

ResGrid: A Grid-aware Toolkit for Reservoir Uncertainty Analysis

Zhou Lei, Dayong Huang, Archit Kulshrestha, Santiago Pena, Gabrielle Allen
Center for Computation & Technology, Louisiana State University, U.S.A.
Xin Li, Richard Duff, Subhash Kalla, Chris D. White, John R. Smith
Department of Petroleum Engineering, Louisiana State University, U.S.A.
{zlei, xli, dayong, duff, architk, santiago, gallen}@cct.lsu.edu
{kalla2, cdwhite, jsmith5}@lsu.edu

Abstract

Many efforts in Grid communities have focused on middleware research and development. However, Grid application-level tools are needed which can build higher-level functionality on top of core middleware services. We work with specific classes of scientific applications and present a Grid-aware toolkit ResGrid for reservoir uncertainty analysis. With the help of the ResGrid, a reservoir engineer can transparently take advantage of Grid resources and services for compute-intensive and data-intensive uncertainty analysis as well as enforce the understanding of multiphase reservoir modeling. This paper explains a typical reservoir uncertainty analysis scenario and evaluates the current limitations a reservoir engineer faces. The ResGrid is introduced in terms of overview, architecture, implementation status. In the case studies, the design and implementation of the ResGrid are verified via a reservoir uncertainty analysis process on the CCT Grid testbed. The ResGrid releases the unbound capabilities of the Grid and improves the efficiency for reservoir researchers. The ResGrid is also utilized by other application areas, such as coastal modeling.

1. Introduction

Reservoir studies are targeted at obtaining accurate assessments and predictions of reservoir performance, for the purpose of guiding development and operational decisions. Through such studies, engineers strive to forecast the results and consequences of different development and production scenarios. However, accurately predicting the reservoir performance is a challenging issue due to the many different sources of uncertainty which can seriously impact reservoir performance. Uncertainty analysis is critical for conducting reservoir performance prediction. Ideally, one would like to perform an exhaustive evaluation of reservoir simulations with uncertainty factors to char-

acterize the impacts, but such an undertaking is computationally infeasible. Therefore, people pursue two different approaches to address and improve reservoir uncertainty analysis. One is to develop optimization algorithms to minimize the search space for the “most plausible” sets of model parameters [17]. The other is to push the limits of the latest computational technologies to provide large-scale data and computing capabilities for massive simulation executions. This paper demonstrates our contribution towards the latter approach.

Grid computing technologies provide tools for coordinated resource sharing to support distributed, dynamic, and heterogeneous virtual organizations. Grid computing is an active area of research, which holds great potential promise for large-scale science and engineering applications. Around the world, there have been many efforts to research and develop the core middleware to support Grid communities. Technologies such as the Globus Toolkit [7], Grid Security Infrastructure (GSI) [8], GridSphere [13], and Condor [21], have been (or are becoming) the de facto standards in the Grid computing area. Since the issue on the lack of application-level Grid programming abstraction was recognized by computational scientists [1], Grid Application Toolkit (GAT) [2] has been implemented and Simple API for Grid Application (SAGA) [19] has been proposed. The GAT is a high-level application programming toolkit. It provides a unified simple programming interface for the Grid infrastructure, tailored to the needs of Grid application programmers and users. The GAT implementation handles both the complexity and the variety of existing Grid middleware services via so-called adaptors. The SAGA is the standardization effort for Grid application programming abstraction, pursued through the Global Grid Forum Research Group on Programming Models. The GAT is one of the reference implementations of SAGA. GAT/SAGA bridged the gap between the existing Grid middleware and application needs.

Although Grid application programming abstraction

significantly improves and simplifies the usage of the Grid, the scientists from specific application areas still expect the ultimate solutions for their applications. According to the recent Grid user survey taken by Argonne National Laboratory [16], one of the chief complaints is that many Grid tools offered horizontal functionality but no end-to-end solutions to a given problem. Therefore, the development of application-oriented toolkit, which integrates the Grid application level technologies, such as the programming abstraction (e.g., GAT/SAGA and GridRPC [18]) and Grid portal (e.g., GridSphere), is drawing more and more attentions of Grid communities to solve one specific problem.

