Software Environment of a Grid-based Virtual Organization for Flood Prediction

L. Hluchy, O. Habala, J. Astalos, V. D. Tran, B. Simo, M. Dobrucky, E. Gatial and M. Maliska[a]

[a] *Institute of Informatics of the Slovak Academy of Sciences,Dúbravká cesta 9, Bratislava, Slovakia*

hluchy.ui@savba.sk

Abstract: In this paper we present architecture and current state of implementation of a collaborative computation environment based on Grid infrastructure, used as a mean of support for large scientific virtual organizations. The environment consists primarily of a collaboration-supporting user interface, workflow system capable of submission of jobs to the Grid and a Grid-based data management suite.

The whole system is controlled via a web-based user portal, which enables to design and control simulations, to preview results and to communicate with other users. The workflow management module is responsible for executing a set of consecutive computation jobs and linking their input/output data. This computation engine is backed by a data management module, which performs data storage/retrieval tasks and metadata queries.

A prototype of such an environment is deployed and tested for a flood forecasting system. The system consists of workflow system for executing simulation cascade of meteorological, hydrological and hydraulic models, data management system and a set of web portals.

Keywords: Flood prediction; Grid computing; Problem Solving Environment; Modelling

1. INTRODUCTION

The problems of flood prediction are very actual every summer in Europe, with vast catastrophes in several countries each year. Better than acting after the flood occurs is to predict it, prevent it, or at least minimize the damage it may cause.

We are developing a software suite for modelling and prediction of flood, using state-of-the-art technologies. The whole system is a Problem Solving Environment (PSE) [Gallopoulos et al., 1994] consisting of a Grid-based simulation core, monitored and controlled by a comfortable user interface, accessible through a stand-alone Java application or a web portal.

This paper describes the architecture of this system, its usage scenario and possible interactions with its environment. Then the most important components are described in more detail, together with their history and directions of future development.

2. ARCHITECTURE OF THE SOFTWARE ENVIRONMENT FOR FLOOD PREDICTION

The software of our PSE consists of several hierarchically organized modules (Figure 1). We may say that this architecture is principally similar for every PSE supporting a scientific virtual organization (VO).

The user accesses the system through a comfortable interface. He may choose between a Java application and a web-based portal, depending on his preferences for security and mobility. Underlying this interface are three tools for the management of the Grid systems and applications of the core computational facility – the workflow management system, data management and collaborative tools.

The workflow system is responsible for the management of the set of applications (simulation models) used for flood prediction (these will be described later). Because of the large amounts of data consumed and produced by the simulation

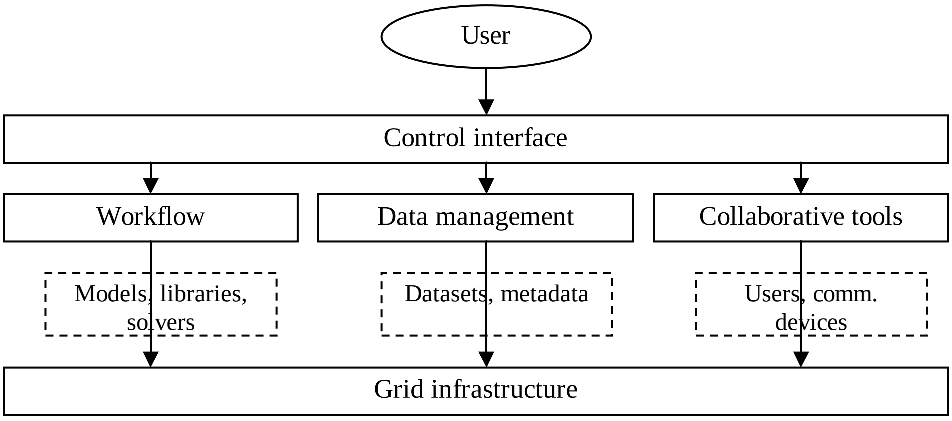


Figure 1. Architecture of a Grid infrastructure for scientific VO

processes involved, an automated data management suite is also necessary. To actually enable two or more scientists to work together towards achieving a common goal, a collaboration and communication suite is integrated in the infrastructure.
The rest of this paper describes the parts mentioned above, as well as the set of simulations used for flood prediction.

