# Infrastructure for Grid-Based Virtual Organizations

L. Hluchy, O. Habala, V.D. Tran, B. Simo, J. Astalos, and M. Dobrucky

Institute of Informatics, Slovak Academy of Sciences
Dubravska cesta 9, 84507 Bratislava, Slovakia
`hluchy.ui@savba.sk`

**Abstract.** This paper presents architecture of a collaborative computation environment based on a Grid infrastructure, used as a 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. 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 for storing and accessing different computed and measured data, and a set of web portals.

## 1 Introduction

In recent years a number of scientific projects with international (even global) participation emerged as an answer to increasingly complicated problems of modern science – a well-organized business, with dense network of cooperation between people, organizations and countries. Such cooperation also requires an effective toolset for communication, experiment management and results sharing. The natural way to produce such a toolset is to develop a network-enabled software suite. Such software suites exist – although mainly incomplete and not mature – and in recent years are becoming more oriented toward the paradigm of virtual world-wide resource sharing – Grid computing.

## 2 The Architecture of a Grid Infrastructure for Virtual Organization

The basic perpetual cycle of work in a scientific virtual organization is simple – data is processed, another data is created. Of course, new data also enters the virtual organization and produced data is used, viewed, analyzed and interpreted to obtain results as the main purpose of the actual existence of the organization.

So we begin the analysis of the software infrastructure with the word data, and this tells us that the support of work with this data is one important part of the infrastructure – the data management. Another part of the infrastructure is in the core of the

cycle – the processing facility. This is the oldest part of any software and the very first PSEs were just a layer of control above such a computational core. Current efforts for computation management widely employ workflows (sequences of simpler tasks) and workflow management. This becomes important with the steadily increasing complexity of scientific computation, making a single integrated computational module too robust and inflexible.

To actually enable two or more scientists to work together towards achieving a common goal, a collaboration and communication suite integrated in the infrastructure is necessary. The word „integrated" means that this part is connected together with other controls available to the user.

```
                        ┌───────────┐
                        │   User    │
                        └─────┬─────┘
                              ▼
        ┌─────────────────────────────────────────────┐
        │              Control interface               │
        └──┬──────────────────┬──────────────────┬─────┘
           ▼                  ▼                  ▼
   ┌────────────┐      ┌──────────────┐   ┌──────────────────┐
   │  Workflow  │      │ Data management │ │ Collaborative tools │
   └────────────┘      └──────────────┘   └──────────────────┘
   ┌ ─ ─ ─ ─ ─ ┐      ┌ ─ ─ ─ ─ ─ ─ ┐   ┌ ─ ─ ─ ─ ─ ─ ─ ┐
     Models, libraries,   Datasets, metadata    Users, comm.
     solvers                                    devices
   └ ─ ─ ─ ─ ─ ┘      └ ─ ─ ─ ─ ─ ─ ┘   └ ─ ─ ─ ─ ─ ─ ─ ┘
        ┌─────────────────────────────────────────────┐
        │              Grid infrastructure             │
        └─────────────────────────────────────────────┘
```
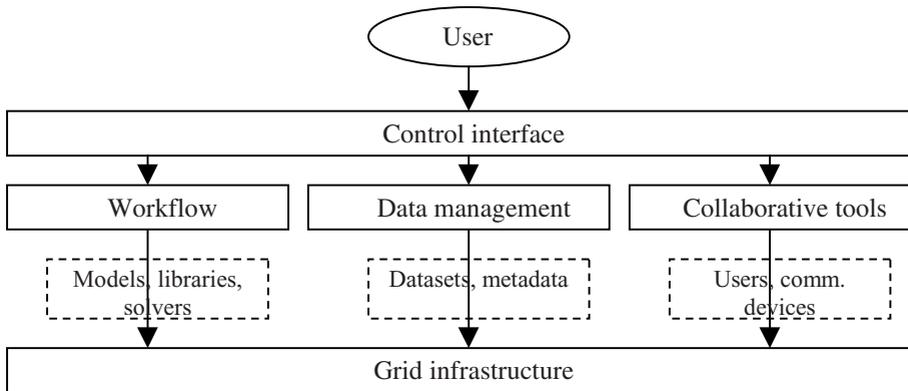
**Fig. 1.** Architecture of a Grid infrastructure for scientific VO

So we may say that the Fig. 1 is a basic and general architecture of a modern Grid-based collaboration environment. Many concepts inherent to the environment are abstracted and the data flow is also much more complicated, with parts of the workflow requiring access to data and data management tools, users accessing the stored datasets and all parts actively communicating with users.

## 3   Workflow Management

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. Similar to the definition of a workflow in a business process management [2], 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.

### 3.1  Existing Systems

Most of grid workflow systems being developed focus on the web services, although 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 [3] – 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 [4] 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 [5] web page. There are two main specifications of workflow languages: Web Services Flow Language (WSFL) [14] targeting web services and the Grid Services Flow Language (GSFL) [15], which builds upon Open Grid Services Architecture (OGSA) [16]. 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.

### 3.2  Workflow Management Components 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.

## 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, used also by us in the FloodVO portal design is CHEF [17].

CHEF (CompreHensive collaborativE Framework) is a collaborative environment based on Jespeed portal engine [6]. The collaborative tools (teamlets) are written as portlets in 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 effects 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 for Scientific Virtual Organizations

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). Thus, also the data management efforts are divided into two main streams – replica management and metadata storage/lookup.

## 5.1  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 duplicitly (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. The software developed in work package 2 of the EU IST DataGrid project [10][18] 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 [19] and OGSA [16] architecture. Anyway, it is lacking a modern and scalable metadata repository.

