# Towards Dynamic Data-Driven Management of the Ruby Gulch Waste Repository*

Manish Parashar[1], Vincent Matossian[1], Hector Klie[2], Sunil G. Thomas[2],
Mary F. Wheeler[2], Tahsin Kurc[3], Joel Saltz[3], and Roelof Versteeg[4]

[1] TASSL, Dept. of Electrical & Computer Engineering, Rutgers,
The State University of New Jersey, New Jersey, USA
{parashar, vincentm}@caip.rutgers.edu
[2] CSM, ICES, The University of Texas at Austin, Texas, USA
{klie, sgthomas, mfw}@ices.utexas.edu
[3] Dept. of Biomedical Informatics, The Ohio State University, Ohio, USA
{kurc, jsaltz}@bmi.osu.edu
[4] INL, Idaho, USA
roelof.versteeg@inl.gov

**Abstract.** Previous work in the Instrumented Oil-Field DDDAS project has enabled a new generation of data-driven, interactive and dynamically adaptive strategies for subsurface characterization and oil reservoir management. This work has led to the implementation of advanced multi-physics, multi-scale, and multi-block numerical models and an autonomic software stack for DDDAS applications. The stack implements a Grid-based adaptive execution engine, distributed data management services for real-time data access, exploration, and coupling, and self-managing middleware services for seamless discovery and composition of components, services, and data on the Grid. This paper investigates how these solutions can be leveraged and applied to address another DDDAS application of strategic importance - the data-driven management of Ruby Gulch Waste Repository.

## 1 Introduction

The dynamic, data driven application systems (DDDAS) paradigm is enabling a new generation of end-to-end multidisciplinary applications that are based on seamless aggregation of and interactions between computations, resources, and data. An important class of DDDAS applications include simulations of complex physical phenomena that symbiotically and opportunistically combine

* The research presented in this paper is supported in part by the National Science Foundation Grants ACI-9984357, EIA-0103674, EIA-0120934, ANI-0335244, CNS- 0305495, CNS-0426354, IIS-0430826, ACI-9619020 (UC Subcontract 10152408), ANI-0330612, EIA-0121177, SBR-9873326, EIA-0121523, ACI-0203846, ACI-0130437, CCF-0342615, CNS- 0406386, CNS-0426241, ACI-9982087, CNS-0305495, NPACI 10181410, Lawrence Livermore National Laboratory under Grant B517095 (UC Subcontract 10184497), Ohio Board of Regents BRTTC BRTT02-0003, and DOE DE-FG03-99ER2537.

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part III, LNCS 3993, pp. 384–392, 2006.

computations, experiments, observations, and real-time data to provide important insights into complex systems.

In a recent DDDAS project, we have developed several key technologies to enable a new generation of data-driven, interactive and dynamically adaptive strategies for subsurface characterization and reservoir management. This "Instrumented Oil-Field" project [1, 2, 3] aimed at completing the symbiotic feedback loop between measured data and the computational models to provide more efficient, cost-effective and environmentally safer production of oil reservoirs, which can result in enormous strategic and economic benefits. Our work in this project has led to conceptual and infrastructure solutions [1, 2, 3, 4, 5, 6, 7], which include advanced multi-physics, multi-scale and multi-block numerical models as well as an autonomic DDDAS software stack. The software stack provides a middleware for autonomic DDDAS applications and consists of a Grid-based execution engine that supports self-optimizing, dynamically adaptive applications, distributed data management services for large scale data management and processing, and self-managing middleware services for seamless discovery, access, interactions and compositions of components, service and data on the Grid.

In this work, we look at the application of these technologies in the more general class of knowledge-based, data-driven subsurface management applications. This paper is concerned with the problem of dynamic data-driven waste management at the Ruby Gulch Waste Repository, and more specifically, the Gilt-Edge site. It investigates how the DDDAS models, methodologies and software stack can be applied to address this challenging application.