3. WORKFLOW MANAGEMENT

3.1 Introduction

As the grid infrastructure matures it is being used by scientists for more and more complex computations. Each such computation can include executions of several applications and transfers of required data. The complexity of the process is becoming too high to be handled manually. Therefore the employment of workflow concept seems quite natural. The business community is widely using the workflow concept and in this context it has been defined as follows:
"Workflow is the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules."
Similar to the definition of a workflow in a business process management, a grid workflow is an automation of a grid process, during which documents, information or data are passed from one grid service to another for action, according to set of procedural rules.

3.2 Existing systems

Most of grid workflow systems being developed focus on the web services, al-though there are older systems that do not use the web services paradigm.
One example of system not using web services can be the Condor DAGMan – a meta-scheduler for the Condor workload management system. It uses a directed acyclic graph (DAG) to represent a set of programs where the input, output, or execution of one or more programs is dependent on one or more other programs. Pegasus is a system for transforming abstract workflow descriptions into concrete workflows, which are then executed using the Condor's DAGMan.
Web services oriented workflow systems are mostly in early stage of development as can be seen in the Scientific Workflow Survey web page [Survey]. There are two main specifications of workflow languages: Web Services Flow Language (WSFL) targeting web services and the Grid Services Flow Language (GSFL), which builds upon Open Grid Services Architecture (OGSA) [Foster et al. 2003]. OGSA is based on grid services, which are web services with additional grid-oriented features, and allows distributed resources to be shared over a network. Currently, we are not aware of any existing workflow system using either of these languages.

3.3 Workflow management for flood prediction

We need an interactive portal-based workflow system that enables the user to construct a workflow or to choose from predefined ones.

As for the interactivity, it means the possibility to view the results of each task (activity) instantly after it has finished without waiting for the whole workflow to finish and ability to clone existing (possibly running) workflow and submit it with modified parameters. The modification may cover one or more tasks.

Important feature is the ability to replace selected step or steps in the workflow with user selected or defined "output" in order to let the user perform various parameter studies. Such replacement must be possible both during workflow definition and during workflow execution.

A workflow system that we designed for our flood prediction system enables the user to define whole cascade execution in advance as a workflow and run it with the possibility to inspect every step.

The whole flood simulation uses three main steps – meteorology, hydrology and hydraulics - to produce the final result – the prediction of the parts of the target area that are going to be flooded. When the expert wants to use already computed results or does not need to compute the last step of the cascade, just parts of the cascade are required. The run of a single simulation model represents the simplest case.

So we have several possible workflow templates that may be executed. We have decided to constrain the workflow selection to several predefined workflows in the first version. Workflow is defined for each target area based on the computation dependencies for that particular area. The changing part of the workflow is mainly hydrology because the run-off in the target catchment is computed from several sub-catchments.

An expert who wants to perform a simulation chooses a target area and time for which to make the prediction, then the workflow template from the list of templates available for the area of interest and a model to be used in each step. The possibility to select more models for the same step or even to enter user defined values instead of running a particular simulation step makes it possible to have several parallel in-stances of a workflow giving several results for the same time and area.

4. COLLABORATIVE TOOLS

The need of cooperation between scientists and users from many organizations in Grid projects requires sophisticated tools for collaborations in portals. The scientists need to access and share data, analyze them, and discuss with other scientists via the collaborative tools. Therefore, collaborative tools are one of the key elements of virtual organizations.

Collaborative tools may be mailing lists, instant messaging, file-sharing tools, discussion groups, etc. However, one single tool cannot provide all features necessary for the collaborations. Therefore, there are several projects that aim to provide an integrated and extensible collaborative environment via portals. One of such projects is CHEF [CHEF].

CHEF (CompreHensive collaborativE Framework) is a collaborative environment based on Jespeed portal engine. The collaborative tools (teamlets) are written as portlets in Jetspeed [Jetspeed] that are extended to special features for multi-user group work nature of collaborative tools:

- Resource access security: The users can only view and modify what they have permissions to.
- Automatic updating of displays: as a user makes changes that affects the display other users are viewing, their display is automatically updated.
- Multi-user safe: if several users are using the same tool at the same time, they work together to avoid conflicts.
- Presence: every user can see who else is using the same tool in the same area at the same time.
- Notification: every user can request to be notified of changes made through the tool by other users.

