Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation, and Mathematics

PROCEEDINGS

17th International Conference on Process Control 2009

Hotel Baník, Štrbské Pleso, Slovakia, June 9 – 12, 2009

ISBN 978-80-227-3081-5

http://www.kirp.chtf.stuba.sk/pc09

Editors: M. Fikar and M. Kvasnica

Schmid, C.: About Grid Supported Learning Environments and Collaborative Virtual Control Laboratories, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 17th International Conference on Process Control '09*, Štrbské Pleso, Slovakia, 554–566, 2009.

Full paper online: http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/072.html

ABOUT GRID SUPPORTED LEARNING ENVIRONMENTS
AND COLLABORATIVE VIRTUAL CONTROL LABORATORIES

C. Schmid*

Ruhr Universität-Bochum,
Institute of Automation and Computer Control,
44780 Bochum, Germany
fax : +49 234/32-04093; e-mail : cs@atp.rub.de

Abstract: In this paper, grid technologies are introduced to build e-learning environments for virtual laboratories to be used in control education. Service-oriented grids open new fields of applications, the Learning Grids. The learning services concept and their deployment through grid technologies are excellent means to integrate virtual laboratories into collaborative e-learning environments for control engineering education. An introduction into this area, a review of the grid techniques and example applications from a virtual laboratory demonstrate grid based solutions.

Keywords: Collaboration, Engineering education, Learning Grid, Learning systems, Virtual laboratory.

1 INTRODUCTION

The current generation of electronic learning (e-learning) solutions has adopted the rather narrow pedagogic paradigm of information transfer, which features the teacher as someone who selects particular pieces of information and makes them available to students on the Web. However, there is no evidence that this approach to technology enhanced learning is in anyway effective. It has been adopted simply because it is an easy way to use the Web's facilities. As soon as the learner tries to mediate the personal knowledge with the others, to compare the results, and finally tries to use the synergy between personal and collective construction of knowledge, he has to fall back to conventional communication channels. The current solutions mainly offer textual, or conference tools at the most.

Remote or virtual laboratories with real or simulated experiments are becoming accepted in the engineering community for providing distance education and for augmenting traditional laboratories. Students have to modify instruments for a better understanding of the principle on which the plant operates. They even have to set their personal or within groups negotiated conditions in an environment for collaborative experimenting. From a pedagogical point of view, in this kind of environments the student has an active and central role in the learning process. In keeping them at the centre of the learning process, personalisation, individualisation and collaboration become relevant aspects to be supported by technologies through the creation of the right context. A learning grid can contribute to the achievement of these objectives through the definition of the learning services concept and their deployment through grid technologies.

In the following an introduction into grid computing and into the Virtual Control Laboratory (VCLab) is given. Section II describes the transition from computing to service-oriented grid and describes its main properties. Section III introduces into the specific properties of Learning Grids and Section IV goes into details of the implementation of main features of a Learning Grid using VCLab examples. Section V extends this on the area of collaborative learning in a grid supported environment and Section VI summarizes the results.

2 WHAT IS THE GRID?

The term grid is used here as a paradigm of the commonly known concept of power grids, where the consumer is not aware where and how the power is ex-

actly produced. He only receives the final product with a defined quality of service from the plug in the wall. In case of a computational grid, the client receives the computational power not knowing where it comes from and what the resources are. When it comes to a service-oriented grid, the user receives the functionality he needs with the desired quality of service.

Historically the term grid has been used describing a worldwide communication infrastructure for clustered computers, the nodes, that allows seamless transparent access to data and computing power on demand to solve large-scale computational problems. Such computing grids cost a fraction of what a supercomputer costs. They are commonly known from engineering, science and commerce. Grid is also a new paradigm for the information technology. The well known World Wide Web will be succeeded by the upcoming World Wide Grid. In this context a new type of grids, the service-oriented grids find applications in quite new areas not previously considered as the environments for a grid. An example of such a new area is education. This is the topic addressed in this article

2.1 Grid basics

From a general point of view, a grid is considered as a collection of clustered computational machines, the nodes. In order to have a powerful supercomputer by a grid the computational problem has to be split into slices and assigned to these nodes. Each node processes its slice individually and after the completion of its slice the results are put back together. Grid nodes do not need to be placed in one geographic location; moreover, machines collaborating in the grid may have different architectures and operating systems. It is obvious that these nodes need to communicate with each other based on some standards. Therefore a vital topic of security is involved for the interchange of data between nodes. Depending on the application the data should be kept confidential and protected from undesired external changes. Also other issues must be addressed, e.g. redundancy of nodes, quality of service and scalability.