## 5.2  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 [20], 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.

## 6  Test Case – Virtual Organization for Flood Prediction in the CROSSGRID Project[1]

The efforts presented above are applied and tested in an international IST project CROSSGRID [13]. Collaborative tools, data management as well as workflow control have been employed and used in a way needed for flood prediction in the Grid-based Virtual Organization for Flood Prediction (FloodVO).

### 6.1  Workflow in FloodVO

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.

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 subcatchments.

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 instances of a workflow giving several results for the same time and area.

## 6.2   Data Management in FloodVO

The general schema of possible data sources for FloodVO operation was described in previous work [9]. 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 are radar images of current weather conditions, and the third type are the ground-based water level, precipitation and temperature measurements provided by SHMI's (Slovak Hydrometeorological Institute) network of measurement stations.

Data management in the prototype of FloodVO was implemented using the DataGrid software. The metadata database was implemented using the MySQL [11] RDBMS and an OGSA frontend. A service and a client application have been implemented. The client enables to add, modify, locate and delete metadata for given file in the FloodVO. The data management will be incorporated into the workflow and portal system in the near future. Users will be able to locate data and construct workflows based on metadata descriptions.

## 6.3   User Interfaces for Collaboration

There are three different user interfaces for the flood application. We have developed GridPort [22] 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.

### Application Portal Based on the Jetspeed Portal Framework
The Jetspeed [6] 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 (Fig. 2). This framework is also being used by other Grid projects such as Alliance portal [7] and the new version of the GridPort toolkit – GridPort 3.0 [8].

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 automatization 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.
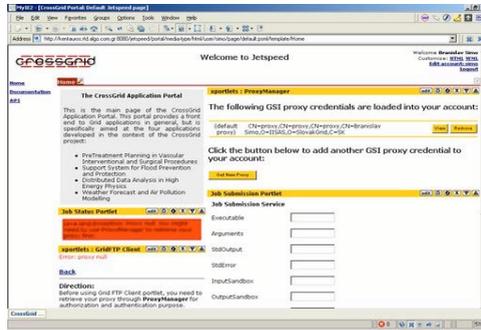


**Fig. 2.** Screenshot of the Jetspeed based application portal

**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. 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    Future Work

Many simulation models, in some cases, are not very reliable and are also dependent on many other factors (physical phenomena), which are not included in evaluation process of the models. Knowledge based treatment of historical data could provide enhanced functionality for the simulation models which strictly relies on the recent data sets. It also allows constructing several predicates of states according to knowledge evaluation with simulation run. More work will be also done in the area of workflow control and data management, where a more distributed and robust approach will be necessary.

# References

1.  Gallopoulos, S., Houstis, E., Rice, J.: Computer as Thinker/Doer: Problem Solving Environments for Computational Science. IEEE Computational Science and Engineering Magazine, 1994, Vol. 2, pp. 11-23.
2.  Workflow Management Coalition Terminology & Glossary.
    http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf (visited December, 2003)
3.  Condor DAGman. http://www.cs.wisc.edu/condor/dagman (visited December, 2003)
4.  Pegasus. http://www.isi.edu/~deelman/pegasus.htm (visited November, 2003)
5.  Scientific Workflows Survey. http://www.extreme.indiana.edu/swf-survey/ (visited December, 2003)
6.  Jetspeed. http://jakarta.apache.org/jetspeed/site/index.html (visited October, 2003)
7.  Alliance portal. http://www.extreme.indiana.edu/alliance/ (visited October, 2003)
8.  Grid Port 3.0 Plans presentation.
    http://www.nesc.ac.uk/talks/261/Tuesday/GP3%20HotPage%20Combined%20Edinburgh%20Presentation.ppt (visited December, 2003)
9.  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, Florida, USA, pp. 350-355.
10. Hoschek, W., et. al.: Data Management in the European DataGrid Project. The 2001 Globus Retreat, San Francisco, August 9-10 2001.
11. Widenius, M., Axmark, D.: MySQL Reference Manual. O'Reilly and Associates, June 2002, 814 pages.
12. Bell, W., et. al.: Project Spitfire - Towards Grid Web Service Databases. Technical report, Global Grid Forum Informational Document, GGF5, Edinburgh, Scotland, July 2002
13. Development of Grid Environment for Interactive Applications. IST-2001-32243. http://www.eu-crossgrid.org/ (visited December, 2003)
14. Web Services Flow Language.
    www.ibm.com/software/solutions/webservices/pdf/WSFL.pdf (visited October, 2003)
15. The Grid Services Flow Language (GSFL).
16. http://www-unix.globus.org/cog/projects/workflow/ (visited October, 2003)
17. Foster, I., Kesselman, C., Nick, J. M., Tuecke, S.: The Physiology of the Grid; An Open Grid Services Architecture for Distributed Systems Integration.
    http://www.globus.org/ogsa/ (visited December, 2003)
18. CHEF Information site. http://www.chefproject.org/ (visited December, 2003)
19. Peter Kunszt, Erwin Laure, Heinz Stockinger, and Kurt Stockinger. Advanced Replica Management with Reptor . In 5th International Conference on Parallel Processing and Applied Mathematics, Czestochowa, Poland, September 7-10, 2003. Springer Verlag.
20. W3C Web Services Activity web site. http://www.w3.org/2002/ws/ (visited December, 2003)
21. Joseph, S., Hoshiai, T.: Decentralized Meta-data Strategies: Effective Peer-to-Peer Search. IEICE Trans. Commun., Vol. E86-B, No. 6. June 2003.
22. W3C Resource Description Framework web site. http://www.w3.org/RDF/ (visited December, 2003)
23. 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.