/O could easily become a bottleneck hindering the overall turnaround time. This problem demands attention particularly in the case of real time simulations, where high performance is desired to obtain the results in least possible time. High performance parallel I/O mechanisms working synchronously both from the application side and from the underlying resource's side can prevent I/O from becoming a bottleneck. In the WRF model simulation, the use of a parallel I/O over a sequential NetCDF has improved the overall wall clock time between the application and the data (MICHALAKES et al., 2004).

The produced output data may be required to be modified for further processing, visualization or for backup and archival. Hence processing upon the output data will again bring in the computation tasks and the data management challenges associated with it like in the pre processing stage of the data.

Staging-out the Output Data

Staging-out of the data from the computation site may be done to archive the output data or to feed it as input into an analysis or visualization system. This will involve the movement of data from the computation site to the archive or visualization site. In case of staging the output data into a visualization system, storage allocation, caching and data placement related issues will arise. In case of archiving the data, the system not only has to take care of the three issues, but also has to register the data with the archive's metadata catalog to maintain the output data's provenance.

With the large amounts of the output data being generated at a high speed from applications like hurricane track prediction, archiving the data can become a tedious task. Though storage allocation may not become a problem in this case, the slow disk speeds can create I/O bottle neck in archiving the data and thus can block the resources during archival process for a long time. The use of non archival storage resources to cache the data during large data transfers can consume the resources easily.

Visualizing the Data

The requirements for visualizing the data may include the need to visualize and comprehend increased amount of output data than before, real time visualization of model results on the fly, multiple sets of output data for comparison and correlation, data of a multi dimensional nature and the need for interactive visualization. Some times during the interactive visualization of data, even the model processing of input data is done to produce data required for representing the output data.

In case of increased amounts of output data, the underlying visualization infrastructure should be able to deal with the storage and caching needs from the size of the large data sets, processing and display requirements. If the data to be visualized is at a remote site, then network performance and latency issues also will have to be taken care of. In case of real time visualization of model results on the fly, the underlying system has to be tuned for the automation of data processing, resource optimization and data integration issues so that the entire processing cycle is done in a coordinated manner to optimize the total turnaround time. During comparison of different output data sets during visualization, additional data processing overhead will be placed when the system is used to compare data sets in different formats and those which need conversion before visualization.

THETIS (NIKOLAOU et al., 1997) is one such system which tries to address the requirements of scientists, engineers and decision-makers to access, process and subsequently visualize data collected and stored in different formats and held at different locations. The need exists for tools that enable the integration of the data, together with their associated data models, data interpretation techniques, and visualization requirements. The objective is to build an advanced integrated interoperable system for transparent access and visualization of such data repositories, via the Internet and the World Wide Web.

CHALLENGES TO THE USER

With above data management issues, the end user of the application has to complete the end-to-end processing cycle. In this process, he is faced with many technical challenges which are explained as follows:

Underlying Complexity of the Resources

With the heterogeneous, distributed and geographically distant nature of the resources, the user has to keep track of the specifications of the tasks involving each of the distinct resources. In this process, he is forced to learn and apply knowledge of the underlying hardware and software resources at each site. The heterogeneous nature of the involved computation, storage and network resources will push the end user scientist to learn the different computational procedures, file structures and directory mechanisms, data placement and movement procedures. If the computation and storage resources support heterogeneous transport mechanisms of data, the user has to learn and apply different data placement techniques specific to each different mechanism required in the entire set of data placement tasks to be performed.

In this scenario, a higher level abstraction which would mask the underlying heterogeneous nature and complexity details of the resources from the user is desired. This will enable the coastal scientist in having to specify just the higher level task specifications and need not having to worry about the underlying mechanism so as to be able to concentrate more on the data analysis and visualization.

Manual data management

With the increased amount of coastal data sources, the user is required to gather the inputs from different sources. The user has to query one or more metadata catalogs to get the list of available storage resources holding the required data. As mentioned earlier, the availability of the input data affects the timely performance of the end-to-end processing cycle. Replication mechanisms can be used to place the data set in multiple locations based on projections and forecasts of the demand for the required data. If a robust replication mechanism is not available, the user would have to spend time on moving the data from a distant location which can take longer than expected.

Once the required data is located and selected, the user would have to use the supported transfer mechanism by the source location to move the data. During the movement of data from source to computation sites, the users may have to get the required storage allocation done in order to ensure that the size of the input data is being supported by the destination.