A grid shows some limitations and has to fulfil some requirements. The grid is applicable only for tasks that can be easily split into smaller slices and that do not require the characteristics of a real-time challenge. In order to reduce the complexity of a grid, a special layer is introduced that is for gluing the nodes on a logical level. This layer of software sandwiched between the operating system and the applications is commonly called middleware. Its spectrum ranges from execution environments responsible for the management of processes on nodes, to full development environments. What traditional grids lack, are the standards on that they are built. In most cases when considering computational grids, the methods of communication, the level of integrity between

nodes and the architectures are each specially designed for a particular project.

2.2 Service oriented grid

During recent years a new approach for building grids has emerged. Instead of perceiving the grid nodes only as computational elements of an infrastructure they became providers of services (Foster et al. 2001). This shift, from strict computational capabilities to service suppliers, opens new fields of applications for grids. The nodes, instead of only delivering their computational and storage capacity, are now regarded as providers of particular services. They may be parts of some code existing in multiple instances allowing the parallelization of the execution of an application. The nodes may offer individual services best suited to their own capabilities. Moreover, services developed for the usage in one application or grid may be reused in new applications. The service-oriented approach has additional advantages. It introduces well-defined standards, allows the creation of searchable catalogues of services. Further details are described in Section 2.3.

Figure 1 presents basic interactions between elements of a service-oriented grid. Services published into a Grid Registry are queried and when discovered then instantiated depending on the user request. Mainly for sake of efficiency the client's communication with the service is direct but may also be virtualized.
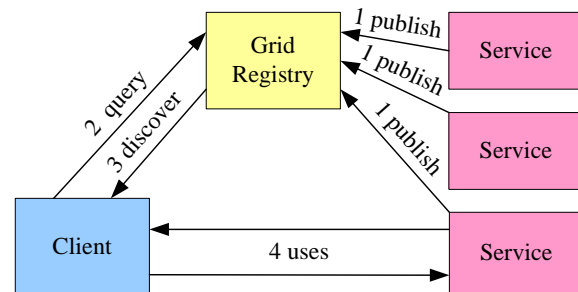

Fig. 1.   Structure of a service-oriented grid environment.

2.3 Technology and standards

The realization of service-oriented grids needs clear standards to have that interoperability of grid elements and their reuse in other applications. The two main organizations involved in standardization of grid technologies are the Open Grid Forum (OGF 2009) favouring the family of the Open Grid Services Architecture (OGSA 2009) standards and to some degree the competing Organization for the Advancement of Structured Information Standards (OASIS 2009) promoting the Web Services Resource Framework (WSRF 2009) standards. Both organizations adopt the currently widely recognized Web services and their extensions as their building blocks. These families of standards differ in the depth of the middleware integration, in the choice of the platform

and in the programmatic languages of the implementation. But their general approach towards the grid is the same.

The main functionalities delivered by the middleware of a service-oriented grid are:

Location – allows the determination, whether the required service exists and at which locations it is accessible

Instantiation – allows the instantiation of the service on that host, which matches the capabilities required for the service running with a given quality of service.

Orchestration – allows the dynamical composition of more complex services.

In the examples shown in this paper, the middleware called GrASP (Vallés et al. 2003) is used, which was developed in an EU funded project. It follows the OASIS recommendations based on the implementation of the WSRF called WSRF.NET, which uses Microsoft's .NET Framework as the implementation environment (WSRF.NET 2009).

3 LEARNING GRIDS

Grids yield significant benefits to applications. The question to be answered here is what advantages may yield a grid particularly to educational systems.

3.1 *Learning Objects (LOs) and Units of Learning (UoLs) in a grid environment*

In the concept of using LOs the learning content is split into reusable elements. These elements are used to build complex learning resources. In the world of service-oriented grids the LOs are becoming fully functional services with their own user interface. They are independently interoperable blocks, which may be used as they are, or, moreover, are reused to build new more complex blocks using other grid services, e.g. orchestration. LOs themselves can be nested. For illustration consider the complex LO example from Figure 2. Delivering a nested LO for an experiment several components are necessary and each of them is implemented as a separate LO. The required components would be: the LO rendering the experiment environment, the LO displaying an Excel worksheet for evaluating results and a scope LO displaying the experiment signal histories. These components would be embedded into another LO, therefore constituting a new composed unit called e.g.: Experimenting 1, which itself could be nested in a more general LO. Due to the well-defined grid standards, the learning courses can be built from LOs delivered by different grid services. The grid techniques offer the capabilities of cataloguing and easy managing LOs by using metadata.
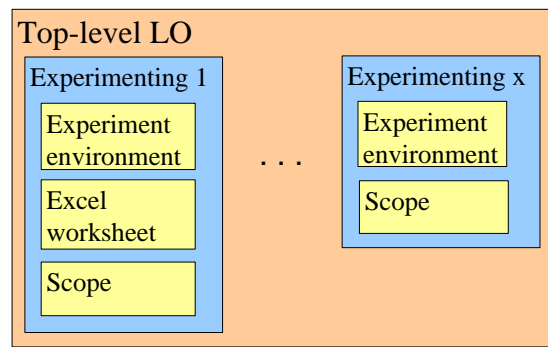

Fig. 2.   Example showing an arbitrary nesting of LOs.