Our work focuses on the design and development of an integrated problem solving environment (PSE) for reservoir uncertainty analysis, leveraging Grid computing technologies and contemporary simulation software. Uncertainty analysis with reservoir simulations are critical for forecasting reservoir performance. In modeling a reservoir, there are many different sources of uncertainty, such as structural/stratigraphic permeability architecture, fluid properties, drive mechanisms, and spacing/orientation of producing and injecting wells. Uncertainties of these factors lead to uncertainty of reservoir performance prediction. Therefore, a reservoir engineer adopts uncertainty analysis/sensitivity study to measure and predict performance. In an uncertainty analysis process, various combinations of uncertainty factors and factor levels are assessed to construct diverse models for reservoir simulations, and simulation results are analyzed to estimate the sensitivity issues. The resulting Grid-aware toolkit *ResGrid* enables a reservoir engineer to easily take advantage of Grid resources and services for uncertainty studies, and enforce the understanding of reservoir modeling. The design and implementation of *ResGrid* are outlined in this paper.

This paper is organized as follows. In Section 2, we provide an introduction to reservoir uncertainty analysis and describe a typical usecase scenario along with its current limitations. In Section 3, we describe the architecture, implementation and case studies for the *ResGrid* software. Section 4 discusses related work, and finally Section 5 provides conclusions and details of future work.

2. Reservoir Uncertainty Analysis

2.1. Problem Description

Although technological advances have improved significantly in contemporary petroleum exploration and development, risk has not been reduced in all cases. For instance, high costs associated with platform design and well construction for deepwater projects lead to large initial capital investments being made with only limited knowledge of

reservoir architecture and geology. Prior to investments, petroleum exploration and production engineers need be able to identify the reservoir characteristics and various uncertainty factors, and then quantify and analyze these uncertainty factors in the data acquisition program.

Reservoir simulation [3] is the main approach for characterizing a reservoir in the planning and evaluation of sequential development phases. A reservoir can be represented by a mathematical model by applying the mass conservative law (i.e., Darcy's law), relative permeability and capillary pressure relationship in a differential equation:

$$\nabla \cdot (\rho_m K \lambda_m \nabla P_m) - q_m = \frac{\partial (\phi \rho_m S_m)}{\partial t}$$

where m = oil, water, or gas; ρ_m = density; K = permeability; λ_m = mobility; P_m = pressure; q_m = production rate; ϕ = porosity; S_m = saturation; and t = time. To obtain an analytical solution of a reservoir, numerical simulation is required. A reservoir simulation consists of the following steps: 1) Geologists build a representative geological model using seismic, well logging and other geological data. 2) Geostatistical realizations are generated to sample the uncertainty of geological parameters. 3) Reservoir engineers combine geology, fluid and flow parameters, along with well locations and other engineering factors to constitute a base model. 4) This model is simulated to obtain production profiles and recovery factors for a chosen recovery process. 5) Economic performance indicators, such as ROI (Return on Investment) and NPV (Net Present Value), are calculated.

As uncertainty analysis and sensitivity studies play a key role in reservoir performance prediction, experimental design and response surface methodology [22] provide mechanisms to assess uncertainty by providing inference with a number of reservoir simulations, as well as to quantify the influences on production and economic forecasts. A *design* is a set of factor-value (varied parameters) combinations for which responses are modeled. More than two levels (not just low value and high value) of each factor must be considered for a non-linear oil and gas reservoir response. A response surface model associated with a combination of uncertainty factors and factor levels is an empirical fit of reservoir simulation results as follows:

$$\begin{aligned} \hat{y}_j(\vec{x}) = & \hat{\beta}_{j,1} + \sum_{i=1}^k \hat{\beta}_{j,i+1} x_i + \hat{\beta}_{j,i+k+1} x_1 x_2 + \hat{\beta}_{j,k+2} x_1 x_3 \\ & + \dots + \hat{\beta}_{j,1+k(k+1)/2} x_{k-1} x_k + \sum_{i=1}^k \hat{\beta}_{j,k+1+k(k+1)/2} x_i^2 \end{aligned}$$

where y = responses; x = uncertainty factors; k = the number of uncertainty factors; and β = regressors. The number of simulation runs involved in these kinds of factorial designs is very large, particularly if many uncertainty factors

are considered, motivating the improvement of both computation technologies and optimization studies.

2.2. Typical Uncertainty Analysis Scenario

Reservoir engineers have adopted various hardware platforms and software packages to perform uncertainty analysis with the experimental design and response surface methodology for reservoir studies. Used hardware ranges from personal computers to high performance clusters. Software includes diverse open source or commercial reservoir simulators, geostatistics toolkits, visualization tools, etc. Using reservoir modeling and simulation runs with different combinations of uncertainty factors and factor levels, the sensitivity of each uncertainty factor can be identified. There is no integrated, secure, and ease-to-use problem solving environment available for use although some efforts [23] have been made. A reservoir engineer needs to manually make these toolkits work together.