The user would want to extract only subset of data from the collected input data, reformat and rearrange the data to provide optimized access to the application. This would involve him to run pre-processing programs upon the input data. This processing upon the input data sets would block the resources for a long time if the input data set is large. Scripts, programs and other processing mechanisms would have to be employed by the user upon the input data during this pre-processing.

The user has to consider the storage, caching, network and computational complexities involved to make a selection of set of resources to complete the processing cycle. This selection would involve either moving the input data to the application or vice versa or even moving both application and data to a new site for processing. In all these cases, the user has to do the staging manually unless he employs special programs to do so.

During the actual processing of the data, the efficient allocation of processing mechanism with the availability of large amounts data upon the limited amount of computational, storage and network resources chosen will affect the timely completion and overall performance of the application. The specification of the process scheduling mechanism upon the chosen resources is another task in this regard to be taken care of manually by the user.

The user will be needed to perform processing of the output data for the required reformatting or rearrangement of the output data to make it suitable for the analysis or visualization. In this stage, the user will need to take care of staging-out the data either manually. In case of archiving the data, the user would be needed to gain the access, authorization, and allocation of the destination storage. The user will also have to maintain track of the archived data by registering it with metadata catalogs or by naming the data appropriately for later reference.

Manual intervention during failures

Due to heterogeneous and distributed nature of the resources that may be involved in the end-to-end processing cycle, a broad range of constraints and complexities come into picture which have to be taken care of by the user. This additional work may also be due to the absence of inherent workflow mechanism to co-ordinate all the heterogeneous resources for maximum efficiency. This mechanism also has to rectify any resource or process failure during the end-to-end cycle so that the user does not have to intervene during the data management failures.

Due to the underlying hardware and software fabrics involved, data placements in heterogeneous environments may not always be perfect. Network failures, packet loss during data transfer, long time hanging of transfers and data corruption may occur during the movement of data from one point to another. With the dynamic nature of the resources, the data placement system used for moving data has to adapt to the changing environment to optimize the data placement process. In the absence of data placement schedulers like STORK (KOSAR, 2004) or reliable data movement tools like RFT (ALLCOCK et al., 2004), the user has to retry the failures that occur. Network failures, storage source or destination crashes may not easily be detectable for the user to take timely rectifying actions. In such cases, the data placement system has to identify failures in a timely manner and has to take the required steps without bothering the user.

DISCUSSION

According to the 'Strategic Plan for the US Climate Change Science Program (CCSP)', one of the main objectives of the future research programs should be *"Enhancing the data management infrastructure"*, since *"The users should be able to focus their attention on the information content of the data, rather than how to discover, access, and use it."* [CCSP 2003]. This statement by CCSP summarizes the goal of many cyberinfrastructure efforts initiated by DOE, NSF and other federal agencies.

There have been several efforts in achieving this goal using state-of-the art techniques. Kosar et al introduced the concept that the data placement efforts which have been done either manually or by using simple scripts should be regarded as first class citizens and these tasks should be automated and standardized just like the systems that support and administer computational jobs. Data intensive jobs need to be queued, scheduled, monitored and managed in a fault tolerant and efficient manner. They have designed, and implemented the first prototype batch scheduler specialized in data placement: Stork [KOSAR 2004, KOSAR 2005]. Stork provides a level of abstraction between the user applications and the underlying data transfer protocols; allows queuing, scheduling, and optimization of data placement jobs.

NSF has recently funded development of a related project, PetaShare: an innovative distributed data archival, analysis and visualization instrument for data intensive collaborative research. [PETASHARE, 2007] PetaShare will enable transparent handling of underlying data sharing, archival, and retrieval mechanisms; and will make data available to the scientists for analysis and visualization on demand. An initial prototype of PetaShare will be deployed at five Louisiana campuses. PetaShare will leverage the existing 40 Gigabit per second Louisiana Optical Network Initiative (LONI) infrastructure to make the interconnections, fully exploiting high bandwidth low latency optical network technologies.

Other related efforts include work on end-to-end workflow management. The collected data need to be moved from the source sites to the computation sites for processing as required, and the results then sent to the interested parties for further analysis and visualization or to the storage sites for long term archival. This end-to-end process needs to be well managed and coordinated. Only with a thorough orchestration of job and data workflows, an end-to-end system with least human intervention can be developed.

During the design of the end-to-end job and data workflows, several criteria need to be taken into consideration; such as workflow mapping criteria and timing, workflow delegation criteria and decision points, the decision of resource assignments, pre- and post-staging criteria, and workflow extension and reduction criteria. The workflow planning software needs to construct a high-level plan for the entire workflow ahead of time and provide the workflow execution software with a general structure of the workflow.