Metadata for describing LOs and ontologies for the semantic modelling of the learning domain can be used to build and execute distributed learning applications on a Learning Grid. They take the form of UoLs. Each UoL is described by an ontology, which defines the set of concepts to be taught. Each concept corresponds to a Learning Object (LO) that constitutes the learning material. The ontology describes the order in which the LO will be delivered to achieve the teaching goal for a learner. Each UoL realizes a certain learning model, which is modelled by the ontology. The LOs represented by UoLs and combined with the learner specific requirements are being delivered personalized by a suitable IMS-LD player.

3.2 *Collaboration and communities*

The use of a common platform allows a better collaboration, both in sense of interpersonal communication for collaborative learning, as well as collaboration between applications existing within the grid. A Learning Grid is a natural environment for its participants to create virtual learning communities for collaborative experimenting. All participants belong to the same community of grid users sharing the same tools, creating and sustaining professional relationships through time.

3.3 *Scalability*

An outstanding advantage of a Learning Grid environment is the approximately linear scalability inherited from its predecessor, the computing grid. When the number of students enrolled to a particular course gets larger, more instances of a particular service will be created on the hosts within the grid. When additional hosts are needed they do not have to belong to the same university or run the same operating system as long the services are implementing the same interface. Figure 3 illustrates the characteristics of the average response time observed when only one node or n nodes are available on the grid related to a particular service. Grids may grow from few resources to millions. In principle, there is no restriction in size, but the availability and latency of resources must be observed.
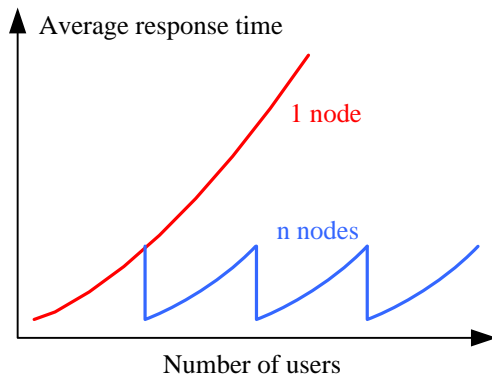
Fig. 3. Average response time characteristic depending on the number of users of a particular service available from one or more than three nodes on a grid.

3.4 *Personalisation*

A very important feature of a Learning Grid is the fact that it can deliver learning contents from heterogeneous resources in a unified fashion and personalised according to the profile of the learner. The following procedure is in analogy to the power grid. A learner with a well-defined profile introduces himself to the Learning Grid and requests some contents relevant to his learning needs. The Learning Grid starts here to find the best suitable service for the learner's needs. This would match closely as possible the user's profile taking in account the user's location, language skills, level of advancement in selected topic, preferred form of delivering content, etc.

3.5 *Virtual Organizations (VOs)*

One of the main advantages of the new grid technologies is their capability to integrate heterogeneous environments to an abstract entity. This property can be used to group resources of different universities to build a VO, e.g. a virtual university. Such an approach would allow specialization of universities in concrete areas and sharing the best offer with other universities.

4 VCLAB AS A GRID SUPPORTED VIRTUAL LABORATORY ENVIRONMENT

The VCLab (Schmid 1998) has been originally developed as a generic tool to support students in control engineering using professional design and simulations of process automation. It uses a 3D virtual user environment to recreate and to visualize experimenting plants. One can interact with a displayed scene in a similar fashion like with real devices. The dynamical behaviour of the plant is generated by a simulator driven by simulation models. VCLab has in its repository the generic components and services necessary for building LOs for experiments on a Learning Grid of the control engineering domain.

The grid supported implementation of VCLab is an appropriate case study to present the introduced properties of service-oriented grids in a practical manner to the control community.

4.1 *Architectural perspective of the grid implementation*

VCLab is deployed on the ELeGI grid (ELeGI 2009) by the IWT (Capuano et al. 2006), which is a content and user management system. For learners registered through its portal, it provides a personalized profile, which must be compatible with the IMS Learning Design specification (IMS-LD 2009). These data describe the learner as well his preferences regarding the social context and learning styles. From the technical point of view, the profile is a standardized data structure. It is being used in two ways. First, it allows making a better choice regarding the content that should be provided to the learner; second, it also allows adjusting the selected content depending on the learner preferences.

