26 ANALYSIS OF REQUIREMENTS FOR COLLABORATIVE SCIENTIFIC EXPERIMENTATION ENVIRONMENTS

Ersin C. Kaletas, Hamideh Afsarmanesh, L. O. Hertzberger
University of Amsterdam, Informatics Institute, THE NETHERLANDS
{kaletas, hamideh, bob}@science.uva.nl

Scientists face a number of challenges when performing their complex experiments. Collaborative Experimentation Environment (CEE) addressed in this paper is a support environment for scientific experimentations, with an emphasis on supporting joint multi-disciplinary projects and collaborations. In order for a support infrastructure to help scientists tackle the challenging characteristics of their experiments, it must properly address their requirements. This paper presents the results of characterization and requirements analysis for CEEs towards supporting the collaborative activities of scientists, and introduces the extensions necessary for the VLAM-G scientific experimentation environment.

1. INTRODUCTION

Among the main challenging characteristics of emerging experiments in e-science domains, one can mention the complexity and diversity of experiments, the size and heterogeneity of data generated by these experiments, and the need for collaboration among heterogeneous and autonomous sites when performing joint experiments.

Several solutions have been proposed to support scientists with their complex experimentations. *Science Portals* (Ashby, 2001), (Pierce, 2002) only provide a single point of access with simplified interfaces to a specific set of resources that are of importance to a certain scientific community. A *Problem Solving Environment* (PSE), on the other hand (Allen, 2001), (Schuchardt, 2002) is a system that provides all the computational facilities needed to solve a specific target class of problems in a certain problem domain (Gallopoulos, 1994). Finally, a *Virtual Laboratory* (VL) (Afsarmanesh, 2002), (Messina, 2002) provides a generic electronic workspace for distributed collaboration and experimentation in research, to generate and deliver results using distributed ICT (Vary, 2000). It supports an aggregation of people who pursue a related set of research activities and share resources, where the resources including the people may be geographically distributed and associated with different institutions (Messina, 2002).

Collaborative Experimentation Environment (CEE) addressed in this paper is a support environment for scientific experimentations, which refers to a virtual

laboratory in its broadest sense, with an emphasis on supporting joint multi-disciplinary projects and collaboration, specifically information sharing among organizations and scientists. It is an integrated solution and support environment that addresses different aspects of experimentation and that supports scientists during the entire life cycle of experiments.

In order for a CEE to help scientists tackle the challenging characteristics of their experiments, it must properly address all their requirements. In a CEE, there are different types of users that perform different activities. Consequently, each of these users has different needs and expectations that mainly reflect the major activities they perform within the CEE. A detailed **characterization** of the CEE allows for the identification and characterization of both the different types of CEE users and the activities that they perform. Such a characterization leads to the identification of **user requirements**. Furthermore, a proper fulfillment of user requirements in turn puts a number of **ICT requirements** on the necessary base CEE infrastructure, which also need to be carefully analyzed.

This paper presents the results of characterization and requirements analysis for CEEs towards supporting the collaborative activities of scientists. In the remaining of this paper, first a characterization of CEE is provided, and the performed use case analysis is described. The paper then provides the results of a detailed analysis of requirements, with the focus on collaboration-related requirements. The collaborative extensions planned for the VLAM-G experimentation environment are introduced next. Finally, the paper presents its conclusions.

2. CEE CHARACTERIZATION

The Collaborative Experimentation Environment (CEE) is characterized in this section by distinguishing its major constituents; namely *experiments*, *users*, *data*, *functionality*, and *infrastructure*.

2.1 Experiment Characterization

The focus here is on the characterization of the life-cycle of a typical e-science experiment, which consists of three 'recursive' phases (**Figure 1**). During the **design phase**, the aim of the experiment is usually formulated as a question, and the methodology to answer this question is mapped to an experiment design. In the **execution phase**, the experimental procedure designed in the first phase is executed. It may include laboratory activities, using an instrument, or data gathering. During the last phase of **result analysis**, the data generated by the experiment is analyzed and interpreted by scientists. Several analysis tools can be used during this phase.

