

# Runtime System for Extreme Scale Comparative Analytics on Spatio-Temporal Datasets

Tahsin Kurc and Joel Saltz  
Center for Comprehensive Informatics,  
Emory University

Phone: 404-712-9903, E-mail: tkurc@emory.edu

The objectives of this research are to address the following questions: “*What data structures and runtime methods are needed to carry out coordinated runtime optimizations to enable comparative analysis applications on extremely large low-dimensional, spatio-temporal scientific datasets?*” The runtime optimizations include 1) judicious management and staging of large and complex data structures across memory hierarchies -- from globally accessible disks, to local SSDs and disks, to distributed memory, to local memories, and to memories on GPUs, for instance; 2) distributing and rearranging computations and data to minimize data movement; 3) increasing locality by carefully caching and replicating portions of input datasets and output from analysis operations, 4) coordinated scheduling and mapping of analysis operations to CPU cores and GPUs to increase overall application throughput, and 5) masking data movement costs with computation. “*What programming abstractions are needed to easily compose comparative analyses while achieving high performance and scalability?*” Potential candidates for these abstractions are the MapReduce and filter-stream frameworks and combined use of these frameworks to express the range of data access and processing patterns in comparative analyses.

**Challenges Addressed:** This project aims to address the challenges associated with *programming* comparative analysis applications on extreme scale machines and *scaling* them to extremely large datasets. Scientific simulation applications on leadership scale machines generate very large volumes of data. Advanced instruments are also making it possible to rapidly collect vast quantities of high-resolution sensor data. Comparative analysis facilitates the process of studying, understanding, and quantifying how an ensemble of inter-related datasets (e.g., datasets from obtained from sensor readings and simulations over the same physical domain) differ and correlate. This type of analysis requires access to and processing of subsets of multiple datasets stored on disk as well as to data being generated on the fly (e.g., by multiple simulations running concurrently). Data access in common operations used in comparative analyses range from local and regular data access, to indexed data access, to irregular and global access to data. Processing patterns encapsulate several types of processing structures. Thus, runtime support for comparative analysis has to carefully coordinate query and retrieval of data from multiple sources, concurrent execution of multiple analysis methods with different processing structures, and management and movement of data across millions of cores, accelerators such as GPUs, and multiple memory hierarchies.

**Maturity:** The proposed research builds on and extends results from earlier research by our group as well as other groups. Tools and methods developed in the earlier research have targeted some of these challenges in distributed clusters, Grid and Cloud environments[1-8], or focused on specific types of functions such as high performance I/O[9-13], management and program coupling[14, 15], data preprocessing, indexing, and query[16-18], and file systems[19-22].

**Uniqueness:** Numerical simulations on extreme scale machines will be one main source of datasets for comparative analyses – for example, a researcher may run multiple large-scale simulations to look at various what-if scenarios. In addition, as advances in sensor technologies make it possible to capture data faster and at higher resolutions, the sizes of sensor datasets will rapidly reach extreme scales. These trends

will necessitate the use of extreme scale machines for data storage and analysis for scientific progress and discovery.

**Novelty:** Traditionally most of the research in runtime systems on leadership scale machines has focused on supporting complex numerical simulations. Runtime support for data intensive applications and data analysis has primarily targeted optimizing I/O operations for writing and reading large volumes of simulation output and managing, indexing, analyzing, and visualizing single datasets. The proposed approach complements these efforts in that it targets analyses on ensembles of interrelated datasets. It also aims to provide support for implementation of analytics applications from elementary operation categories that are common to comparative analyses on low-dimensional, spatio-temporal datasets. These elementary operation categories are data cleaning and low level transformations; data subsetting, filtering, subsampling; spatio-temporal mapping and registration; segmentation and object classification; multi-dimensional aggregation; and change detection, comparison, and quantification. The runtime environment is a component-based and service-oriented platform. Application specific implementations of the elementary operation categories are represented as components. These components are combined to form analysis pipelines using an abstraction combining MapReduce[8] and filter-stream processing[1] styles. Services are associated with elementary operations as well as data storage, indexing, and I/O to provide runtime support, such as coordinated scheduling of I/O operations and application-specific implementations of the elementary operations.

**Applicability:** We define low-dimensional scientific datasets as those in which (1) data elements are defined at points in a multi-dimensional coordinate space with small number of dimensions and at multiple time steps and (2) a point is primarily connected to (or interacts with) points in its spatial neighborhood. Many sensor and simulation datasets have these properties: microscopy images in biomedicine; simulation and sensor data in weather and climate modeling; satellite data in large-scale monitoring and change analysis; seismic surveys and numerical simulations in reservoir characterization; and data from telescopes in astronomy. Comparative analytics plays key roles in many phases of scientific research, including validation of numerical models, parameter studies, error estimation, predictive modeling, and sensitivity studies. In validation, analysis and comparison of multiple datasets, generated from simulations and experimental measurements, can be used to quantify and evaluate how much numerical models and measurements differ or agree. In predictive modeling, comparative analysis of multiple datasets can assist scientists to, for example, examine co-occurrences of features across time and space under different configurations or initial conditions of the problem under study. Thus, results from the proposed research will have applicability in a wide range of scientific research scenarios and domains.