The VCLab components are integrated as LOs into the IWT framework by using drivers. The procedure of the instantiation of VCLab components as LOs is presented in Figure 4. When the learner decides to take part in a course that uses VCLab resources, the portal receives a request to create the instance of the particular LO. The portal combines the original request with data withdrawn from the learner's profile and submits the created request document to the grid, in this instance to the GrASP middleware. GrASP searches for the best service and best service provider matching the request and instantiates the driver service. The driver service returns the portlet which renders the LO for the learner depending on the personalization definitions in the learner's profile.

In order to understand the benefits of the VCLab on the grid, the architectural perspective of implementation and the interactions between the services are presented. Figure 5 shows the services, on which the functionality of the VCLab is based and their interrelations with the elements of the grid infrastructure. VCLab consists of two services, Computation and Simulation. Their operation is orchestrated by the Driver Service, which implements that interface necessary for an e-learning application.

The Grid Catalogue plays a central role on the grid, which contains the metadata describing both, the services incorporated within the grid and the stored resources.

When a user is accessing a grid application, it identifies the user and obtains his profile. Then the application uses the Grid Catalogue to provide the list of the resources matching the learner queries and the profile. When the user chooses a UoL to work with, the e-learning application contacts the Grid Catalogue, which provides the list of services required to deliver the resource to the user. In the case of VCLab
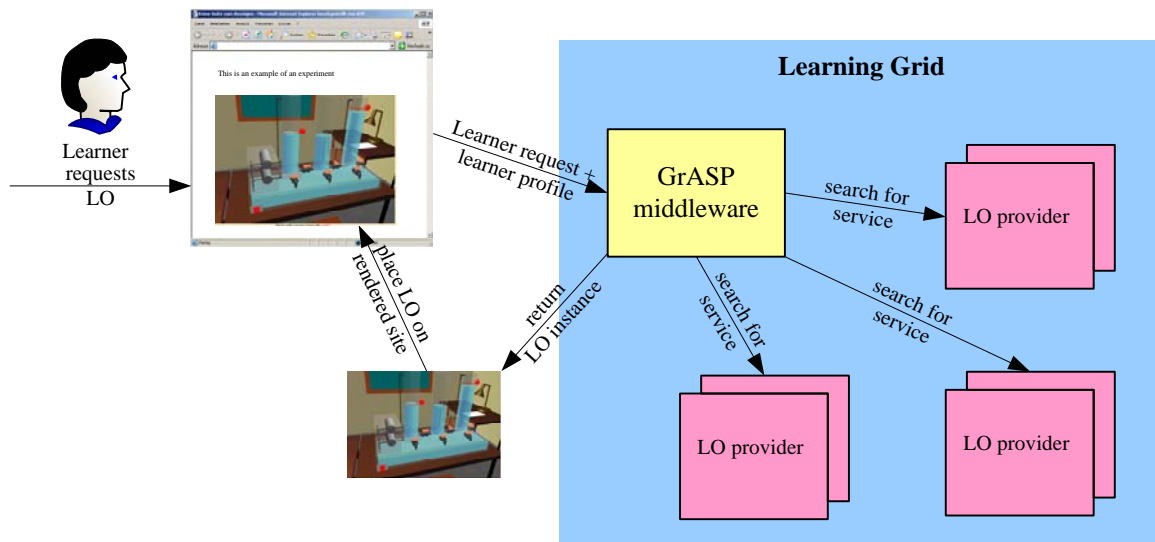
Fig. 4.   Process of instantiation of a Learning Object from VCLab in a Learning Grid.

such a service is the Driver Service shown in Figure 5, which implements a particular interface understood by the e-learning application. Such a driver service takes the control over the execution of the resource and requests the required services from the catalogue. In the VCLab case these are the Simulation and Computation Service within a Simulation LO.

This grid architecture shows a high degree of separation of services during the implementation and favours the extensibility of the application itself. The e-learning application needs not to be aware of the services used by VCLab; it only needs to find a suitable driver for a particular LO to be delivered.

4.2   *The Learning Model for Experimentation*

Achieving a successful integration is not only performed on a technological level, but also in regard to pedagogical aspects. For this purpose a generic model for the delivery of virtual experiments has been developed, which can be applied to control engineering experiments. This model splits a UoL into four macro phases: Presentation, Practical Situation, Abstract Situation and Institutionalization phase, see Figure 6.