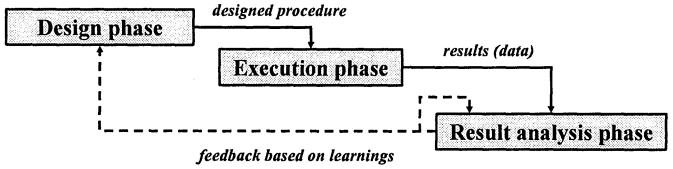

Figure 1. Life-cycle of a typical e-science experiment

The analysis phase is of ad-hoc nature and intuitive. Learning from the analysis results, the scientist may decide to use different analysis methods/algorithms on the same experiment results, or may decide to make a new experiment. The recursive nature of the experiment life-cycle comes from this last point.

2.2 User Characterization

Four target user groups are distinguished for the CEE (**Figure 2**). **Scientists** are the actual users of the CEE. A scientist is typically associated with an e-science domain (e.g. molecular biology). Inexperienced scientists usually follow a pre-defined procedure when making their experiments, while experienced scientists can also define new, customized procedures. **Domain experts** are scientists who have extensive knowledge and experience on a given e-science domain and on the experiments being performed in that domain. They are responsible, for instance, for modeling experimental information, designing experiment procedures, defining protocols to be used for certain activities in the laboratory, and defining parameters to be used for certain hardware and software. Another type of user for the CEE constitutes the **developers of support tools**. **Administrators** are responsible for the tasks related to the proper management and operation of the CEE, such as resource management, infrastructure maintenance, and user management.

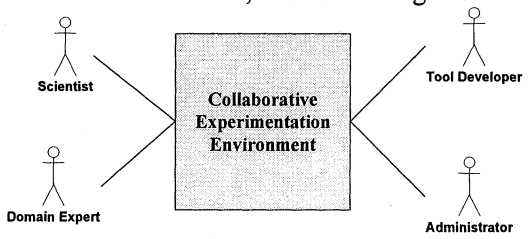

Figure 2. Users of the CEE

2.3 Data/Information Characterization

This section outlines the important aspects of data/information handled in a CEE (**Figure 3**). Large **size** of generated data is one of the most common characteristics of e-science experiments. However, the composition of generated results differs for each experiment. For instance, microarray experiments generate smaller size of results in comparison to material analysis experiments, but they are performed more frequently. **Storage** of data depends mainly on its size, structure, and usage. Large and/or unstructured data sets are generally stored in files, while structured data and/or data that needs to be queried are stored in databases. **Manipulation** of scientific data also varies from one experiment to another. Information generation can be step-wise over time, or at once. Information access can be on-demand basis for a single element, in the form of aggregate queries, or as data scanning. Information is usually **modeled** differently at each organization, following a quick-and-dirty approach, without considering standards, compatibility or possible future extensions. Scientific data is heterogeneous by nature. Among different types of **heterogeneity**, one can mention model/paradigm heterogeneity, data definition

and/or manipulation language heterogeneity, semantic heterogeneity, and system heterogeneity. **Interoperability** is important to support sharing and exchange of data among collaborating centers. In line with the different types of heterogeneity, interoperability must address syntactic interoperability, semantic interoperability, and system interoperability.

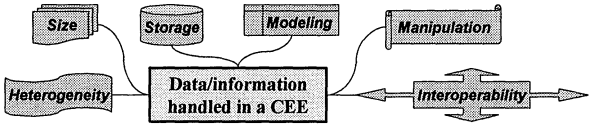

Figure 3. Characteristics of data/information handled in the CEE

2.4 Required Functionality Characterization

The following outlines the main functionalities required from a CEE (**Figure 4**): **Experiment management** (i.e. definitions, models, and mechanisms for managing an experiment during its entire life-cycle); **data/information management** (i.e. mechanisms for storage, querying, retrieval, and modification of wide variety of experiment-related information); **resource management** (i.e. efficient and coordinated management of resources needed and used during experiments); **user management** (i.e. definition and manipulation of users, roles, and their access rights); **security provision** (i.e. mechanisms for authentication of users and authorization of their requests); and **collaboration support** (i.e. support for collaborative activities among scientists, such as resource sharing, knowledge and experience sharing, and cooperative work among remote users).