A workflow management system can coordinate the entire end-to-end process to ensure the reliable and timely completion of the entire workflow. By managing the task dependencies and by interacting with the underlying scheduling and execution components the entire sequence of the end-to-end processing cycle can be maintained by the workflow manager and scheduled automatically retrieving from failures if any occurring in between.

One such tool is Pegasus (DEELMAN et al., 2006) workflow planning system developed by ISI at USC. Pegasus consults various Grid information services to find the resources, software, and data that are used in the workflow. A Replica Location Service (RLS) (CHERVENAK et al., 2005) and Transformation Catalog (TC) (WILDE et al., 2004) are used to locate the replicas of the required data, and to find the location of the logical application components respectively. Pegasus also queries Globus Monitoring and Discovery Service (MDS) (KESSELMAN et al., 2001) to find available resources and their characteristics. The workflow execution of Pegasus is based on static planning and its executable workflow is transformed into Condor jobs for execution management by Condor DAGMan (COUVARES et al., 2006).

In order to ensure the reliable completion of the computational procedures, practice of using PBS, Condor, and LSF is already in use. Likewise, specialized data discovery systems like MDS, SRB MCAT (BARU et al., 1998), and Metadata catalog service (SINGH et al., 2003) can be used to gain centralized access to widely distributed data. Globus RLS, ADA, GFRM are some of the replication mechanisms developed to ensure the timely availability of the data in high demand. Scientific data archives and their data access brokering technologies like SRB, EOSDIS (CARLONE, 1992) can be used for the archival of the data for retention and permanent access.

CONCLUSION

The increase of data and computational requirements of the application are bringing in the issues of data management in coastal applications. Such issues are forcing additional computational and data management work upon the coastal scientist which is amounting to more than the actual science of his domain. Through this paper we have discussed the various causes for the data management challenges. The issues that are culminating from the arising data management challenges are explained. The additional work being imposed upon the user in a way to deal with these data management issues are explained. We have explained as how the user could make use of the specialized tools for automation, failure recovery and data management. To hide the underlying complexity of the end-to-end processing workflow, the user can make use of the workflow managers so that he can concentrate more on his science than having to take care of the data management challenges.

ACKNOWLEDGEMENTS

This work was supported by NSF grant CNS-0619843, Louisiana BoR-RCS grant LEQSF (2006-09)-RD-A-06, and CCT General Development Program.

REFERENCES

(Earth Scan Lab, 2005): Earth Scan Laboratory Staff, 2005. The Earth Scan Laboratory, Louisiana State University, A description and Brief History.

(Ramamurthy, 2002): Ramamurthy, M., Jewett, F.B., 2002. Cyber infrastructure for Hurricane Ensemble Prediction. NCSA Faculty Fellow project. University of Illinois, Urbana Champaign, Presentation.

(NOS, 2005): NOS Gulf of Mexico Storm Surge Partnership Project Data collection and Observations 2005. Enhanced Resiliency of Coastal Communities to Storm Surge and Flooding through Improved Data, Models, Tools, and Methodologies, outreach material.

(Goth, 2005): Goth, G., 2005. "Weather Data Industry on Brink of 'Explosion,'" IEEE Dist.Systems Online, vol. 6, no. 11.

(Sura, 2005): Sura It Committee Meeting Summary and Report. November 2005.

(Nsf, 2002): National Science Foundation, 2002. Cyber infrastructure for Environmental Research and education. Report from a workshop held at the National Center for Atmospheric Research October 30 - November 1, 2002.

(Geoss, 2003): Global Earth Observation System of Systems 2003. Earth Observation Summit, Integrated Earth Observation System Intergovernmental Ad Hoc Working Group. Concept Press release.

(Nasa Ess, 2003): National Aeronautics and Space Administration 2003. Earth Science Enterprise Strategy.

(Uah It, 2006): University of Alabama Huntsville IT and Systems Center 2006. Scoop Catalog Services. Application Developer's guide.

(Reynolds, 1988): Reynolds, R. W., 1988: A real-time global sea surface temperature analysis. J. Climate, 1, 75-86.

(Deelman et al., 2006): Deelman, E; Kosar, T; Kesselman, C; and Livny, M., 2006. What Makes Workflows Work in an Opportunistic Environment? In Concurrency and Computation, Practice and Experience, Vol.18 No.10 (2006) pp.1187-1199.

(Nsf Dddsas, 2006): Nsf. Dddas: Dynamic Data Driven Applications Systems. 2006. DDDAS Workshop report.