The phase of Presentation provides the description of the didactic experience that the learner is about to start. To such aim, the description of the different phases, the necessary information for the learner about the character and goals of the experiment and about the general reference regarding the operation of the software will be provided.
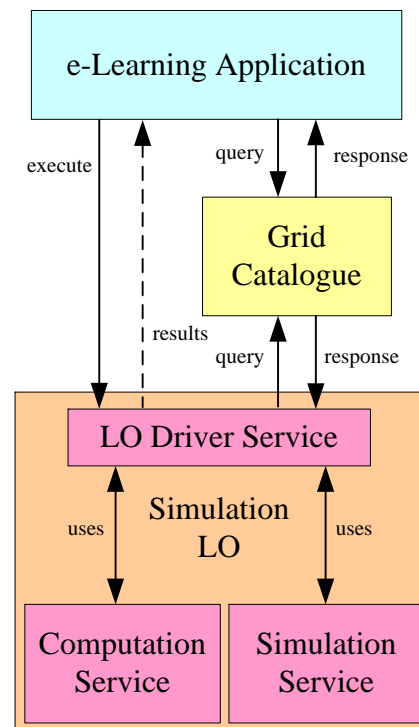

Fig. 5.   VCLab grid architecture.

The Practical Situation represents the phase in which the learner live the concrete experience. Simulation and the presence of a collaborative environment are available in which the personal learners' experience can be mediated from the interaction with the other learners. This phase has an iterative character and consists of five micro phases:

Active Situation – A fascinating and interactive scene in 3D is proposed, inside of which the learner will be able to move and manipulate objects. Simulations are run by a series of controls that the learner can opportunely vary, modifying in real time the behaviour of the simulation, observing its response and actively gaining personal knowledge.
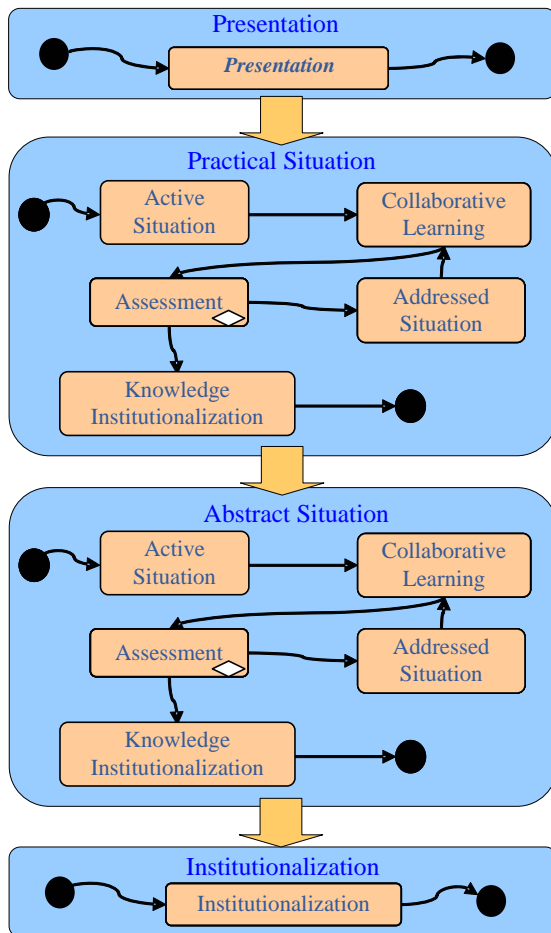
Fig. 6.   Learning model for experimentation.

Collaboration – During this phase the learner has the possibility to mediate the personal knowledge with the others, to compare the results, and finally use the synergy between personal and collective construction of knowledge.

Assessment – This micro phase marks the transition from action to opinion by giving the learners a variety of questions to judge the current validity of the learning process. If the output is not adequate a possibility is offered to enter in a facilitated didactic situation, which leads to the phase of the Addressed Situation. The learner can enter again into the phase of Active Situation or Collaboration in order fill own gaps. This ends in a further assessment with a loop back if not successful. Otherwise the phase of Knowledge Institutionalization is entered.

Addressed Situation – This optional phase, to which learners may be redirected in case of an unsuccessful Assessment, may provide an altered version of the Active Situation and give additional hints which should allow a facilitated understanding of the experiment.

Knowledge Institutionalization – It is the last micro phase of the Practical Situation when the knowledge validity is shown to the learner with a correct solution and a list of concepts which should be known after completing this activity.

The Abstract Situation macro phase is to extrapolate from the previously context an abstract model. It consists of the same micro phases as the Practical Situation and its execution is governed by the same rules. But instead of the simulation of a concrete case the activities will be set up on a greater interaction between theory and practice to induce the learner to test knowledge in order to achieve new goals. For example instead of a 3D scene in the Active Situation, e.g. the learner has to deal with a set of equations describing the experiment.