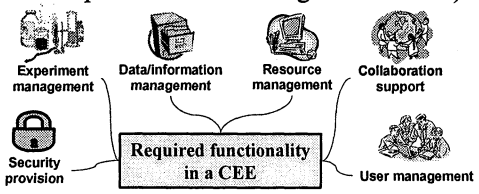

Figure 4. Required functionality from the CEE

2.5 Required Infrastructure Characterization

The infrastructure provided by the CEE must provide the necessary **computing facilities** for the analysis of large data sets (e.g. clusters of personal computers, virtual clusters), high-bandwidth/high-speed **networking facilities** for the transfer of data and/or processes among distributed computing, storage, or visualization facilities, and **software environment** that is open for adding new resources and scalable for coping with increasing number of users and workload.

3. USE CASE ANALYSIS

Use case modeling is a technique used to describe what a system should do. The primary components of a use case model are use cases, actors, and the system modeled (Eriksson, 1998). An *actor* is a person that interacts with the system. *Use cases* correspond to the main activities that an actor performs when interacting with the system. The *system* here corresponds to the CEE.

In addition to the users of the CEE (described in Section 2.2), another 'actor' of the CEE is the **ICT developer**, who develops the base infrastructure. The use cases identified for different CEE actors are provided in **Figure 5**. The use cases presented in this figure are high-level, mainly because e-science experiments are of ad-hoc and intuitive nature, where scientists may follow different routes and perform different activities within a use case.

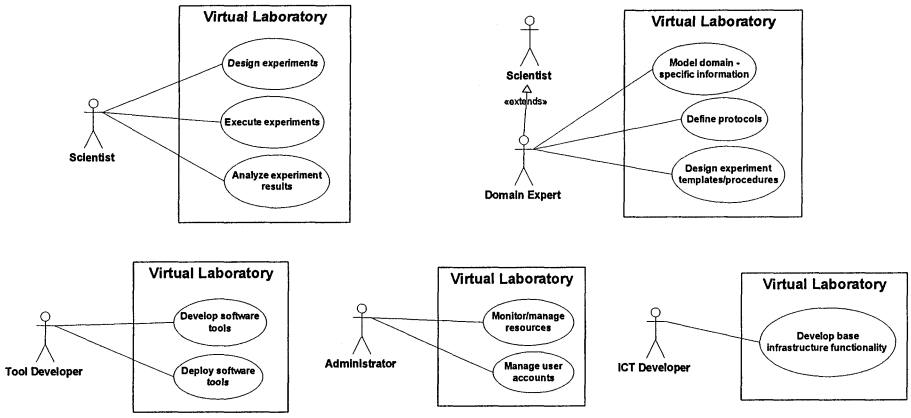

Figure 5. Use cases for CEE actors

4. CATEGORIZATION AND ANALYSIS OF REQUIREMENTS

4.1 Classification of Requirements

Based on the use case analysis, this section classifies all requirements into two categories, namely *user requirements* and *base ICT infrastructure requirements* (**Figure 6**). The former group is further classified into four groups corresponding to the different types of CEE users. The latter group is further classified into two, namely general CEE requirements and information management requirements. Identification and analysis of the base ICT infrastructure requirements constitute a first step towards providing a solution to user requirements. Therefore, in **Figure 6**, this relation is represented with block arrows from user requirements to base ICT requirements. This section presents the results of performed requirements analysis.