For accessing to the collaborative tools in CHEF, users need a standard web browser and access to the portal.

5. DATA MANAGEMENT IN SCIENTIFIC VIRTUAL ORGANIZATIONS

5.1 Introduction

The increasing needs for volume and accessibility of data in scientific computations in the last decade leads also to increased demands on better data management tools. The main responsibilities of such software are:

- To track available datasets in the virtual organization.
- To store and maintain these datasets in a coherent fashion.
- To publish their properties and enable their discovery.
- To enable their download and usage.

The data stored in a virtual organization's data storage facilities has two main parts - the actual datasets and their metadata (meaning their description by another layer of data). Especially the metadata management is evolving rapidly in recent years, as the already quite mature global network infrastructure enables to create larger virtual organizations and data collections with more complicated search and discovery of relevant datasets. Thus, also the data management efforts are divided into two main streams – replica management and metadata storage/lookup.

5.2 Replica management

The actual storage and maintenance of a coherent dataset collection is performed by replica management software. It keeps track of the datasets, potentially stored at multiple places redundantly (replicated). The creation of replicas of a single dataset may be well used for better security and protection against an unwanted loss of the dataset because of a sudden storage device failure, as well as for better access to the file by making it more local to the place that requires it. Although the term replica management may be pertinent to several areas distributed computing research, we deal mainly with the Grid and Grid computing paradigm. For the Grid, a replica management suite has been developed in the European DataGrid Project [Hoschek et al.]. The software developed in work package 2 of the DataGrid [Kunszt et al. 2003] covers the registration, lookup, transfer and replication tasks of a mature replica management suite, with sufficiently distributed control. Its last implementation is based on the modern paradigm of web services and OGSA [Foster et al. 2003] architecture. Anyway, it is lacking a modern and scalable metadata repository.

5.3 Data sources for flood prediction

The general schema of possible data sources for the operation of the Virtual Organization for Flood Forecasting - FloodVO was described in previous work [Hluchý et al. 2003] and also included in the virtual organization architecture. From these sources, only some were realized in the prototype stage of FloodVO.
The most important data in FloodVO storage are the boundary condition for the operation of our meteorological prediction model ALADIN. The second type of data implemented in the prototype stage of FloodVO are radar images of current weather conditions in the pilot operation area, and the third type of currently available data are the ground-based water level, precipitation and temperature measurements provided by SHMI's (Slovak Hydrometeorological Institute) network of measurement stations. The last type of data currently under development are satellite images of the pilot operation site.

5.4 Metadata management

The problem of managing and searching the descriptive information of the dataset collection of a virtual organization (especially for large international scientific Grid-based virtual organizations) is in its nature very similar to the problems of recent peer-to-peer computing efforts. Potentially, the space of storage nodes in such an organization is very large and the especially the distributed lookup is a non-trivial problem. Various solutions have been proposed and evaluated [Joseph and Hoshiai 2003], but the more efficient of them pose severe restrictions on the stored metadata. But several peer-to-peer infrastructure problems, connected with the high instability of the whole network may be disregarded in Grid computing, and in such a controlled environment a decentralized and efficient metadata registry may be deployed. Also, considering the better and more reliable network infrastructure available in grids, a certain level of centralization may be tolerable, without the fear of creating a single point of failure of a bottleneck in the metadata lookup middleware.
The situation in metadata management becomes also more complicated with the re-cent advances in semantic web and Grid. The introduction of ontologies into resource description (like the RDF standard) creates a new area of problems, connected with replication, lookup and especially security of ontology graphs. These problems have yet to be solved, but certainly graph representation of metadata will in the near future begin to replace current, relational representation.

5.5 Prototype implementation

Data management in the prototype of FloodVO was implemented mainly using these software tools, provided by the European DataGrid (EDG) IST project (EDG Replica Manager and underlying services). The metadata database was implemented using the MySQL RDBMS and the EDG Spitfire Grid interface to this RDBMS. A service and client application have been implemented. The client enables to add, modify, locate and delete metadata for given file in the FloodVO SE (identified by its GUID). The metadata service is also accessible through the portal interface, where a user can locate datasets and their details based on their properties.