Figure 1 demonstrates a typical reservoir uncertainty analysis scenario. First, a reservoir engineer specifies the uncertainty factors and these factor levels. Once the reservoir models associated with the combinations of uncertainty factors and levels are constructed, the user submits them to a high performance computing facility (e.g., a cluster) for multiple simulation runs. A batch file or the job scheduler on the facility invokes multiple executions of geostatistics algorithms and a reservoir simulator. Simulation results are then moved to a data storage device. Post processing software is used to view and analyze the results. To do so, the user need locate the data files containing the appropriate results in the data storage device, and move files to a local machine using data transfer tools, such as FTP. Typically, analysis and visualization of results make use of Windows-based commercial software packages.

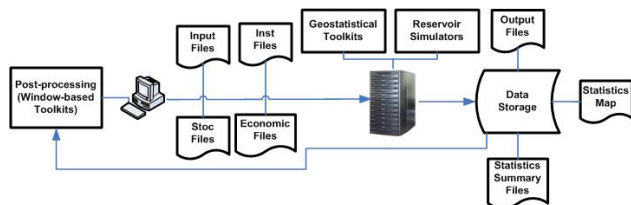


Figure 1. Workflow involved in a typical reservoir uncertainty analysis scenario

2.3. Limitations

Limitations restrain advanced reservoir uncertainty analysis. A single high performance computing facility cannot satisfy the requirements of massive reservoir sim-

ulation runs. Large-scale data storage is required for both modeling-related data and simulation results. There is no ease-to-use integrated environment for uncertainty analysis. Security issues hinder effective collaborations between researchers interested in reservoir studies.

Response surface and experimental design methods are frequently used for uncertainty study of complex reservoir systems, which are computing-intensive and data-intensive processes. Let us take an example. These methods are applied to a single-well water-drive gas reservoir with a radial geometry [10]. Fourteen factors are considered: eleven geologic factors (initial pressure, horizontal permeability, connate water saturation, critical gas saturation, gas end point, water end point water Corey exponent, gas Corey exponent, non-Darcy coefficient, aquifer size, anisotropy ratio) and three engineering factors (completion length ratio, tubing head pressure, tubing diameter). The simulation runs for full factorial design would be $4^6 \times 3^8 = 26,873,856$ if there are six factors each of which has four levels and eight ones each of which has three levels. Conservatively assumed a single simulation run with a grid-block size of 50 feet for a middle scale reservoir consumes 6 minutes CPU time, the total execution time would be 2,687,386 hours (or over 100 days on a 1024 processor cluster). Meanwhile, large-scale data are involved in such a study. Geological&geophysical (G&G) data and well logging data are geographically distributed, which size scale is terabytes, even petabytes. The average result dataset of a single simulation reaches up to 50 Megabytes. Massive simulations lead to storage needs which cannot easily accommodated with a typical storage resource.

The lack of an integrated solving environment is another issue that limits studies on reservoir uncertainty analysis. A reservoir engineer needs to manually handle all stages of the process, including provisioning, staging, visualization, result retrieval, and sensitivity analysis.

Security concerns make it difficult to form effective collaborations among the reservoir study communities. The exploration and production data sets are very sensitive due to the potential commercial benefits.

To conduct an uncertainty analysis process, a reservoir engineer needs to minimize the number of uncertainty factors and the factor levels, which may often cause the loss of correct conclusions.

3. ResGrid: A Grid-aware Toolkit for Reservoir Uncertainty Analysis

3.1. Overview

Using Grid computing technologies, along with portal technologies, the ResGrid addresses four concerns: (i) how

to integrate large-scale model-related data with uncertainty analysis; (ii) how to provide supercomputing capabilities for massive reservoir simulations; (iii) how to allow reservoir engineers to concentrate on uncertainty analysis rather than computing resource management; and (iv) how to deal with security issues across a Grid.

The ResGrid provides a reservoir engineer with a PSE for uncertainty studies. It allows a user conveniently to collect G&G data and well logging data, specify the uncertainty parameter space, invoke numerical reservoir simulations across the Grid computing resources, then analyze and visualize simulation results. All these operations are completed via a Grid portal, interacting with various Grid services and resources. Load balancing strategies make sure load sharing in a Grid. GSI ensures high security among all the processes and data transfer. In brief, the ResGrid provides an integrated, easy-to-use and secure PSE for reservoir uncertainty analysis.

