

The Migrating Desktop as a GUI Framework for the “Applications on Demand” Concept

Mirosław Kupczyk, Rafał Lichwała, Norbert Meyer, Bartosz Palak,
Marcin Płóciennik, Maciej Stroiński, and Paweł Wolniewicz

Poznań Supercomputing and Networking Center, Noskowskiego 10, Poznań, Poland
{miron,syriusz,meyer,bartek,marcinp,stroins,pawelw}@man.poznan.pl

Abstract. The Migrating Desktop is a ready to use GUI framework for making use of grid applications and putting into practice the “application on demand” concept. On-demand computing, contrary to the traditional approach of assigning resources to applications, refers to the concept of pooling system resources and dynamically allocating them to meet shifting demands. We introduce a ready-to-use framework for the integration of different computing systems into a comfortable working environment with support for “applications on demand”. This work is done under the EU CrossGrid project IST-2001-32243.

1 Introduction

In this paper we describe the idea of using the Migrating Desktop as a ready-to-use GUI framework for making use of grid applications and also putting into practice a new generation concept of computing on demand. We focus on the architecture and functionality rather than technical details. This functionality refers to two different grid projects: the EU CrossGrid project IST-2001-32243 [1], and Progress (co-founded by Sun Microsystems and the Polish State Committee for Scientific Research) [2].

The “On Demand” concept is a term that covers several additional buzzwords like grid computing, utility computing and others. Utility computing is a combination of two approaches: according to the first one companies can call upon a third party to host and manage their IT infrastructure, and according to the second one, companies can pay for the resources they use. Grid computing is similar to utility computing but with a different approach. Grid computing is a form of virtualization that can handle computation-intensive tasks, using a large number of systems and combining them into one grid. Such grids can include widely distributed systems or systems within one data centers. Grid technology is enabling computing resources to be shared globally and easily managed, and the infrastructure becomes incredibly flexible. Nowadays the infrastructure is a pool of virtual resources that the user can call on as needed. Our main aim was to give the user a possibility of easy and flexible usage and gaining profits from new technology. We also introduce mechanisms that give an opportunity to look ahead to support future technologies.

We propose a transparent user work environment, independent of the system version and hardware. A flexible system structure enables defining and adaptation requirements of individual groups of users. We have to look ahead so that our users would not be bound to hardware configuration in order to have access to their working environment, applications, or resources. Our proposition is to give the user their own user-friendly environment with easy access to the applications and High Performing Computing (HPC) infrastructures on demand, independently of the location and hardware/software configuration. That is the reason why we have created the Migrating Desktop, which is a graphical user working environment (see Fig. 1.).

The Migrating Desktop is just a front end to the Roaming Access Server (RAS), which intermediates to communication and gives roaming access to different grid infrastructures. All user settings are stored in RAS therefore the Migrating Desktop started from another workstation looks the same. One of our aims is to have a common protocol for dealing with different infrastructures and to have one interface to the end users. The server implementing this interface is plug-in based, and plug-ins are responsible for communication with specific grid infrastructures. This architecture provides an important future direction with respect to the general acceptance of services and protocols. A grid infrastructure is being simultaneously developed in the framework of many academic and commercial projects. One of the biggest challenges for grid designers is managing the enormous complexity of Grid-based systems, making them interoperable with the existing systems and other emerging technologies and standards.

2 Architecture Overview

The architectures of the entire CrossGrid and Progress projects are presented in appropriate documents [3,4,5]. For the purpose of this paper we show only the basic components and interfaces between the corresponding modules.

The heart of the architecture is the Roaming Access Server that provides web services for accessing HPC resources.

The Roaming Access Server [3] offers a well-defined set of web-services that can be used as an interface for accessing HPC systems and services (based on various technologies) in a common, standardized way. All communication bases on web services technology.

The Roaming Access Server is a set of modules and plug-ins that provides interfaces to work with different grids. It consists of several independent parts responsible for application management, monitoring, user profile management, data management and authorization (see Fig. 2.). The provided functionality contains a wide range of different services that are common for many various grid projects. Currently the RAS provides plug-ins for interoperability with two separate grids - Polish Progress and EU CrossGrid; however, its infrastructure is open for expansion and attaching other HPC systems. The number of RAS servers depends on the amount of users and their requirements; it will be a fully replicated part of the system.