## 2 The Dynamic Data-Driven Waste Management Problem: The Ruby Gulch Waste Repository

The Gilt Edge Mine is located near Deadwood, South Dakota. Mining for gold and silver at this site occurred from 1880-1999. In 1999 the Dakota Mining Corporation (which operated the mine through its subsidiary, Brohm Mining Company) declared bankruptcy, and the site reverted to the state of South Dakota. Mining activities had resulted in several negative environmental impacts on the site. One of the main environmental issues was the presence of multiple sources of ARD (Acid Rock Drainage). The primary source was the Ruby Waste Rock Repository. This repository is a valley in which mine rock was disposed of post leaching. It contains approximate 11 million cubic yard of waste rock. As ARD

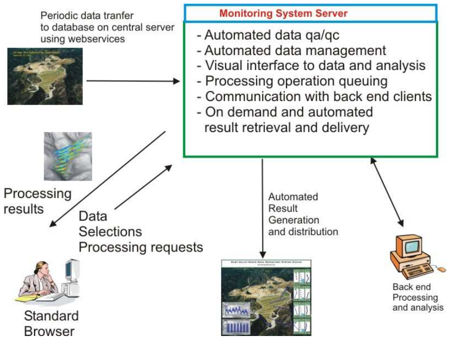

**Fig. 1.** Gilt Edge Monitoring System Server

flowing from this repository would severely impact drinking water quality downstream of the site, this ARD needs to be captured and treated.

In order to minimize the amount of water coming out of this repository a ROD (Record of Decision) for this repository called for the emplacement of a cap over the site. As EPA had interest in the monitoring of the performance of this cap, a monitoring system was designed and installed by scientists from the Idaho National Laboratory (INL) [8, 9]. The monitoring system autonomously collects continuous data using the following sensors: (1) a weather station operated by SDENR (South Dakota Department of Environment and Natural Resources); (2) an outflow meter at the bottom of the Ruby Repository; (3) temperature sensors in four well boreholes; (4) advanced tensiometers are located within boreholes and measure matrix potential (related to water saturation); (5) a multi-electrode resistivity system.

In addition to the autonomously collected data, gas-ports and porous-ceramic cup-lysimeters in the wells are sampled monthly (starting in June 2004). Gas analysis is done in the field for CO2 and O2. Water samples are sent to a laboratory for analysis. In addition to this data there is a substantial amount of historical data (collected by BMC and DENR), which has been aggregated into the database. Figure 1 illustrates the monitoring process at the Gilt Edge mine.

## 3 Models, Methods and Middleware for Data-Driven Subsurface Management Applications

### 3.1 The Integrated Parallel Accurate Reservoir Simulator (IPARS)

IPARS represents a new approach to parallel reservoir simulator development, emphasizing modularity, code portability to many platforms, ease of integration and inter-operability with other software. It provides a set of computational features such as memory management for general geometric grids, portable parallel communication, state-of-the-art non-linear and linear solvers, keyword input, and output for visualization. A key feature of IPARS is that it allows the definition of different numerical, physical, and scale models for different blocks in the domain (i.e., multi-numeric, multi-physics, and multi-scale capabilities). A more technical description of IPARS and its applications can be found in [10].

### 3.2 Optimization Algorithms

Novel stochastic optimization algorithms [4, 5] have been included in the framework, namely, the Very Fast Simulated Annealing (VFSA), the Finite Difference Stochastic Approximation (FDSA) and the Simultaneous Perturbation Stochastic Approximation (SPSA). The VFSA and SPSA, in particular, have potentials for large scale implementations. The critical issue here, is that parameter estimation may involve several hundreds of thousands of variables and the objective function evaluation is a highly demanding process since it involves a full simulation run. This also rules out the possibility for using gradient-based methods, especially in a multi-model environment that systematically aims at different physics and algorithms for which sensitivity coefficients are not trivial either to compute or reformulate.

### 3.3 Storage and Management of Large Volumes of Data

Effective solutions to the waste management problem targeted in this work will require gleaning and extracting information from results of optimization processes using complex numerical models and from data gathered by field sensors. Datasets generated by an optimization run consists of the values of the input and output parameters along with the output from simulations of a numerical model of the physical domain. Datasets gathered from field sensors consist of readings obtained from each sensor, the location of the sensor, and the date of the reading. This information provides a dynamically updated (as more readings are obtained) historical record of the field under study. By maintaining simulation and sensor datasets, a large-scale dynamic knowledge base can be created. This knowledge base can be used to speed up the execution of optimization runs, to carry out post-optimization analyses, to refine numerical models using field data, and to control where and how much field data should be collected, thus implementing a dynamic, data-driven application system approach. Common types of queries against these datasets include computing data subsets via range queries, aggregations such as counts, averages on over regions of meshes, and differences between regions of interest on multi-resolution datasets. With the help of inexpensive disk-based storage, we are seeing the emergence of large scale, hierarchical storage platforms with varying capacity/bandwidth (from larger, slower disk pools to smaller, faster disks to memory on cluster) and distance from compute nodes. A challenging issue is to be able to use different levels of storage and computing hierarchy in a coordinated way to maximize the bandwidth of data retrieval and processing. A number of strategies, such as multi-level hierarchical indexing [11], partial replication [12], caching [13], and adaptive data redistribution can be employed.