6. USER INTERFACES FOR COLLABORATION

There are three different user interfaces in various stages of development that provide access to the grid for the flood application.
We have developed GridPort [Thomas et al. 2001] based application portal, we are developing flood application specific portlets for the Jetspeed portal framework based application portal and we are being integrated with Java based client called Migrating Desktop.

6.1 Application portal with the Jetspeed framework

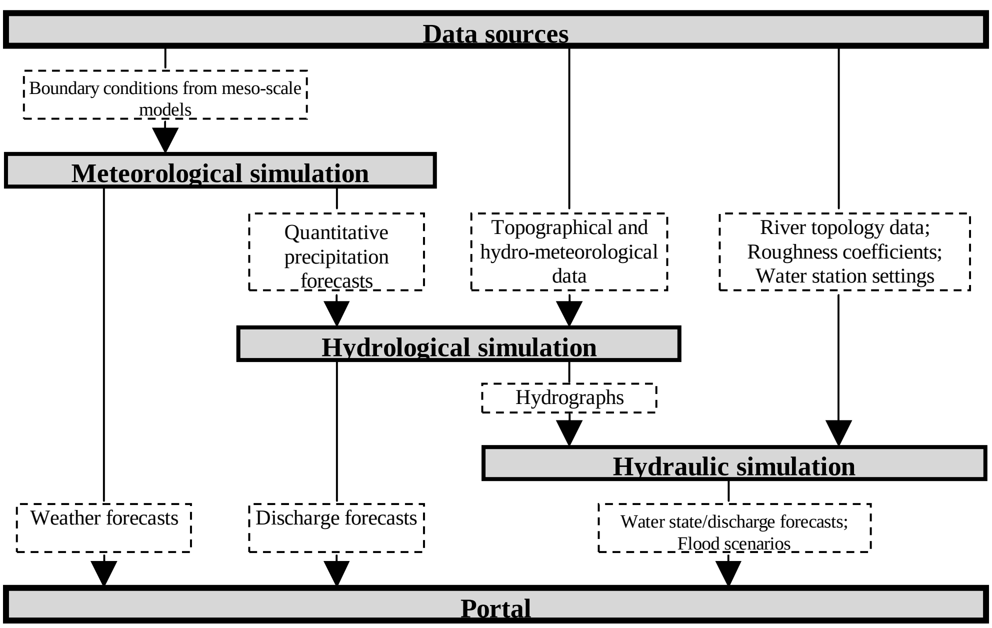
The Jetspeed portal framework has been chosen in the CROSSGRID project as a modern powerful platform for creating grid application portal for the applications in the project. This framework is also being used by other grid projects such as Alliance portal and the new version of the GridPort toolkit – GridPort 3.0.
Jetspeed is implemented as a server-side Java based engine (application server). Client services are plugged in using software components called portlets. Each portlet has a dedicated space on the screen, which it uses for communication with users. Portlets are independent from each other and user can arrange their position, size and visibility.
Jetspeed provides framework for building information portals (pluggable portlets mechanism, user interface management, security model based on permissions, groups and roles, persistence of information etc.) but does not provide any support for grid services and applications. Common Grid portlets that can be used in Jetspeed are being developed in CROSSGRID and other projects.
Portlet for submission of specific simulation models of flood application has been developed and now we are focusing on automation of a computation of the flood simulation cascade by employing workflows. We are also investigating the possibility of using groupware portlets from the CHEF project.

6.2 Migrating desktop

Migrating Desktop is a Java client being developed in the CROSSGRID project. The idea was to create user interface with greater interactivity than could be possible to achieve by using web technology.
Current version provides access to basic Grid services such as authentication, job management, and file management. Support for specific application features is addressed by application and tool plugin interfaces that enable to plug in code handling application specific parameter definition and visualization. We have implemented both plugins for the flood application.