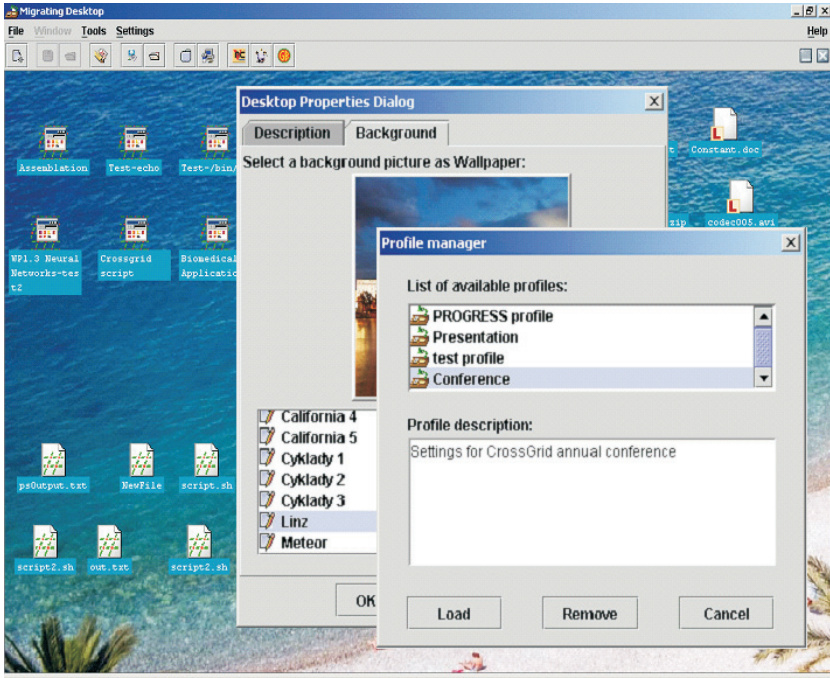


Fig. 1. Migrating Desktop main window and sample configuration dialogs

System architecture

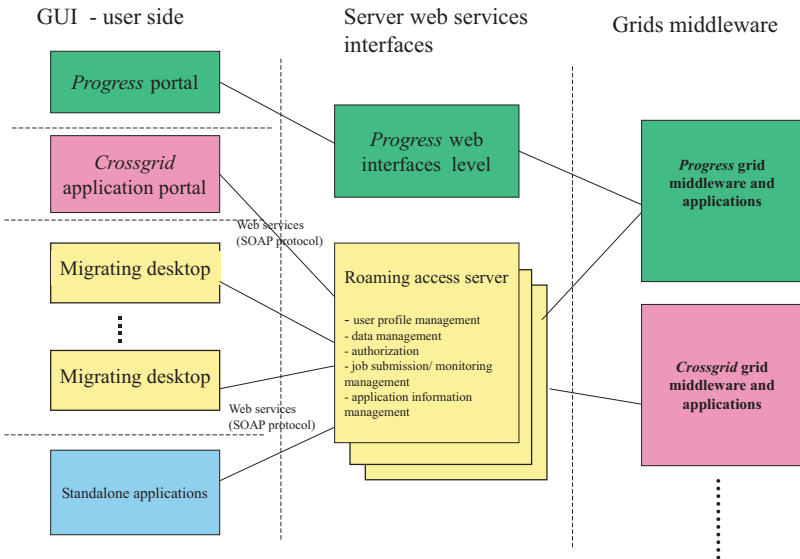


Fig. 2. RAS architecture overview

The Migrating Desktop is an advanced user-friendly environment that serves as a uniform grid working environment independent of specific grid infrastructure. Java based GUI is designed especially for mobile users and is independent of the platform (MS Windows, Linux, Solaris) and hardware. It is a complex environment that integrates many tools and allows working with many grids transparently and simultaneously. The main functionality concerns local and interactive grid and remote application support, local and grid file management, security assurance, authorization of access to resources and applications, and single sign-on technology based on X509 certificates.

A general concept of the Migrating Desktop was to provide the containers - frameworks for plug-ins written by application developers. Such approach allows increasing functionality in an easy way without the need of architecture changes. It is possible to add various tools and applications and support visualization of different formats using that mechanism. For example, the Migrating Desktop loads appropriate plug-in for visualization of a non-standard graphics format file on demand from the network. That makes our product independent of specialized tools designed only for a specific application.

3 “Applications on Demand” Services

Application Framework

The Migrating Desktop supports different kinds of accessing remote/HPC applications. We have focused mainly on supporting frameworks for grid applications, unifying different interfaces into one common solution. The Migrating Desktop calls appropriate web services on the RAS. The server job submission interface gives uniform access to different resource brokers. Clients can submit all jobs with this interface and then appropriate plug-ins are called to convert job description to the specific language. Currently we are using plug-in for CrossGrid/DataGrid resource broker to convert job description to Job Definition Language used in DataGrid and Progress plug-in to convert job description to Extended Resource Specification Language used in Progress. In the future Job submission interface will support XML job description, which is under development by the Job Submission Language group from the Global Grid Forum [6].

