The CO_2-Community Grid - A Virtual Research Environment for the Simulation of CO_2-Sequestration

K. Johannsen [1] , P. Binning [2] , C. Anderlik [3] , A. Kopp [4] and O. Tourunen [5]

1) *Department of Computational and Applied Mathematics, University of Bergen, Norway*

2) *Technical University of Denmark, Department of Environmental Engineering, 2800 Kgs. Lyngby, Denmark*

3) *Bergen Center for Computational Science, University of Bergen, Thormøhlensgate 55, N-5008 Bergen, Norway*

4) *Institut für Wasserbau, University of Stuttgart, Pfaffenwaldring 61, 70569 Stuttgart, Germany*

5) *Nordic Data Grid Facility, Kastruplundgade 22, DK-2770 Kastrup, Denmark*

Corresponding Author : K. Johannsen, klaus.johannsen@bccs.uib.no

ABSTRACT

In this paper we present a Grid infrastructure for massively parallel simulations related to CO_2-sequestration in porous media. It currently allows for the usage of a number of supercomputers located in Denmark and Norway and is based on the NorduGrid *ARC* middleware. The infrastructure provides an unified access to the supercomputers and supports the parallel execution of applications on the base of *MUFTE-UG*, a simulation platform for Multi-Phase-Multi-Component flow simulations. The Grid-System allows for concurrent execution of multiple instances of *MUFTE-UG*.

1 INTRODUCTION

The idea of grid computing dates back to the work of Faber et. al. in the 1970s [1,2] and has been followed up in 1992 by C. Catlett and L. Smarr with the concept of metacomputing [7]. The final breakthrough has been obtained by Foster et. al. [5,6]. Since then various grid technologies have been developed. Major interest is to create a pool of distributed resources, which can be accessed tranparently by various user groups. Grid computing has been dominated a long time by the high energy physics communities, but recently other communities make use of grid technologies to share compute and/or storage resources.

In this paper we describe a Grid infrastructure, which supports massively parallel computing. The research focus of the activity is on modeling and simulation of CO_2-sequestration. The infrastructure is based on the middleware *Advanced Resource Connector* [3] developed by NorduGrid. It provides a solution for the sharing of resources and eases the use of compute facilities. The numerical software used for the massively parallel simulations is *MUFTE-UG*, a general purpose simulator for multi-phase, multi-component flow in porous media [4]. A collaborative workspace realized by a by a portal (application server), which allows for process control and data exchange.

2 INFRASTRUCTURE

The Grid Infrastructure is based on *ARC*, a middleware developed by NorduGrid. It is an open source software solution for production quality computational and data grids. The middleware builds upon standard Open Source solutions like the *OpenLDAP*, *OpenSSL* and *Globus Toolkit* libraries. On the base of *ARC* an *Application Server (AS)* is realized to provide Grid access for users via a `ssh` portal.

2.1 The Command line User Interface

The AS allows the user to interact with the Grid infrastructure by means of a slim command line interface (UI) written in Perl. The UI provides one single command `grid`, which provides access to different submodules. The submodules in turn implement the job control functionalities. The four submodules `submit`, `status`, `cancel` and `retrieve` are realized with the following functionalities.

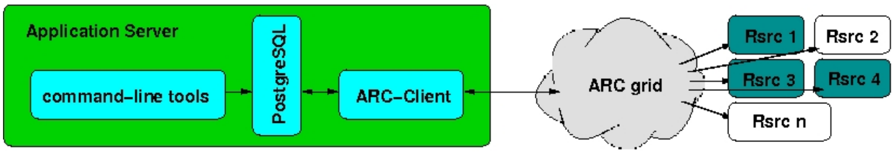

Figure 1. Structure of the application server. The command line tools interact with the database, which in turn communicates with the *ARC* client. The latter communicates with the Grid infrastructure.

- `submit` - This submodule submits a job to the Grid.
- `status` - This submodule queries the status of all jobs or a particular job.
- `cancel` - This submodule cancels one or more jobs.
- `retrieve` - This submodule retrieves the results of a job.

The UI interacts with a management database containing the information about jobs, users and the Grid infrastructure. This structure of the AS is displayed in Fig. 1. The common workflow for a job is:

(1) The user logs on the application server using `ssh`.
(2) The user prepares a job using `grid submit`. This create the initialization, the tarball containing the software and updates the management database.
(3) The user is able to monitor the progress of the job using `grid status`.
(4) Once the task completes the final results or any partial results of the task will be available in a location defined by the user.