**Effort:** To develop an efficient and effective runtime system to provide the functionality outlined in this paper, research will need to be carried out in the following areas: (1) programming models and abstractions to facilitate the development of comparative analytics applications; (2) data structures and techniques for efficient representation, indexing, storage, and staging of data; (3) methods and data structures for optimized management and movement of data across memory hierarchies during analysis; (4) strategies for mapping and execution of data processing tasks in an environment with billions of cores and CPU-GPU nodes. The research effort should be performed by multi-disciplinary teams of computer science and application researchers to produce not only an efficient and scalable runtime system, but also one that can be easily integrated into scientific research projects.

## Bibliography

- [1] M. Beynon, T. Kurc, U. Catalyurek, C. Chang, A. Sussman, and J. Saltz, "Distributed Processing of Very Large Datasets with DataCutter," *Parallel Computing*, vol. 27, pp. 1457-2478, 2001.
- [2] H. Andrade, T. Kurc, A. Sussman, and J. Saltz, "Active Proxy-G: Optimizing the Query Execution Process in the Grid," in *Proceedings of the ACM/IEEE Supercomputing Conference (SC2002)*, Baltimore, MD: ACM Press/IEEE Computer Society Press, 2002.
- [3] S. Kumar, T. Kurc, V. Ratnakar, et al, "Parameterized Specification, Configuration and Execution of Data-Intensive Scientific Workflows," *Cluster Computing: the Journal of Networks, Software Tools and Applications*, Special Issue on High Performance Distributed Computing, pp. 315-333, 2010.
- [4] C. Chang, T. Kurc, A. Sussman, and J. Saltz, "Optimizing Retrieval and Processing of Multi-Dimensional Scientific Datasets," in *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS 2000)*, Cancun, Mexico, 2000.
- [5] E. Deelman, J. Blythe, Y. Gil, et al. "Pegasus: Mapping scientific workflows onto the grid," *Grid Computing*, vol. 3165, pp. 11-20, 2004.
- [6] I. Foster, J. Vockler, M. Wilde, and Y. Zhao, "Chimera: A virtual data system for representing, querying, and automating data derivation," *14th International Conference on Scientific and Statistical Database Management*, Proceedings, pp. 37-46, 2002.
- [7] T. Condie, N. Conway, P. Alvaro, J. Hellerstein, K. Elmeleegy, and R. Sears, "MapReduce Online," EECS Department, University of California, Berkeley, Technical Report: UCB/EECS-2009-136, 2009.
- [8] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Communications of the Acm*, vol. 51, pp. 107-113, Jan 2008.
- [9] P. M. Dickens and J. Logan, "A high performance implementation of MPI-IO for a Lustre file system environment," *Concurrency and Computation-Practice & Experience*, vol. 22, pp. 1433-1449, 2010.
- [10] H. Abbasi, J. Lofstead, F. Zheng, K. Schwan, M. Wolf, and S. Klasky, "Extending I/O through high performance data services," *CLUSTER*, pp. 1-10, 2009.
- [11] H. Abbasi, M. Wolf, G. Eisenhauer, S. Klasky, K. Schwan, and F. Zheng, "DataStager: scalable data staging services for petascale applications.," *Cluster Computing*, vol. 13, pp. 277-290, 2010.
- [12] J. Lofstead, F. Zheng, S. Klasky, and K. Schwan, "Adaptable, Metadata Rich IO Methods for Portable High Performance IO," *Proceedings of IPDPS'09*, May 25-29, Rome, Italy, 2009.
- [13] C. Docan, M. Parashar, and S. Klasky, "Enabling high-speed asynchronous data extraction and transfer using DART," *Concurrency and Computation-Practice & Experience*, vol. 22, pp. 1181-1204, 2010.
- [14] C. Docan, M. Parashar, and S. Klasky, "DataSpaces: An Interaction and Coordination Framework for Coupled Simulations Workflows," *the Proc. of 19th International Symposium on High Performance Distributed Computing (HPDC'10)*, June, 2010.
- [15] C. Docan, F. Zhang, M. Parashar, J. Cummings, N. Podhorszki, and S. Klasky, "Experiments with Memory-to-Memory Coupling for End-to-End Fusion Simulation Workflows," *the 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid'10)*, May, 2010.
- [16] F. Zheng, H. Abbasi, C. Docan, et al., "PreData - Preparatory Data Analytics on Peta-Scale Machines," *the 24th IEEE International Parallel and Distributed Processing Symposium (IPDPS'10)*, Atlanta, Georgia, April, 2010.
- [17] S. Narayanan, T. Kurc, U. Catalyurek, and J. Saltz, "Database Support for Data-driven Scientific Applications in the Grid," *Parallel Processing Letters*, vol. 13 pp. 245-273, 2003.
- [18] K. Wu, S. Ahern, E. W. Bethel, et al, "FastBit: interactively searching massive data," *Scidac 2009: Scientific Discovery through Advanced Computing*, vol. 180, 2009.
- [19] J. Piernas, J. Nieplocha, and E. J. Felix, "Evaluation of Active Storage Strategies for the Lustre Parallel File System," *2007 ACM/IEEE Supercomputing Conference*, pp. 240-249, 2007.
- [20] K. Shvachko, H. R. Kuang, S. Radia, and R. Chansler, "The Hadoop Distributed File System," *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, 2010.
- [21] P. A. Lopes and P. D. Medeiros, "pCFS vs. PVFS: Comparing a Highly-Available Symmetrical Parallel Cluster File System with an Asymmetrical Parallel File System," *Euro-Par 2010 Parallel Processing*, Pt I, vol. 6271, pp. 131-142, 2010.
- [22] P. Schwan, "Lustre: Building a file system for 1000-node clusters.," *The 2003 Linux Symposium*, 2003.