Grid applications available for users are grouped in a user-friendly way in a Job Wizard in the MD. This Wizard simplifies the process of specifying parameters and limits, suggesting user defaults or recently used parameters. The Wizard is responsible for proper preparation of the user’s job and consists of several panels. One panel is application specific plug-in and the rest can be used to set job information, resource requirements, files and environment variables. Application plug-in asks user for application parameters and that parameters are then passed to the command line. Application plug-in is just a Java class inherited from the ApplicationPluginBase and must implement its abstract method. The most important methods are getArguments and setArguments called when the job is going to be submitted or when job is presented to the user. For application developers that do not like to prepare plug-in we created generic XML

plug-in that interprets argument description given in XML to a graphical form. Example plug-ins are presented in Fig. 3.

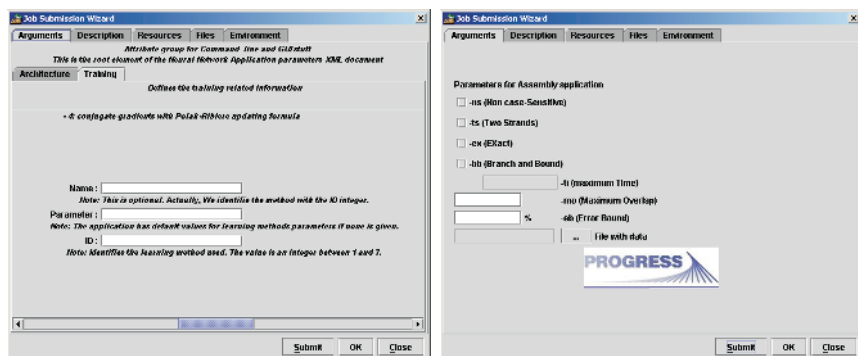


Fig. 3. Example Application plug-ins. Left - generated from XML, right - Java class

The Migrating Desktop also allows running remote Java tools. In contrast to grid applications that are submitted to the resource broker, tools are Java applets downloaded from different network localizations on demand.

Some applications require a graphical visualization of job results. In our approach visualization are just tools and can also base on the Tool Plug-in. Tool Plug-ins can visualize a single file or a single job. All available file and jobs visualization are registered in the Migrating Desktop database, and the appropriate Tool Plug-in is chosen depending on the file extension or type of the job.

We also provide a framework for interactive application as the VNC [7] plug-in. Interactive applications can be started and managed using the edg2.0 middleware. This middleware is developed in the DataGrid project and is still under enhancement in CrossGrid.

Using the VNC plug-in is also our proposed solution for other types of interactive remote applications from different providers. In our approach the user can have only a simple terminal, a freeware system with a web browser, and gain profits at an opportunity of using remote resources. An application instance run, for example, on a specialized application server could be transparently offered to the user on his/her demand. The user does not have to possess expensive products to benefit from the usage of its profits, he/she just uses time of that application making use of the next generation networks.

The features mentioned above give an easy way for including open and dynamic architectures of applications available on demand. Our framework supports a wide range of different remote applications types so it can fulfil growing demands of step-ahead technology users.

Application State Monitoring

Monitoring can be performed on different moments/levels of the application life. We assumed to present all monitoring types supported by application providers. At the moment we focus mainly on the grid jobs state monitoring framework.

Users can check the state of the launched application/submitted job, suspend or delete the selected job using the Job Monitoring tool in the MD. The Job Monitoring dialog presented in Fig. 4. is a useful tool for tracing the status of the previously submitted jobs. This dialog contains all the information about the submitted job including an extended job status and job log. Some information like the extended status can have different formats, because they are returned by different grid information systems. The RAS plug-ins read information from a specific grid information system and presents them to users in a text or XML format.

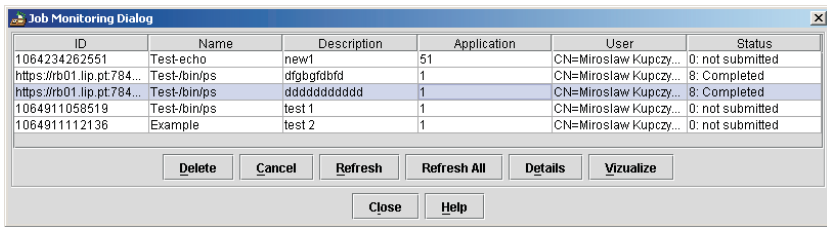


Fig. 4. Job Monitoring Dialog

Data Management

Data management is a very important and complicated part in every system. After many analysis a common set of interfaces that allow operation on data and metadata was created. The designed framework allows extending MD/RAS infrastructure easily so that many management systems could be attached as a plug-in. The main file Management tool in the Migrating Desktop is called Grid-Commander (Fig. 5.). The Grid Commander is a two-panel application similar to the Commander family tools. A single panel can represent a local directory, gridftp or ftp directory or other protocol to the native storage. Each protocol is defined as a plug-in that makes file management in MD easy to extend to other protocols. From the technical points of view graphical elements operate on generic file systems that invoke appropriate functions within the supported plug-in.