To support the knowledge base, we make use of three middleware systems to support the data management and processing requirements as described above of optimization based studies for waste management application. STORM [14] is a service-oriented middleware that supports data select and data transfer operations on scientific datasets, stored in distributed, flat files, through an object-relational database model. Mobius [15] provides support for management of metadata definitions, on-demand database creation, and federation of distributed XML databases. It uses XML schemas to define the structure of data elements and XML documents to represent instances of data elements. DataCutter [16] provides a coarse-grained data flow system and allows combined use of task- and data-parallelism. In DataCutter, application processing structure is implemented as a set of components, called *filters*, that exchange data through a *stream* abstraction.

### 3.4 Autonomic Grid Middleware Substrate

Emerging knowledge-based and dynamic data-driven subsurface management and control applications, such as the applications described in this paper, combine computations, experiments, observations, and real-time data, and are highly heterogeneous and dynamic in their scales, behaviors, couplings and interactions. AutoMate [17], an Autonomic Computational Engine for management and control, investigates conceptual models and implementation architectures to address these challenges and enable the development and execution of such self-managing Grid applications. Key components include:

- The *Seine/MACE Computational Engine* [18] that implements a dynamic geometry-based shared space interaction model to support the dynamic and complex communication and coordination patterns required by the multiblock parallel multi-block simulations.
- The *Accord Programming System* [19] that enables the definition of autonomic components and the dynamic composition, management and optimization of these components using externally defined rules and constraints. Autonomic components in *Accord* export sensors and actuators for external monitoring, control and adaptation.
- The *Autonomic Runtime Environment* [20] provides policies and mechanisms for both "system sensitive" and "application sensitive" runtime adaptations to manage the heterogeneity and dynamism of the applications as well as Grid environments.
- The *Content-based Grid Middleware Substrate* [17] that supports autonomic application behaviors and interactions, and to enable simulation components, sensors/actuators, data archives and Grid resources and services to seamlessly interact as peers. Key components of the middleware include the Meteor, a decentralized infrastructure for decoupled associative interactions, the Squid content-based routing engine and decentralized information discovery service, and the Pawn peer-to-peer messaging substrate.
- The *Discover Collaboratory* [21] that provides collaborative problem solving environment and enables geographically distributed scientists and engineers to collaboratively monitor, interact with, and control high performance applications in a truly pervasive manner using portals.

## 4 Dynamic Data-Driven Waste Management: Modeling the Gilt-Edge Site with Dynamic Data

In order to make better predictions from the measurements at the Gilt Edge site, the first task is to develop a model that suitably explains the observations from the experiments at the waste repository. For instance, it was observed that there exists a diurnal/seasonal variation in the outflow measurements. Using IPARS, a system of modified air-water equations can be used to model the problem. This solution takes into account that water can exist in the air phase as vapor, and can explain the diurnal variations qualitatively. An example water pressure profile reproduced by the model is shown in Figure2.

Once the predicted model is calibrated, the next challenge is to determine the physical parameters of the site such as permeability, porosity and capillary pressure in order to reproduce the exact measured outflows at the site. This task can be modeled as a parameter estimation problem using the numerical model of the environment. This is where an efficient optimization method such as SPSA or VFSA plays a significant role. At present, the SPSA method is being implemented on the hydrology model in IPARS. Using the autonomic computational engine, the execution of IPARS and the optimization methods can be dynamically orchestrated in a Grid environment [1, 2].