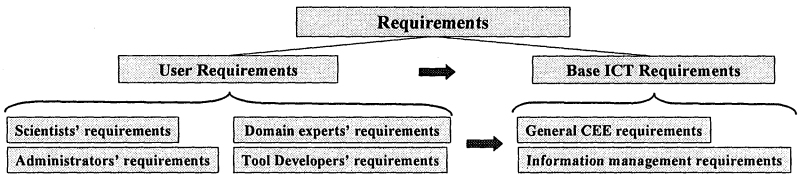

Figure 6. Classification of requirements

4.2 Analysis of User Requirements

The requirements analysis presented in this section aims to answer the following question: *What do the different CEE users require to properly perform their activities within the CEE?*

Scientists' requirements include the following:

- A proper means for clearly and sufficiently formulating the experiment objective/question in an experiment design
- A repository for experiment designs, results, and any other related information
- Necessary infrastructure for the execution of experiments
- Availability of both generic and most commonly used problem solvers
- Flexibility in the analysis phase that can cope with the ad-hoc and intuitive nature of analysis

In addition to scientists' requirements, **domain experts'** requirements include:

- A proper means for modeling standardized activities as protocols, and standardized experiment designs as templates, and a repository to store these
- Well-defined methodologies for modeling highly heterogeneous experiments and related data/information

Below are the **tool developers'** requirements:

- A design philosophy adopted by the CEE for software tools
- Well-documented and easy to understand CEE APIs

Finally, **administrators'** requirements include the following:

- Mechanisms for managing and monitoring the CEE resources
- A proper means for managing user accounts

Moreover, all users require graphical, easy and convenient to use, uniform interfaces for all their activities within the CEE, which are at the same time easy to customize to personal preferences.

Collaboration Related User Requirements

- Mechanisms for sharing and exchanging data and information, instruments, or other resources with collaborating partners in a secure environment
- Various technologies for cooperative work, such as video-conferencing or display sharing, supporting joint experimentation

- When needed, a proper means for easily adding new resources, stopping, re-starting, disconnecting a resource from the pool of resources, and enabling/disabling access to resources for a specific user or for a group of users
- Mechanisms for creating/removing user accounts, defining, updating, removing user roles, and defining and enforcing access rights for various resources

4.2 Analysis of Base ICT Infrastructure Requirements

In the previous section, needs and expectations of different CEE users when performing their activities in the CEE were presented (i.e. user requirements). A proper fulfillment of these user requirements in turn puts a number of requirements on the base ICT infrastructure for the CEE. Requirements analysis for the base ICT infrastructure presented in this section aims to answer the following question: *What functionality and facilities must be provided by the base ICT infrastructure to properly fulfill the requirements of different CEE users?*

General CEE Requirements

General CEE requirements are classified into the following categories: *Infrastructure requirements*, *functionality requirements*, *interface requirements*, and *architectural/technological implementation requirements*. Requirements in the first two categories were already outlined in Section 2 during CEE characterization; therefore, this section focuses on the remaining two categories.

User interfaces in general act as the entry-point of users to the underlying environment, while programming interfaces act as the entry-point of applications. **Interface requirements** include:

- User interfaces must hide the technical details and complexity of the underlying experimentation environment while supporting any possible usage of functionality provided by this environment
- User interfaces must allow an organized working environment, and ease the management and usage of diverse data and available resources and help the scientist to easily find what is where
- The CEE infrastructure must provide platform independent, uniform, well-documented, and easy-to-understand programming interfaces
- The programming interfaces must support the interoperation of domain-specific tools both with the CEE software environment and with other tools

Architectural design plays an important role on the scalability, openness, flexibility and manageability of the overall system. Following are the **architectural/technological requirements** related to the implementation of the base CEE infrastructure:

- The infrastructure must adopt a technical and architectural design philosophy for software development, information management, and resource management. The philosophy must be complemented with well-defined methodologies for each of these activities.
- The architecture must be open, flexible, and scalable to support interfacing with other systems, to improve, extend, or customize the provided functionality when

needed, to sustain a certain level of performance, to support collaboration, and to develop a number of monitors for managing and maintaining the system.
- The system implementation must exploit the existing and emerging standards as much as possible. However, compatibility of a technology with the CEE philosophy and methodologies, and its openness for any future improvements/extensions must be considered beforehand.