2.2 The Grid Back-end

The Grid Back-end takes care of executing user tasks on the available Grid resources. The main components are the Grid Job Manager (GJM) running on the AS and the *MUFTE-UG* application runtime environment (RE) installed on the computational resources. The GJM is the main logical component of the Grid Back-end. It submits jobs to the Grid, monitors job states, retrieves results and provides information about job progress. It hides failures from the user and supports complete re-trying of the failed jobs. The REs implement the software environment needed on the computational resources. They are application specific and take care of compiling, linking and execution of the jobs. Typical workflow of Grid Back-end for a job is:

Resource	Location	#Cores	Type	Alloc. '08 [CPUh]
Fimm	Bergen	172	AMD/Opteron	125.000
Fyrkat	Ålborg	672	Blades	460.000
Hexagon	Bergen	5552	AMD/Budapest	330.000
Stallo	Tromsø	5632	Xeon 2	470.000
Titan	Olso	2152	Xeon/Opteron	841.000
Sum				**2.226.000**

Table 1 Resources available to the Grid Infrastructure. The amount of the allocations are preliminary and subject to approval.

(1) A new job is found in the database.
(2) The GJM submits the job to the Grid. *ARC* finds a suitable resource.
(3) The RE compiles the software for the job on the computing resource.
(4) The *ARC* server submits the job to the local resource management system and executes it.
(5) The GJM checks the status of the job and updates the database accordingly.
(6) When the job finishes, the GJM downloads the results into a user specified directory or into global spool.

All authentication and authorization is based on standard X509 certificate infrastructure omnipresent in current grid middleware. All users are listed in the CO2-CG Virtual Organization Management System (VOMS) run by NDGF and the computational resources authorize users based on the information in the VOMS.

3 RESULTS

The deployment of the Grid infrastructure started in fall 2007. The application server `vre-as.bccs.uib.no` realizes Grid connections in Norway with the computing centers in Bergen (*Fimm*,*Hexagon*), Oslo (*Titan*) and Tromsø (*Stallo*) and in Denmark in Aalborg (*Fyrkat*).

It is used by the group *Parallab/BCCS* at the University of in Bergen, Norway and the group for *Hydromechanics and Hydrosystem modelling* at the University of Stuttgart, Germany. The resources currently available are given in Table 1. The system is in production since February 12th, 2008. Since then 322 parallel jobs have been executed in the infrastructure on 4 to 64 processors with execution times between 30 min and 4 days corresponding to a total of 102.000 CPU hours.

The numerical simulations are related to the sequestration of CO_2 in deep aquifers. The risk assessment for sequestration scenarios estimates e.g. leakage rates from aquifers used for sequestration. A result obtained for a typical radial-symmetric model problem is shown in Fig. 2. In this example CO_2 is injected on the left and leaks out through a well on the right.

ACKNOWLEDGMENTS

The support of this work by the Nordic Data Grid Facility (NDGF) and the Norwegian national infrastructure for computational science (NOTUR) is gratefully acknowledged.

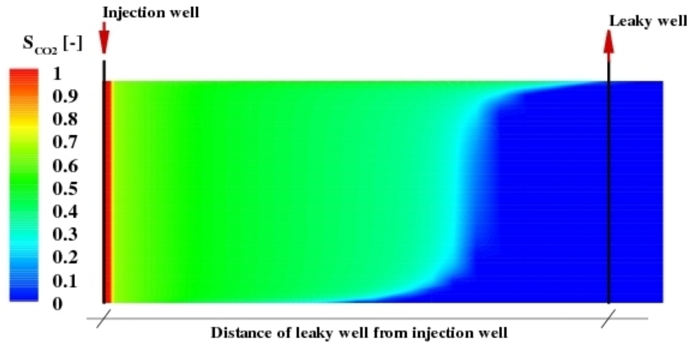

Figure 2. Radial-symmetric CO_2-injection of CO_2 in a aquifer (left) and leakage through a well (right).

REFERENCES

1. Farber D.J. and Larson K.C. The structure of a distributed computing system - software. *In: Proc. Symp. Computer-Communications Networks and Teletraffic (New York)*, pages 539–545, 1972.
2. Farber D.J. and Larson K.C. The system architecture of the distributed computer system - the communications system. *In: Proc. Symp. Computer-Communications Networks and Teletraffic (New York)*, pages 21–27, 1972.
3. Ellert M. et. al. Advanced resource connector middleware for lightweight computational grids. *Future Generation Computer Systems*, 23(2):219 – 240, 2007.
4. R. et. al. Helmig. Architecture of the modular program system mufte ug for simulating multiphase flow and transport processes in heterogeneous porous media. *Mathematische Geologie*, 2:123–131, 1998.
5. Foster I. hat is the grid? a three point checklist. *Argonne National Laboratory and University of Chicago*, 2002.
6. Foster i. and Kesselman C. *The Grid: Blueprint for a Future Computing Infrastructure*. Morgan Kaufmann Publishers, San Francisco, CA, 1998.
7. Smarr L. and Catlett C.E. Metacomputing. *Communications of the ACM*, 35(6):44–52, 1992.