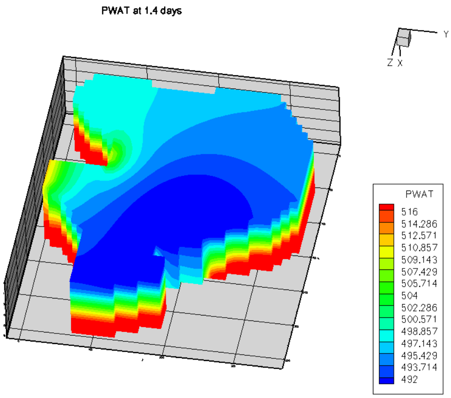

**Fig. 2.** Example of an air-water simulation to predict water pressure profile

To perform parameter estimation a mismatch function based on the difference of measured and calculated outflow of water at a specified location at the site is posed. Similarly, an objective function based on maximum cleanup rate can be designed for optimal site management. To extend these optimizations for data assimilation, the numerical models implemented using IPARS would need to be refined using dynamic data from outflow measurements at the site. To achieve this, an optimal control scheme should be formulated to accommodate dynamic changes into the parameters (i.e., properties) and state variables (e.g., saturations, pressures, temperatures) of the model. Increasing understanding of the physical model and the need to respond more quickly to observations leads to metamodels (surrogate models) or reduced models. These simpler models mimic the behavior of the original predictive model given by IPARS.

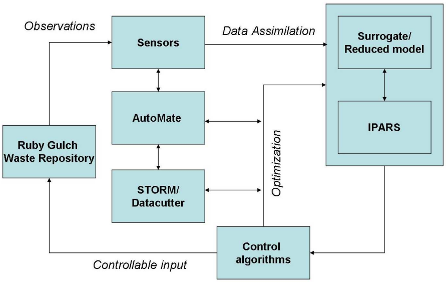

**Fig. 3.** The Dynamic Data-Driven Waste Management framework

Accurate model prediction and optimization capabilities in conjunction with the Grid middleware and data management tools described in Section 3 makeup the fundamental components of the The Dynamic Data-Driven Waste Management framework proposed in this work (see Figure 3). The framework adds autonomic descision making and control capabilities to the monitoring process.

The autonomic computational engine and middleware service provide the infrastructure for: (1) enabling the efficient, large scale, dynamically adaptive multi-block IPARS simulations; (2) for discovering, aggregating and assimilating data from the sensors at the remote Gilt-Edge site and dynamically injecting it into the simulation processes as required; (3) selecting and invoking appropriate optimization services; (4) enabling dynamic composition of services to realize data driven workflows on the Grid; and (4) enabling remote collaborative and interactive access to the simulations and the data using pervasive portals.

The estimation of physical properties of the environment involves search of a parameter space and requires data from outflow measurements dynamically drive the simulation of the numerical models and parameter space search. A knowledge base can be created from the datasets that are generated (via simulations or field measurements) and referenced in this application to speed up the optimization process. Mobius can be used to manage metadata associated with distributed datasets in the knowledge base. At any given step during optimization the knowledge base can be queried to see if a given step, or a subset of numerical simulations at that step, has been already evaluated. STORM and DataCutter can be employed to support queries into distributed collections of large datasets stored as a collection of files. For instance, in post-optimization analyses, a user may want to compare and correlate a subset of results obtained from one optimization run with results from another set of optimization runs.

## 5 Conclusion

Our previous work in the Instrumented Oil-Field DDDAS project has developed advanced numerical models and a suite of software components, which provide support for 1) execution of dynamically adaptive applications in a Grid environment, 2) management and manipulation of large distributed datasets, and 3) seamless discovery of, interaction with, and composition of application components, services, and data in the Grid. We believe that these advanced numerical models and tools can be applied to other types of data-driven subsurface management applications. In this paper we investigated how these technologies can be leveraged to enable data-driven management of Ruby Gulch Waste Repository.

## References

1. Parashar, M., Klie, H., Catalyurek, U., Kurc, T., Matossian, V., Saltz, J., Wheeler, M.: Application of grid-enabled technologies for solving optimization problems in data-driven reservoir studies. In: Proceedings of the Workshop on Distributed Data Driven Applications and Systems,International Conference on Computational Science 2004 (ICCS 2004). Volume 3038., Krakow,Poland (2004) 805 – 812
2. Parashar, M., Matossian, V., Bangerth, W., Klie, H., Rutt, B., Kurc, T., Catalyurek, U., Saltz, J., Wheeler, M.: Towards dynamic data-driven optimization of oil well placement. In: Proceedings of the Workshop on Distributed Data Driven Applications and Systems, International Conference on Computational Science 2005 (ICCS 2005). Volume 3514-3516., Atlanta, USA (2005) 656 – 663