Information Management Requirements

In this subsection, the focus will be on the information management requirements for the base ICT infrastructure for CEE. Information management requirements are classified into the following categories: *Modeling requirements*, *storage requirements*, *manipulation requirements*, *security requirements*, *interoperability requirements*, and *implementation requirements*. The last category is already addressed as part of the general requirements; therefore it will not be included here.

Following are the **modeling requirements** for information handled in a CEE:
- Data models must be capable of properly representing the various types of information handled in the CEE.
- Data models must support modeling and representation of various types of experiments with different experiment flows and at any level of detail.
- Data models must be generic to achieve *uniformity* in representing both heterogeneous experiment types and heterogeneous data types.
- Schemas in the developed data models must be evolvable. They must be flexible for future changes, extendible for future extensions, open for customization to specific domains. Furthermore, the schemas must be compatible with the philosophy adopted by the CEE.

Storage requirements focus on the availability of databases for different types of information about experiments, e.g. for templates for the most common types of experiments, descriptions of previously made experiments, and descriptions of the most common techniques, protocols, etc. used in different types of experiments.

Requirements related to the **manipulation** of information include the following:
- Storage, access and manipulation mechanisms for various types of information must be developed, that are uniform within and across disciplines.
- Provided mechanisms must efficiently utilize the generality and expressiveness of the developed data models.
- Mechanisms for arbitrary queries must be provided.
- Mechanisms must be provided for version control.

Requirements related to the **security** of information are enumerated below:
- Mechanisms to define access rights for data security and information visibility must be made available to any user that owns some information in the CEE.
- All provided information management mechanisms must consider and enforce the access rights that are defined for the information that they manipulate.

Applying standards is among the **interoperability** requirements. In case of accessing multiple data sources, mechanisms to help/assist administrators to resolve model/paradigm heterogeneity or semantic heterogeneity must be provided.

Collaboration Related Base ICT Infrastructure Requirements

The base CEE infrastructure must support the collaborative activities among scientists. In specific, the VL infrastructure must address the following collaboration related requirements:

- Necessary infrastructure and mechanisms must be developed to enable sharing of resources, such as availability of a resource management system, information system for up-to-date status information, mechanisms for adding/removing a resource to/from the pool of shared resources, definition and maintenance of user account mappings, and definition and enforcement of usage rules. User and programming interfaces supporting all these mechanisms must be provided.
- Necessary data models, tools, and functionality/mechanisms must be provided to enable and ease the transfer of knowledge and experience sharing; for instance by making experiment templates, designs, protocols defined by expert users available to novice users, or through on-line (virtual) discussion environments among scientists. Such tools may utilize various technologies to support collaboration among partners: synchronous such as video-conferencing, display sharing/simultaneous visualization, joint sessions, etc. or asynchronous such as data exchange, sharing an instrument in another organization, etc.
- A proper infrastructure must be provided for coordination of joint distributed activities and for secure and authorized sharing of resources, which also considers the autonomy of collaborating organizations. Necessary data models and functionality/mechanisms must also be developed for the definition, management, and enforcement of collaboration rules.

8. VLAM-G COLLABORATIVE EXPERIMENTATION ENVIRONMENT

The Dutch VLAM-G (Grid-based Virtual Laboratory Amsterdam) project (Afsarmanesh, 2002), (Afsarmanesh, 2001) provides the main context for the work presented in this paper. VLAM-G is a multi-disciplinary virtual laboratory environment that provides the required generic environment for multi-disciplinary research in experimental science domains. VLAM-G allows its users to perform multi-disciplinary, collaborative experiments in a uniform, integrated environment, complement their in-vitro experiments with in-silico experiments, define customized experimental procedures and analysis flows, reuse generic software components, and share hardware, software, storage, networking resources as well as knowledge and experience.