7. SIMULATION CASCADE

Flood forecasting requires quantitative precipitation forecasts based on the meteorological simulations of different resolution from meso-scale to storm-scale. Especially for flash floods, high-resolution (1 km) regional atmospheric models have to be used along with remote sensing data (satellite, radar). From the quantitative precipitation forecast hydrological models are used to determine the discharge from the affected area. Based on this information hydraulic models simulate flow through various river structures to predict the impact of the flood. The results of simulations can be interactively reviewed by experts using the PSE, some of them accepted and forwarded to the next step, some rejected or re-run with modified inputs.
So the cascade of simulations, which we have just described, starts by running a meteorological



model. The plan is to use several of them, but the primary one will be the ALADIN model, currently used in SHMI. This model has been modified so that it can be used in a cluster environment, with MPI as a messaging interface. The model will be run twice a day, creating forecasts for the next 48 hours. The forecasts will consist of precipitation, wind speed and direction in several heights.

The output from the ALADIN model will be used to run hydrological simulations. Here we will again use more models, some simple and some more elaborated (and more demanding). These models will be run in a high-throughput environment, using a pool of workstations and the Condor management toolkit. The output from these simulations will be a hydrograph, representing estimated water level in the target area. This hydrograph will be used in the last stage of the simulation cascade, in the hydraulic simulation (Figure 2).

8. FUTURE WORK

The core of our workflow system has been implemented and is running there are some important features that are yet to be implemented. For example, the visualization portlet is not directly connected to visualization jobs, so the user has to browse to the output directory manually.

We also plan to create a new portlet for creation and modification of the workflow templates, and implement features, which would enable a user to see standard error and standard output interactively during the job run.

The metadata service will be more tightly integrated into the user interface and workflow system, so the user may choose data for simulations based on a group of constraints, rather than its physical location.

However, the most important goal is to integrate our portlets into a collaboration tool in order to enable the exchange of ideas and results among the experts working with the portal. We are currently investigating the possibility to integrate with the CHEF collaboration portal that is based on the Jetspeed framework and therefore also uses the portlet paradigm.

9. CONCLUSIONS

The software system we present here is a specialized simulation tool for flood prediction, which uses a cascade of meteorological, hydrological and hydraulic models to predict water flow in a river basin. The system is controlled by a user interface with integrated workflow and data management and tools for user collaboration. The current prototype implementation will be further refined and enriched by new models and features.

10. ACKNOWLEDGEMENTS

This work is supported by EU 5FP CROSSGRID IST-2001-32243 RTD project and the Slovak Scientific Grant Agency within Research Project No. 2/7186/20.

11. REFERENCES

Gallopoulos, S., Houstis, E., Rice, J., Computer as Thinker/Doer: Problem Solving Environments for Computational Science. *IEEE Comp. Science and Engineering Magazine*, 1994, Vol. 2, pp. 11-23.

Hoschek, W., et. al., Data Management in the European DataGrid Project. *The 2001 Globus Retreat*, San Francisco, August 9-10 2001.

Foster, I., Kesselman, C., Nick, J. M., Tuecke, S., The Physiology of the Grid; An Open Grid Services Architecture for Distributed Systems Integration. *The Globus Project*, 2003.

Hluchý L., Habala O., Simo B., Astalos J., Tran V.D., Dobrucký M., Problem Solving Environment for Flood Forecasting. *Proc. of The 7th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2003)*, July 2003, Orlando, USA pp. 350-355.

Kunszt, P., Laure, E., Stockinger, H. and Stockinger, K., Advanced Replica Management with Reptor. *In 5th International Conference on Parallel Processing and Applied Mathematics*, Czestochowa, Poland, September 7-10, 2003. Springer Verlag.

Joseph, S., Hoshiai, T., Decentralized Meta-data Strategies: Effective Peer-to-Peer Search. *IEICE Trans. Commun.*, Vol. E86-B, No. 6. June 2003.

Thomas, M., Mock, S., Boisseau, J., Dahan, M., Mueller, K., Sutton, D., The GridPort Toolkit Architecture for Building Grid Portals. *Proceedings of the 10th IEEE Intl. Symp. on High Perf. Dist. Comp.* August 2001.

CHEF Information site. *http://www.chefproject.org* (visited Dec. 2003)

The Jetspeed portal framework.
http://jakarta.apache.org/jetspeed/site/index.html (visited October, 2003)
Scientific Workflows Survey.
http://www.extreme.indiana.edu/swf-survey/ (visited December, 2003)