Finally the macro phase of Institutionalization provides the means for organizing and formalizing the acquired knowledge.

4.3   *UoLs for VCLab*

Implementing this learning model using the LO and UoL paradigm allows the building of a library of reusable learning units. Each object described by metadata is being easily catalogued and can even be dynamically bound for learning content delivery. UoLs are described by an ontology, which defines the set of concepts to be taught. Each concept contains LOs that constitute the learning material. In a situation, when a teacher has to explain to students how process control works, the plant is described by a LO. The teacher has a manuscript and makes a storyboard with items to be taught. To achieve a successful teaching process those items are structured according to phases of the learning model presented above.

The content managed by the IWT portal uses IMS-LD standardized forms of UoLs. This introduces an abstract layer over the technological aspects of resources in which authors can create their learning content in pedagogical instead of technical terms. This specification describes the recipients of a particular resource, the resource requirements in the sense of services required for executions, and the dependencies between resources.

For VCLab each UoL is described by an ontology, which defines the set of concepts to be taught. Each concept corresponds to a LO that constitutes the learning material. The macro phases from Section 4.2 are delivered as a single LO or as a series of LOs. The ontology describes the order in which the LO will be delivered to achieve the teaching goal for a learner. Each UoL realizes the learning model, which is modelled by the ontology. The LOs represented by UoLs and combined on the fly with the learner specific requirements are being delivered personalized by the IMS-LD player.

A section of an active learning session example with LOs from VCLab is shown in Figure 7. A web browser is used and the screenshot shows a section from the beginning of a nested Active Situation LO. Other sections, like assessment and addressed situation are not shown here. The example is taken from a beginner UoL for modelling Torricelli's law.
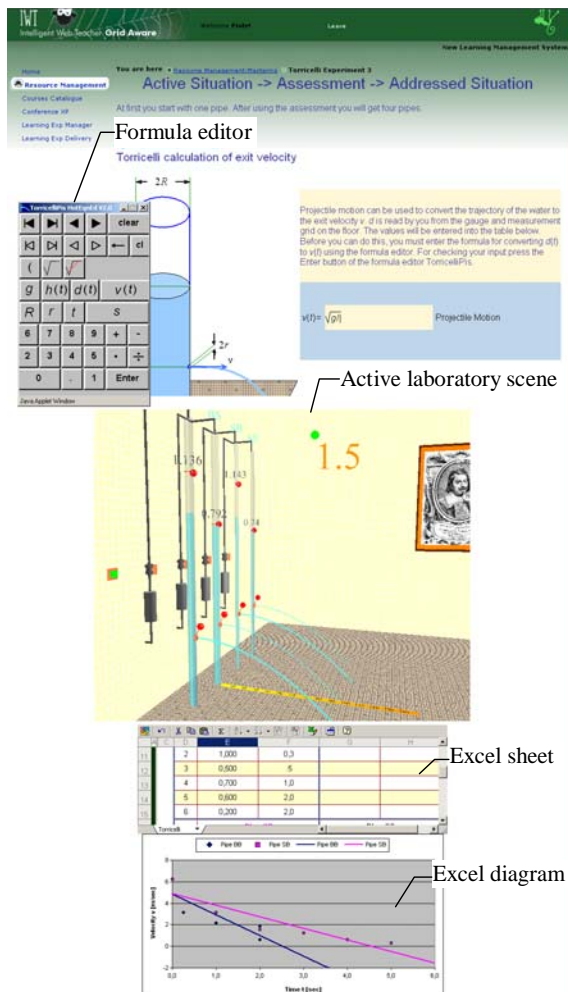
Fig. 7.   Screen shot section of a VCLab session.

The middle part contains the active 3D laboratory scene LO, where four pipes of different diameter can be filled by pumps with water to a given height. In the lower part of these pipes outlet valves of different diameter can be opened to let the water flow to the floor. The learner interacts with the experiment using this scene. Measurements are taken using a tape measure and a measurement grid on the floor and using a watch clock at the top of the scene. The scene is completely animated by the simulation service. The upper part deals with the relation between the trajectory of the water outflow and the exit velocity. The learner has to specify in this LO this relation in symbolic notation. The Formula editor is for entering the formula into the small window on the right-hand side. The symbolic user input is checked by the computation service and used below in the Excel sheet LO to covert the user measurement data into exit velocity data. The Excel diagram is for illustration purposes.