The architecture of the VLAM-G and interaction among its components are shown in **Figure 7**. **Front-End** is the user environment of the VLAM-G, which presents the VLAM-G functionality to its users in a uniform way. **Session Manager** manages the active user sessions, and is responsible for coordinating the interactions

among the VLAM-G components. The distributed computing and networking resources on the Grid are made available to VLAM-G users through the **Run Time System** (RTS), which provides an API to encapsulate the Grid computing code within a simple interface. **Module Repository** is a persistent storage for binaries of software entities to be executed by the RTS. **VIMCO** is the information management platform of VLAM-G, and provides the necessary mechanisms for the manipulation of different types of experiment-related information.

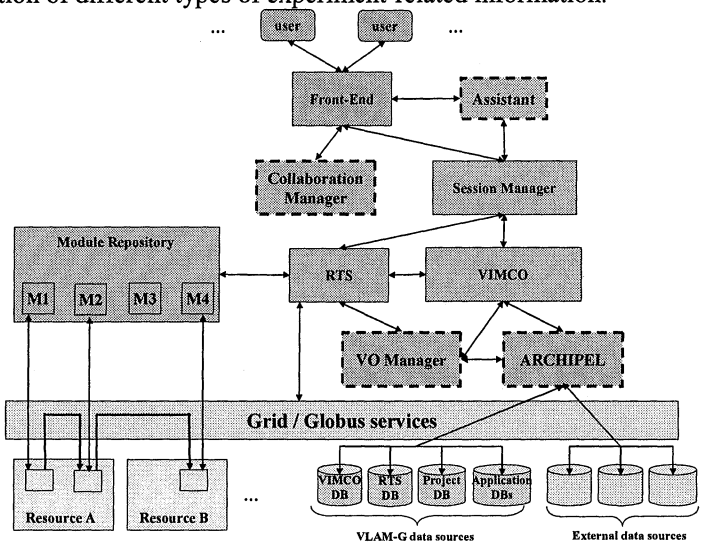

Figure 7. VLAM-G architecture extended with collaboration components

As mentioned earlier, collaboration is one of the main characteristics of e-science experiments. Emerging scientific experiments are evolving towards collaborative efforts involving several partners from different disciplines, different organizations, and different countries. With the increasing complexity and cost of scientific experiments, sharing expertise and sharing resources have become two of the most important motivations for collaboration. VLAM-G already addresses some of the main issues related to collaborative experimentation, such as multi-disciplinary projects, sharing (parts of) experiments, and basic mechanisms for controlling the collaboration (e.g. sharing policies to ensure the semantic consistency of the shared information and basic access rights). In addition, VLAM-G addresses sharing of software and hardware resources.

However, some of the collaboration requirements that were identified in this paper still need to be fulfilled. These requirements are integration of heterogeneous data from autonomous sources, setting and enforcing rules and regulations for a proper collaboration among partners within the context of a virtual organization, and supporting cooperative work among scientists. **Figure 7** shows the initial ideas on extending the VLAM-G architecture with four collaborative components, namely *Archipel*, *VO Manager*, *Collaboration Manager* and *Assistant*.

Archipel is a generic federated information management framework being designed within the context of the VLAM-G project, supporting uniform access to a variety of heterogeneous and distributed information sources. The VO support

infrastructure in VLAM-G, called **VO Manager**, is at the design stage (Kaletas, 2004), and it will make use of the other VLAM-G components; for instance, it will use the Archipel for sharing data resources and controlling access to these resources, or Grid for enforcing the sharing and access policies on hardware/software resources, as well as sharing the Grid security credentials needed to use these resources. **Collaboration Manager** will enable simultaneous collaborative design and execution of experiments through cooperative work environments (e.g. chatboxes). Finally, the **Assistant** will assist users during their experiments, for instance, by suggesting the most efficient software to perform a specific task.