The ResGrid use case scenario is illustrated in Figure 2. Typically, there are ten steps :

1. A user logs into the ResGrid portal and retrieves a GSI certificate from a proxy server. The certificate authorizes the user to access the Grid resources and implement secure data transfer.
2. The user specifies the uncertainty factor parameter space and the size of reservoir grid block, which will be used for reservoir model construction and result analysis.
3. By clicking on the “Launch” button, the user invokes the execution of the ResGrid services.
4. The first service is reservoir modeling. This triggers a data archiving tool and analyzes the uncertainty factor parameter space specified by the user in Step 2.
5. The modeling service constructs reservoir models and starts the Grid resource brokering service.
6. The resource brokering service captures the dynamic information from the information service provided by the Grid, decides the appropriate resource for each single simulation run with the help of load balancing strategies, and then calls the massive simulation execution service.
7. The simulation executions are invoked to all the resources available on the Grid.
8. Once all the simulation runs have been finished, the sensitivity analysis service is activated to analyze the simulation results.
9. The visualization service visualizes the simulation results from the sensitivity analysis.
10. The user views on the ResGrid portal the results generated by the visualization service.

What a reservoir engineer needs to do in this scenario is only to interact with a Web-based Grid portal for uncertainty analysis. The ResGrid services take care of security issue, data acquisition, resource management complexities, result analysis and visualization, etc.

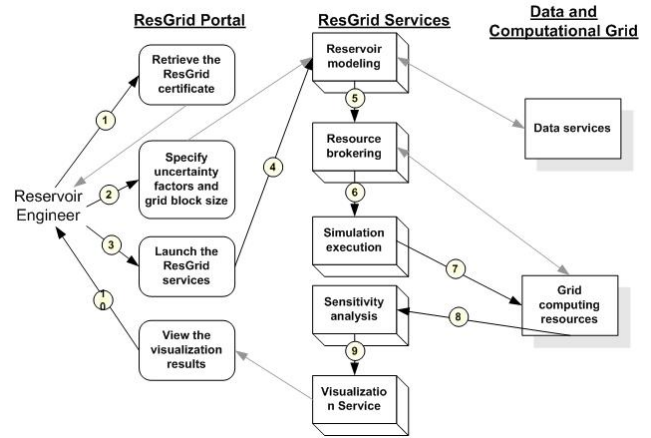


Figure 2. ResGrid usage scenario for a reservoir engineer

3.2. Architecture

The ResGrid is designed for reservoir uncertainty analysis to handle large-scale data management, massive reservoir simulations, multidisciplinary integration, and easy-to-use solution with high security. All these functionalities are accomplished by five components: ResGrid Portal, Reservoir Modeling, Massive Reservoir Simulation, Sensitivity Analysis, and Visualization. The ResGrid Portal is the start point. The Reservoir Modeling component prepares data and constructs reservoir models. With reservoir models, the Massive Reservoir Simulation component invokes the executions on the Grid. The Sensitivity Analysis and the Visualization components take care of post processing. The ResGrid users view simulation results via the ResGrid Portal.

A. ResGrid Portal

The Portal provides the entry point to the ResGrid, unifying Grid portal technologies and providing a web-based user interface. First, the portal deals with security — a GSI certificate is retrieved to provide for secure data transfer across the Grid environment and provide the authentication to access Grid resources. Second, the ResGrid Portal provides interfaces to specify the uncertainty factors (e.g., *Nuggets*, x , y , z ranges) and the problem solving scales (e.g., the grid block size of a reservoir). Third, a user can view and visualize results via this Grid portal.

B. Reservoir Modeling

This component archives distributed modeling-related data, creates the uncertainty parameter space, and constructs the reservoir models. Figure 3 shows the structure of the Reservoir Modeling component.