4.4   *VCLab grid resource*

The LOs realizing the macro-phases of the learning model in a UoL are generated using the VCLab Resource. Figure 8 presents the general structure of this very generic resource. It consists of a set of optional abstract elements whereby some of elements may oc-

cur in multiple instances. Customizing this resource in a proper way will yield the different LOs of the learning model. The appearance is also fully customizable by the sequence of elements in the resource as shown for the example in Figure 9.
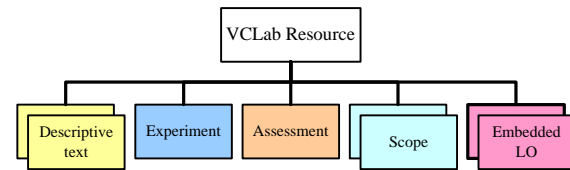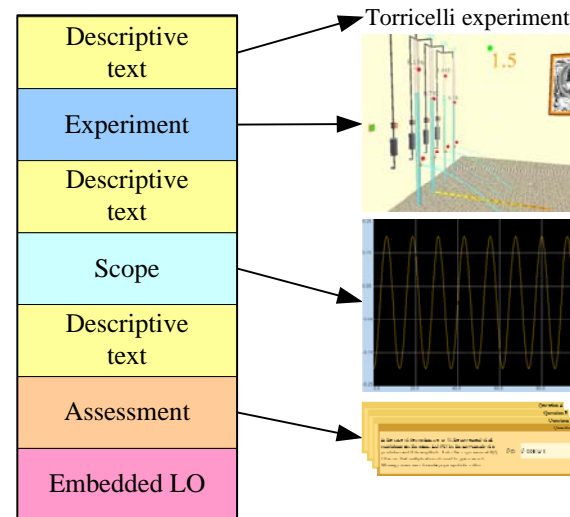

Fig. 8.   VCLab Resource structure.

Fig. 9.   Appearance of elements in a VCLab Resource.

The *Descriptive text* element is used to deliver information describing the elements that are placed above or below of it or may be used to present additional information. From the technical point of view this element is implemented as a HTML formatted text section. It may contain also active elements, e.g. Java applets, which are not standard VCLab Resource elements.

The *Experiment* element realizes the 3D virtual user environment. It animates the behaviour of the experimental plant in the 3D scene and provides the means of altering the experiment parameters through a set of buttons or sliders rendered as 3D objects. For this purpose a 3D player is used combined with invisible controls that animates the scene and intercepts events generated by the learner. The current version of the VCLab Resource provides a LO with one experiment at one time. But as later shown this resource can be cloned and used in a collaborative manner simultaneously by many users.

The *Scope* element provides the possibility to display signal history plots in a similar manner like using an oscilloscope device. Several instances of *Scope* elements may be present showing in parallel different signals using different display modes and scaling parameters.

The *Assessment* element defines a set of multiple/single choice questions or questions that needs to

be answered by entering the answer in symbolic notation, e.g. by mathematical expressions. Answers may be associated to events defining modifications of the experiment parameters.

Using the *Embedded LO* element the resource being created may also make use of already existing LOs to provide their functionality. Such an example may be LO for making notes of the measurements in an Excel sheet or a chat component for communication with other learners.

4.5 *Authoring VCLab UoLs*

Virtual laboratory models and its experiments require a large set of specifications for the related resources. Avoiding the error-prone and tedious work of authoring manually by using chains of several inhomogeneous tools an integrated authoring process of VCLab related grid resources has been developed starting from a manuscript or storyboard and supporting the process until to the final UoL.

Figure 10 presents a typical composition of a single LO, which is used to realize a Practical Situation macro-phase from Section 4.2. It contains metadata describing its content, the usability feature for the learner and the definition for the grid service to be used for its execution. It also contains the necessary data files for its execution and references to other LOs, which may be embedded within this LO.

All tasks to build a UoL are performed by using the VCLab authoring tools as shown in Figure 11. These tools are of generic type to describe the simulation model, the 3D visual representation, the assessment and to compose all resources to a UoL to be published on the ELeGI grid. In addition, these tools are itself grid resources published as grid services available for authors.