Authentication Issues

We try to keep authentication as simple for the user as possible. Authentication to the Migrating Desktop is done with users X509 certificates that have to reside on a local disk, removable disk, or can be downloaded from MyProxy server.

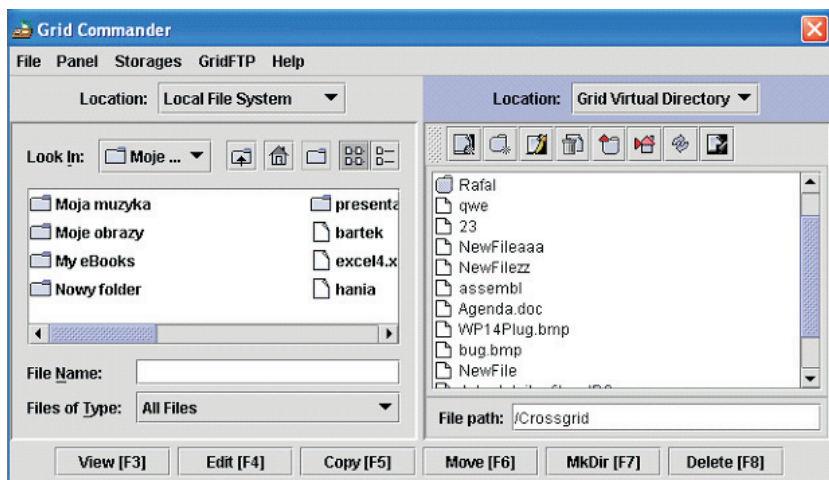


Fig. 5. Grid Commander Dialog

X509 proxy is passed to each RAS interface and is used to access grid resources. For resources that require other methods of authentication (e.g. username and password) the Migrating Desktop can ask about necessary information and then store it in the user profile. Any subsequent authentication can be done automatically. In our current implementation we store username and password for authentication to the Progress grid infrastructure which bases on the IPlanet Portal Server.

User Profile Management

User profile management provides functionality that allows operations on user profiles. User profiles contain all information that define the current user working environment including information about the graphical configuration (e.g. desktop background, icon locations, and colors) and information needed to access specific HPC infrastructures (e.g. username and password). The LDAP protocol is currently used for saving and retrieving stored information.

Other Services

We are going to extend the functionality of the Migrating Desktop/Roaming Access Server not only in the field of grid services. We will design a functionality that will simplify the communication between users for data exchanging. In scope of our research there are also ways of notifying the user via e-mails, SMS etc. about important events connected with the status of submitted jobs.

To add support for a new grid infrastructure the MD/RAS plug-in is necessary. Plug-ins provide systems specific information and are used by job/application submission, monitoring and data management modules.

4 Summary

In this paper we presented an overview of the functionality of the Migrating Desktop and Roaming Access Server. It consists of several components that are using the Web Service technology. It gathers all facilities that make the grid resources available and useful. We especially emphasized support of “application on-demand” concepts. We described a user-friendly framework that makes work with grids easy and comfortable. We have also pointed out advantages and features of the general framework dedicated for presentation of different providers’ applications. We have implemented a new generation product for the end user according to the growing demands and needs.

Currently Migrating Desktop is used for CrossGrid project as a graphical user interface for CrossGrid testbed. It was also used for Polish Progress project. After two years of development we can conclude that the idea of graphical environment for work with grid is accepted by users. “Application on demand” is our next goal we would like to achieve.

References

1. Development of Grid Environment for Interactive Applications. Annex 1, EUCrossGrid Project, IST-2001-32243, <http://www.eucrossgrid.org/CrossGridAnnex1.v31.pdf>
2. Polish Research on Grid Environment for SUN Servers, <http://progress.psnc.pl>
3. Bubak, M., Malawski, M., Zajac, K.: Current Status of the CrossGrid Architecture. Proceedings of the Cracow '02 Grid Workshop, Kraków, Poland, December 2002
4. Kosiedowski, M., Mazurek, C., Stroiński, M.: PROGRESS - Access Environment to Computational Services Performed by Cluster of Sun Systems. Proceedings of the Cracow '02 Grid Workshop, Kraków, Poland, December 2002
5. Kupczyk, M., Lichwała, R., Meyer, N., Palak, B., Płóciennik, M., Wolniewicz, P.: Mobile work environment for Grid Users. Proceedings of the Across Grids Conference, Santiago de Compostela, Spain, 2003
6. Job Submission Description Language Working Group, <http://www.epcc.ed.ac.uk/~ali/WORK/GGF/JSDL-WG/>
7. Richardson, T., Stafford-Fraser, Q., Wood, K.R. and Hopper, A.: Virtual Network Computing. *IEEE Internet Computing*, **2(1)**, (1998), 33–38