10. CONCLUSIONS

In this paper, the CEE solution to support scientific experimentations was characterized and different types of CEE users and the activities that they perform within the CEE were identified and described. Each of these users performs different activities in the CEE, hence they have different needs and expectations from the CEE. In addition to user requirements, requirements for the base ICT infrastructure underlying the CEE were presented, with particular attention on collaboration requirements. Analysis of requirements showed that many requirements are related to each other. User requirements in turn impose a number of requirements on the base ICT infrastructure for CEE. The ICT requirements represent a first step towards providing a solution to user requirements. ICT developers must address these requirements to provide the necessary environment and functionality to CEE users.

As these requirements point out, there are several different aspects that need to be addressed related to collaboration, including VO support infrastructure, federated data access and integration, and cooperative work. With the existence of a collaboration support infrastructure, a number of organizations can join together, sharing their resources and skills towards reaching common goals. As applied to the scientific collaboration domain, VO paradigm can assist organizations in pursuing a common goal, for instance, tight collaboration towards solving scientific problems, where the sub-tasks are distributed among different organizations and the distributed multitasking is coordinated by the VO Manager. As the base necessity in the VO, it will be possible to share (access) privileges on all kinds of resources, from hardware and software to data and information. Furthermore, this sharing and collaboration is regulated by pre-defined sets of rules and policies, which are agreed upon by all collaborating partners. These agreements in form of contracts will further increase the trust among partners, and help them to advance their collaborations.

11. REFERENCES

1. Afsarmanesh H, Kaletas EC, Benabdelkader A, Garita C, Hertzberger LO. A Reference Architecture for Scientific Virtual Laboratories. Future Generation Computer Systems 2001; 17 (8): 999-1008.
2. Afsarmanesh H, Belleman RG, Belloum ASZ, Benabdelkader A, van den Brand JFJ, Eijkel GB, Frenkel A, Garita C, Groep DL, Heeren RMA, Hendrikse ZW, Hertzberger LO, Kaandorp JA, Kaletas EC, Korkhov V, de Laat CTAM, Sloot PMA, Vasunin D, Visser A, Yakali HH. VLAM-G: A Grid-based Virtual Laboratory. Scientific Programming 2002; 10 (2): 173-181.

3. Allen G, Benger W, Dramlitsch T, Goodale T, Hege HC, Lanfermann G, Merzky A, Radke T, Seidel E, Shalf J. Cactus Tools for Grid Applications. Cluster Computing 2001; 4 (3): 179-188
4. Ashby JV, Bicarregui JC, Boyd DRS, Kleese-van Dam K, Lambert SC, Matthews BM, O'Neill KD. A Multidisicplinary Scientific Data Portal. In Proceedings of the 9th International Conference and Exhibition on High-Performance Computing and Networking, 2001, pp. 13-22.
5. Eriksson H-E, Penker M. UML Toolkit. Wiley Computer Publishing, 1998.
6. Gallopoulos E, Houstis E, Rice JR. Computer as Thinker/Doer: Problem-Solving Environments for Computational Science. Computing in Science and Engineering 1994; 1 (2): 11-23.
7. Kaletas EC, Afsarmanesh H, Hertzberger LO. A Collaborative Experimentation Environment for Biosciences. (To appear in) International Journal of Networking and Virtual Organizations 2004.
8. Messina P. The Emergence of Virtual Laboratories for Science and Engineering. iGrid2002 Presentation. http://www.igrid2002.org/ppt/Paul_Messina.ppt
9. Pierce M, Youn C, Fox GC. The Gateway Computational Web Portal. Concurrency and Computation: Practice and Experience 2002; 14 (13-15): 1411-1426.
10. Schuchardt KL, Myers JD, Stephan EG. A Web-Based Data Architecture for Problem Solving Environments: Application of Distributed Authoring and Versioning to the Extensible Computational Chemistry Environment. Cluster Computing 2002; 5 (3): 287-296.
11. Vary JP. Report of the Expert Meeting on Virtual Laboratories. United Nations Educational, Scientific and Cultural Organization 2000; Technical Report CII-00/WS/01.