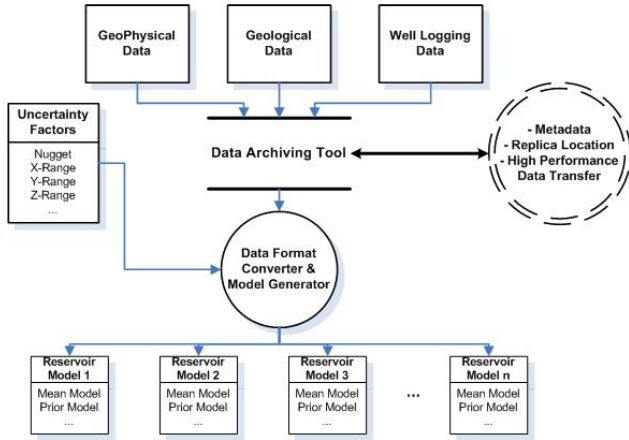


Figure 3. Structure of the Reservoir Modeling component

A data archiving tool is designed for data acquisition. There are three modules in this tool: metadata service, replica location service, and high performance data transfer service. The mechanism employed by this tool is as follows: given the information describing the required data, metadata service retrieves logical filenames, the replica location service locates the physical files which map to the logical filenames, and then these physical files are relocated via high performance data transfer. These modules are lightweight, which means they provide the interfaces to query the external Grid services offered by the Grid.

Modeling-related data include G&G data, exploration well data, production well data, etc. These datasets are geographically distributed with the size of terabytes, even petabytes. With the help of the data archiving tool, a base model is generated by extracting the modeling-related data. Uncertainty factors and factor levels are provided and the uncertainty parameter space is made. Based on this base model and the parameter space, massive reservoir models are constructed, each of which is associated with one combination of uncertainty factors and different factor levels. The number of models depends on the parameter space. Typically, it is up to multiple thousands. These models are the inputs of massive reservoir simulation runs.

C. Massive Reservoir Simulation

The Massive Reservoir Simulation component is in charge of massive simulations management, which includes workflow determination for a single reservoir simu-

lation, resource allocation, and massive simulation invocation.

Figure 4 illustrates the structure of this component. Task farming is engaged as the framework that takes reservoir models as inputs, checks a resource broker for resource allocation, and invokes massive simulation runs. The post process includes result analysis and visualization. Large-scale computation capability is required in this component.

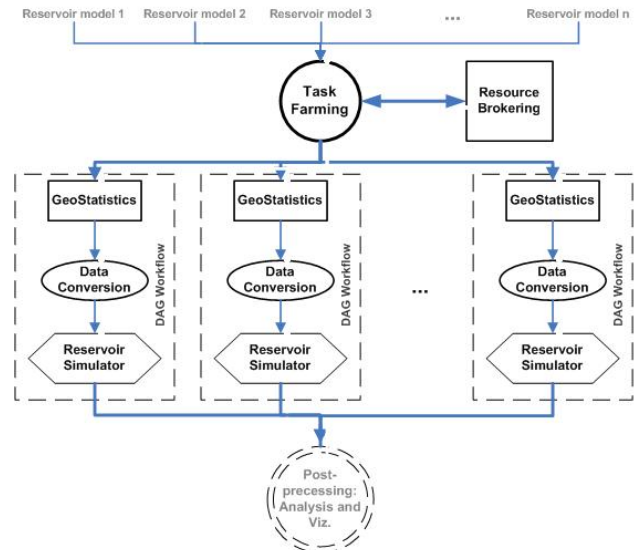


Figure 4. Structure of the Massive Reservoir Simulation component

The workflow of a single reservoir simulation integrates geostatistics algorithms with one execution of a reservoir simulator. Data conversion mechanism is developed between geostatistics algorithms and a reservoir simulator. The definition of such a workflow is open to allow a user to specify his/her own computational model without change on any other component.

A resource broker has been employed to manage Grid resources to share loads across the Grid. It captures resource information and uses load balancing strategies to invoke simulation runs. Two major factors of a resource are considered: computational capability and architecture. There is a matrix adopted to describe the features of a resource and measure its computational capability. The architecture factor of a resource is used to decide which type of binaries of geostatistics algorithms and reservoir simulators should be provisioned.

Task farming is the framework to invoke massive simulation runs. As mentioned above, the basic idea of uncertainty analysis is to run massive reservoir simulations with different models, which requires a number of nearly identical runs to produce the meaningful results. Task farming

over the Grid is a way to utilize multiple resources to meet such a requirement. By this way, massive simulation runs are invoked on various Grid resources with different reservoir models.

D. Sensitivity Analysis

The Sensitivity Analysis component is used to analyze reservoir performance uncertainty produced by various uncertainty factors. Now that the Massive Reservoir Simulation component has provided simulation results under different combinations of uncertainty factors with multiple levels, sensitivity analysis under the response surface models helps a reservoir engineer examine the sensitivity of oil production prediction to well location, absolute horizontal permeability, pore compressibility, aquifer size, skin, vertical permeability, etc.