(Michalakes et al., 2004): Michalakes, J., Dudhia, J., Gill, J., Henderson, et. al, 2004. "The Weather Reseach and Forecast Model: Software Architecture and Performance," in the proceedings of the 11th ECMWF Workshop on the Use of High Performance Computing In Meteorology, 25-29 October 2004, Reading U.K. Ed. George Mozdzynski.

(Tolman, 1997): Tolman, H. L., 1997: User manual and system documentation of WAVEWATCH-III version 1.15. NOAA / NWS / NCEP /OMB Technical Note 151.

(Yang, 2004): Yang, M., McGrath, E.R., and Folk, M., 2004. Performance Study of HDF5-WRF IO modules, WRF Workshop.

(Nikolaou et al., 1997): Houstis, C., Nikolaou, C., MARAZAKIS, M., Patrikalakis, M.N., Sairamesh, J., and Thomas, A., 1997. THETIS: Design of a data management and data visualization system for coastal zone management of the Mediterranean Sea, D-lib Magazine.

(Kosar, 2004): Kosar, T., Livny, M., 2004. Stork: Making Data Placement a First Class Citizen in the Grid. In Proceedings of 24th IEEE Int. Conference on Distributed Computing Systems (ICDCS2004), Tokyo, Japan

(Allcock et al., 2004): Allcock, W.E., Foster, I., and Madduri, R., 2004. Reliable Data Transport: A Critical Service for the Grid. Building Service Based Grids Workshop, Global Grid Forum 11, June 2004.

(Singh et al., 2003): Singh, G., Bharathi, S., Chervenak, A., Deelman, E., Kesselman, C., Manohar, M., Patil, S., and Pearlman, L., 2003. A Metadata Catalog Service for Data Intensive Applications. SC 2003.

(Baru et al., 1998): Baru, C., Moore, R., Rajasekar, A., and Wan, M., 1998. The SDSC storage resource broker. In Proceedings of CASCON'98 Conference.

(Livny et al., 2002): Tannenbaum, T., Wright, D., Miller, K., and Livny, M., 2002. "Condor - A Distributed Job Scheduler", in Thomas Sterling, editor, Beowulf Cluster Computing with Linux, The MIT Press.

(Henderson, 1996): Henderson, R. and Tweten, D., 1996. Portable Batch System, Ames Research Center.

(Foster, 1997): Foster, I., and Keselman, C., 1997. Globus: A Metacomputing Infrastructure Toolkit. Intl J. Supercomputer Applications, 11(2):115-128, 1997.

(Deelman et al., 2005): Deeelman, E., Singh, G., Su, M., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, B.G., Good, J., Laity, Anastasia., Jacob, J., and Katz, S.D., 2005. "Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems". Scientific Programming Journal, Vol 13(3), 2005, Pages 219-237.

(Chervenak et al., 2005): Chervenak, R., Schuler, R., Kesselman, C., Koranda, S., and Moe., B., 2005. Wide Area Data Replication for Scientific Collaborations. Proceedings of 6th IEEE/ACM International Workshop on Grid Computing (Grid2005), November 2005.

(Wilde et al., 2004): Zhao, Y., Wilde, M., Foster, I., Voeckler, J., Dobson, J., Glibert, E., Jordan, and Quigg, E., 2004. Virtual Data Grid Middleware Services for Data-Intensive Science, August 2004. Concurrency, Practice and Experience.

(Kesselman et al., 2001): Czajkowski, K., Fitzgerald, S., Foster, I., and Kesselman, C., 2001. Grid Information Services for Distributed Resource Sharing. Proceedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press., August 2001.

(Couvares et al., 2006): Couvares, P., Kosar, T., Roy, A., Weber, J., and Wegner, K., 2006. "Workflow Management in Condor", Workflows for e-Science, Springer Press, 2006.

(Bogden et al., 2005): Bogden, P., Allen, G., Stone, G., Bintz, J., Graber, H., Graves, S., Luettich, R., Reed, D., Sheng, P., Wang, H., Zhao, W., 2005. "The Southeastern University Research Association Coastal Ocean Observing and Prediction Program: Integrating Marine Science and Information Technology, "Proceedings of the OCEANS 2005 MTS/IEEE Conference. Sept 18-23, 2005, Washington, D.C.

(Carlone, 1992): Carlone, R.V., 1992. "NASA's EOSDIS Development Approach", Technical report, United States General Accounting Office, February 1992.

(CCSP, 2003) CCSP, 2003. "Strategic Plan for the US Climate Change Science Program", *CCSP Report,* 2003.

(PetaShare, 2007) PetaShare, 2007. http://www.petashare.org