3. Klie, H., Bangerth, W., Gai, X., Wheeler, M.F., Stoffa, P., Sen, M., Parashar, M., Catalyurek, U., Saltz, J., Kurc, T.: Models, methods and middleware for grid-enabled multiphysics oil reservoir management. Engineering with Computers, Springer-Verlag (2006)
4. Matossian, V., Bhat, V., Parashar, M., Peszynska, M., Sen, M., Stoffa, P., Wheeler, M.F.: Autonomic oil reservoir optimization on the grid. Concurrency and Computation: Practice and Experience **17** (2005) 1–26
5. Bangerth, W., Klie, H., Matossian, V., Parashar, M., Wheeler, M.F.: An autonomic reservoir framework for the stochastic optimization of well placement. Cluster Computing: The Journal of Networks, Software Tools, and Applications **8** (2005) 255–269
6. Kurc, T., Catalyurek, U., Zhang, X., Saltz, J., Martino, R., Wheeler, M., Peszyńska, M., Sussman, A., Hansen, C., Sen, M., Seifoullaev, R., Stoffa, P., Torres-Verdin, C., Parashar, M.: A simulation and data analysis system for large scale,data-driven oil reservoir simulation studies. Concurrency and Computation: Practice and Experience. **17** (2005) 1441–1467
7. Parashar, M., Muralidhar, R., Lee, W., Wheeler, M., Arnold, D., Dongarra, J.: Enabling interactive and collaborative oil reservoir simulations on the grid. Concurrency and Computation: Practice and Experience **17** (2005) 1387–1414
8. Versteeg, R., Wangerud, K., et al.: Managing a capped acid rock drainage (ard) repository using semi-autonomous monitoring and modeling. In: ICARD 2006, St. Louis, Missouri (2006)
9. Wangerud, K., Versteeg, R., et al.: Insights into hydrodynamic and geochemical processes in a valley-fill ard waste-rock repository from an autonomous multi-sensor monitoring system. In: ICARD 2006, St. Louis, Missouri (2006)
10. (Ipars: Integrated parallel reservoir simulator) The University of Texas at Austin, http://www.ices.utexas.edu/CSM.
11. Zhang, X., Pan, T., Catalyurek, U., Kurc, T., Saltz, J.: Serving queries to multi-resolution datasets on disk-based storage clusters. In: Proceedings of 4th IEEE/ACM International Symposium on Cluster Computing and the Grid (CC-Grid2004), Chicago, IL (2004)
12. Weng, L., Catalyurek, U., Kurc, T., Agrawal, G., Saltz, J.: Servicing range queries on multidimensional datasets with partial replicas. In: Proceedings of the 5th IEEE/ACM International Symposium on Cluster Computing and the Grid (CC-Grid 2005). (2005)
13. Deshpande, P.M., Ramasama, K., Shukla, A., Naughton, J.F.: Caching multidimensional queries using chunks. In: ACM SIGMOD Record, Vol. 27, No. 2. (1998) 259–270
14. Narayanan, S., Kurc, T., Catalyurek, U., Zhang, X., Saltz, J.: Applying database support for large scale data driven science in distributed environments. In: Proceedings of the Fourth International Workshop on Grid Computing (Grid 2003), Phoenix, Arizona (2003) 141–148
15. Hastings, S., Langella, S., Oster, S., Saltz, J.: Distributed data management and integration: The mobius project. In: GGF Semantic Grid Workshop 2004, GGF (2004) 20–38
16. Beynon, M.D., Kurc, T., Catalyurek, U., Chang, C., Sussman, A., Saltz, J.: Distributed processing of very large datasets with DataCutter. Parallel Computing **27** (2001) 1457–1478

17. Parashar, M., Liu, H., Li, Z., Matossian, V., Schmidt, C., Zhang, G., Hariri, S.: Automate: Enabling autonomic grid applications. Cluster Computing: The Journal of Networks, Software Tools, and Applications, Special Issue on Autonomic Computing **9** (2006)
18. Zhang, L., Parashar, M.: Seine: A dynamic geometry-based shared space interaction framework for parallel scientific applications. Concurrency and Computations: Practice and Experience (2006)
19. Liu, H., Parashar, M.: Accord: A programming framework for autonomic applications. IEEE Transactions on Systems, Man and Cybernetics, Special Issue on Engineering Autonomic Systems (2006)
20. Chandra, S., Parashar, M., Yang, J., Zhang, Y., Hariri, S.: Investigating autonomic runtime management strategies for samr applications. International Journal of Parallel Programming **33** (2005) 247–259
21. Mann, V., Parashar, M.: DISCOVER: A computational collaboratory for interactive grid applications. In Berman, F., Fox, G., Hey, T., eds.: Grid Computing: Making the Global Infrastructure a Reality, John Wiley and Sons (2003) 727–744