E. Visualization

Visualization is used to graphically demonstrate uncertainty analysis results and assist decision making for further analysis. A reservoir engineer interacts with the Visualization component via the ResGrid Portal to obtain the visualization images dynamically according to the user's configuration.

3.3. Implementation Status

The three key components of the ResGrid have been finished: Reservoir Modeling, Massive Reservoir Simulation, and Sensitivity Analysis. These components have been used by reservoir researchers and engineers at Louisiana State University (LSU). Meanwhile, some results have been adopted by other Grid application projects (e.g., SCOOP [20]). The ResGrid Portal and the Visualization components are under development.

GridSphere, the de facto Grid portal standard, is employed to implement the ResGrid Portal. It provides various Grid computing related portlets [13], including credential retrieval portlet, job submission portlet, and file browser portlet. By customization, these portlets help unify all the services provided by the other components.

The Reservoir Modeling component has been finished. Data archiving is its essential part. A GAT-based data archiving tool has been developed. The GAT provides a generic and easy-to-use application programming interface (API) for the Grid, which is needed for application developers to create a new generation of "Grid-aware" application. It has a three-tier design consisting of the GAT-API, the GAT engine, and the different GAT adaptors. The GAT-API defines a simple, platform-independent API to generic Grid resources and services. The GAT engine is employed to transparently dispatch the API calls to the GAT adaptors which then provide the implementation of the required functionality. GAT adaptors provide all the functionality

in the GAT, which makes it straightforward to incorporate new services and technologies without any change to application code. Three GAT adaptors have been developed to implement this data archiving tool: MCAT [12] adaptor for meta data service, Globus RLS adaptor for the mappings from logical file to physical files, and GridFTP adaptor for high performance and secure data transfer. Benefiting from the design of the GAT, this tool demonstrates high flexibility and portability in diverse Grid environments according to Grid core service availability and performance requirements. For instance, one can easily replace the MCAT adaptor by a Globus RLS based advert adaptor for meta-data management, GridFTP adaptor by the CURL [6] adaptor for data transfer, or RLS logical file adaptor by the SRB [15] adaptor.

The implementation of the Massive Reservoir Simulation component consists of three aspects. The first one is to determine the workflow of a single simulation. The second one is a resource broker, which keeps track of which machines are available to run jobs, how the machines should be utilized, and when a machine is no longer available. A load balancing strategy has been developed to share large-scale simulation runs on the Grid. It assigns a value to each resource as its weight. Using the weight, the resource broker decides how many reservoir simulation runs should be dispatched on the resource. The third one is to invoke massive runs of reservoir simulation. Condor-G and Globus GRAM are engaged. Condor-G lets one submit jobs into a queue and have a log detailing the life cycle of the jobs along with everything else expected from a job queuing system. GridFTP is used instead of the IO management provided by Condor-G (i.e., Globus GASS) due to data transfer performance. Globus GRAM provides underlying software needed to utilize Grid resources, such as authentication and remote program execution.

The development of the Sensitivity Analysis component is straightforward. First of all, it extracts the influence index from simulation results for various combinations of all uncertainty factors with different factor levels. Then, the sensitivity of uncertainty factors with the different levels are calculated and quantified against the response surface model mentioned in Section 2.1.

The Visualization component of the ResGrid is still under development. HDF5 [11] is adopted for data format and OpenDX [14] to display the results.

The security of all the ResGrid services is based on GSI, the de facto Grid security standard. The GSI provides robust security mechanisms. It includes an OpenSSL implementation. It also provides a single sign-on mechanism, so that once a user is authenticated, a proxy certificate is created. With this certificate, a reservoir engineer can perform data operations within the Grid securely.

3.4. Case Studies

The ResGrid is being developed in close coordination with researchers from Petroleum Engineering Department at LSU and the first application is to compare three different stochastic simulation algorithms according to their flow responses. It is a common way to create permeability fields by stochastic simulation algorithms, which honors the available information at the wells and reproduces the pattern of spatial variability between wells. The stochastic simulation can be categorized into direct (LU matrix Gaussian Simulation) and sequential approaches (Sequential Gaussian Simulation). LU matrix Gaussian Simulation (LUSIM) is rigorous but slow. Sequential Gaussian Simulation (SGSIM) is quicker but inaccurate. We create a hybrid simulation (HYBRID) to take the advantages of the direct and sequential approaches. The flow response is used to find differences among these simulators. In our experimental design and response surface model, there are four geological factors (e.g. nugget effect, x-range, y-range and z-range) involved, each of which has four levels to cover the all feasible factor values. Four-level full factorial design requires 256 simulation runs for each algorithm. Five realizations are created for each geostatistical parameter combination. The total simulation run is 3840(= 256 × 5 × 3). All the simulation runs at CCT Grid testbeds, including two Linux clusters, helix (256 nodes) and supermike (1024 nodes). Sweep efficiency, break through time and upscaled permeability are extracted as responses from the summary files. Multiple linear regressions fits response surface models for the four factors and three responses in three directions. Main effects, interacting effects and quadratic effects are obtained (14 regression coefficients). Several points are concluded after the study:

- LUSIM permeability fields give the best prediction for all the responses, but the difference between LUSIM and HYBRID is rather small. LUSIM is much more time-consuming than HYBRID.
- All factors are significant for at least one response.
- Most interaction terms are insignificant.
- All factors have significant quadratic terms for at least one response.
- All algorithms are most sensitive to along the correlation, rather than diagonal and cross the correlation.

Components of ResGrid are also being used for coastal modeling in the SCOOP project. Our study shows that coastal modeling scenario has many similarities to reservoir uncertainty analysis, which needs to archive large-scale datasets and execute thousands of simulations.

As an experimental design framework and PSE environment, ResGrid is expected to become a generic architecture for other related Grid applications in the future.

4. Related Work

GAT/SAGA, are two Grid application abstraction efforts targeted at providing application-level toolkits for building higher-level functionality on top of core Grid services, hiding the complexity of the Grid. The idea of Grid portal is to implement a generic execution environment in which a user can “drop” his/her application for Grid execution while maintaining the convenience and the illusion of a desktop execution. The ResGrid integrates both application level efforts to generate a problem solving environment for reservoir uncertainty studies.

An autonomic reservoir framework [4] has been studied by W. Bangerth, H. Klie, etc. A prototype application was designed and developed to use P2P interactions between applications and services on the Grid to enable the autonomic optimization of an oil reservoir. It optimized the placement and operation of oil wells to maximize overall revenue. The application consisted of instances of distributed multi-model, multi-block reservoir simulation components provided by IPARS, simulated annealing based optimization services provided by VFSA, economic modeling services, and experts connected via pervasive collaborative portals. This framework emphasizes the optimization and the integration of high level services of reservoir management, such as well placement and economical influence. Our efforts focus on reservoir performance prediction and uncertainty analysis based on the G&G characteristics of a reservoir.

COUGAR [5] is an industrial contribution on reservoir studies. It is a reservoir uncertainty analysis tool with the ability to make use of Grid resources to run a number of reservoir simulations and achieve the reduction in the individual result turnaround time. However, it does not address large-scale G&G data integration, and its framework is tied to commercial packages, such as LSF® for execution invocation, ECLIPSE® for reservoir simulator, and security issues are not considered. The ResGrid provides an open generic framework to solve reservoir uncertainty analysis with open source software packages and a tight security consideration.

5. Conclusions and Future Work

Our work focuses on a Grid-aware toolkit ResGrid, which provides an integrated, easy-to-use and secure PSE for reservoir uncertainty analysis. In this paper, the five-component ResGrid architecture has been presented.

A data archiving tool has been implemented using the GAT. With this tool, a reservoir engineer can archive different G&G datasets and well logging data. Based on this data tool and the uncertainty factor parameter space generation mechanism, the Reservoir Modeling component was implemented.

A Grid resource broker and a task farming framework have been developed. The resource broker captures Grid resource information and uses load balancing strategies to invoke reservoir simulations across Grid resources. Task framing is used to contribute reservoir simulation runs combined with geostatistics algorithms across a Grid. Both have been integrated into the Massive Reservoir Simulation component.

The Sensitivity Analysis component was implemented to analyze the sensitivity produced by various uncertainty factors with multiple factor levels.

The reservoir researchers from the Petroleum Engineering Department at LSU have adopted the ResGrid as a simple and practical integrated toolkit for their uncertainty studies. Coastal modeling researchers from the SCOOP project are also using parts of this toolkit in their development of an integrated infrastructure for ocean observing and prediction.

However, there remains much work ahead in the further research and development of the ResGrid. The ResGrid Portal and Visualization components are under development, which will provide easy-to-use user interfaces. Efforts are underway to provide monitoring and steering capabilities at runtime during the execution of a given simulation run, to provide the possibility to check job status and terminate the job if an error occurs. Another challenging issue in our future work is how to couple the latest optimization algorithms with history matching into the ResGrid.

6. Acknowledgements

We acknowledge the use of Grid testbeds provided by CCT. This research is supported by the UCOMS grant (DOE DE-FG02-04ER46136) and the CCT.

References

- [1] G. Allen, K. Davis, et al. "Enabling applications on the Grid: A GridLab overview". *International Journal of High Performance Computing Applications, Special issue on Grid computing*, 2003.
- [2] G. Allen, K. Davis, et al. "The Grid Application Toolkit: Towards Generic and Easy Application Programming Interfaces for the Grid". *Proceedings of the IEEE, Vol. 93, No. 3*, pp. 534-550, March 2005.
- [3] K. Aziz, A. Settari. "Petroleum Reservoir Simulation". *Applied Science Publishers Ltd.*, London, 1979.
- [4] W. Bangerth, H. Klie, et al. "An Autonomic Reservoir Framework for the Stochastic Optimization of Well Placement". *The Journal of Networks, Software Tools, and Applications, Special Issue on Challenges of Large Applications in Distributed Environments, Vol. 8, No. 4*, pp. 255-269, 2005.
- [5] COUGAR. http://www.slb.com/media/services/software/research/ressim_ondemand.pdf.
- [6] CURL and LIBCURL. <http://curl.haxx.se/>.
- [7] I. Foster, C. Kesselman. "Globus: A toolkit-based Grid architecture". *The Grid: Blueprint for a New Computing Infrastructure*, pp. 259-278, 1999.
- [8] I. Foster, C. Kesselman, et al. "A security architecture for computational grids". *Proceedings of the 5th ACM Conference on Computer and Communications Security, San Francisco, CA*, pp. 83-92, 1998.
- [9] Globus Toolkit. <http://www.globus.org>.
- [10] S.Kalla, C.D.White, "Efficient Design of Reservoir Simulation Studies for Development and Optimization". *SPE Annual Technical Conference and Exhibition, Dallas, Texas, U.S.A.*, October 9-12, 2005.
- [11] HDF5. <http://hdf.ncsa.uiuc.edu/HDF5/>.
- [12] MCAT. <http://www.npaci.edu/DICE/SRB/mcat.html>.
- [13] J. Novotny, M. Russell, O. Wehrens. "GridSphere, A Portal Framework for Building Collaborations". *IEEE Concurrency and Practice*, April, 2004.
- [14] OpenDX. <http://www.opendx.org>.
- [15] A. Rajasekar, M. Wan, et al. "Storage Resource Broker - Managing Distributed Data in a Grid". *Computer Society of India Journal, Special Issue on SAN, Vol. 33, No. 4*, pp. 42-54, Oct 2003.
- [16] J.M. Schopf, S.J. Newhouse. "State of Grid Users: 25 Conversations with UK eScience Groups". *Argonne National Laboratory Technique Report ANL/MCS-TM-278*, October 25, 2004.
- [17] R.W. Schulze-Reigert, J.K. Axmann, et al. "Evolutionary Algorithms Applied to History Matching of Complex Reservoirs". *2001 SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A.*, February 11-14, 2001.
- [18] K. Seymour, H. Nakada, et al. "Overview of GridRPC: A Remote Procedure Call API for Grid Computing". *Proceedings of the Third International Workshop on Grid Computing, Parashar, M. eds. pp. 274-278*, 2002.
- [19] Simple API for Grid Application. <http://forge.gridforum.org/projects/saga-rg/>.
- [20] SURF Coastal Ocean Observing and Prediction Program. <http://www1.sura.org/3000/3300Coastal.html>.
- [21] D. Thain, T. Tannenbaum, M. Livny. "Condor and the Grid", *Grid Computing: Making the Global Infrastructure a Reality*. John Wiley, 2003. ISBN: 0-470-85319-0.
- [22] C.D. White, S.A. Royer. "Experimental Design as a Framework for Reservoir Studies". *2003 SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A.*, February 3-5, 2003.
- [23] J. Zhang, M. Delshad, et al. "An Efficient Reservoir Simulation Approach to Design and Optimize Improved Oil Recovery Processes with Distributed Computing". *SPE Latin American and Caribbean Petroleum Engineering Conference, Rio de Janeiro, Brazil*, June 20-23, 2005.