The creation of the Experiment resource is more laborious and happens in two stages. First, the experiment has to be described by a simulation model in using hierarchical block structures consisting of a set of differential and algebraic equations. The simulation parameters can be defined such that they can be manipulated by the learner. The simulation results are directed for animation or plotting in real time. This stage is performed by the *Simulation Model Authoring Tool*, which contains a graphical user interface for constructing simulation diagrams.
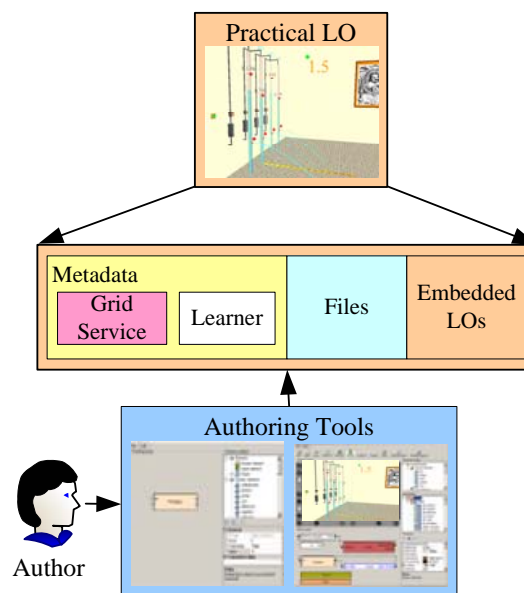

Fig. 10. A typical composition of a LO and its authoring.

Complex mathematical expressions may be added by embedding the HotEqn resource (HotEqn 2009), which interprets LaTeX expressions dynamically and renders them accordingly.

Second, the *Visual Objects Authoring Tool* is used to generate the 3D model for the visual representation of the experimental plant. Then the elements of the 3D model are bounded to the inputs and outputs of the simulation model. The creation of the Assessment resource is supported also by the *Assessment Editor*. This gives direct access to visual elements, which are controlled depending on the answers to provide information for the Addressed Situation.

The implementation of the Active Situation in the Abstract macro-phase is performed by the Scope element, which is integrated with the simulation model. Signal outputs generated by the simulation model are directed to the Scope element.

At the final stage the *Resource Editor* is used to mount all the resources together into one LO and make it available for the direct deployment into the grid environment.

4.6 *A process control example*

In the following a short example of a UoL about experimenting with a thermo-fluid process of a chemical plant is shown taken from (Juhásová and Schmid 2007), which exists in reality at Ruhr-Universität Bochum, Germany, see Figure 12.
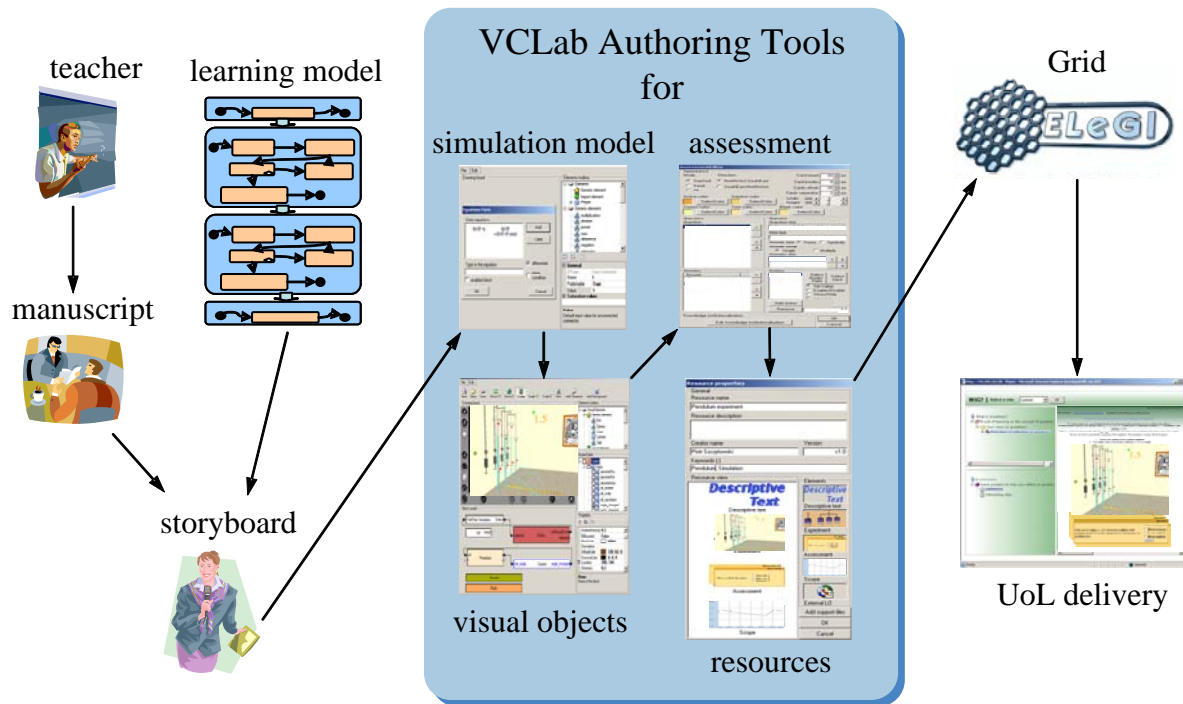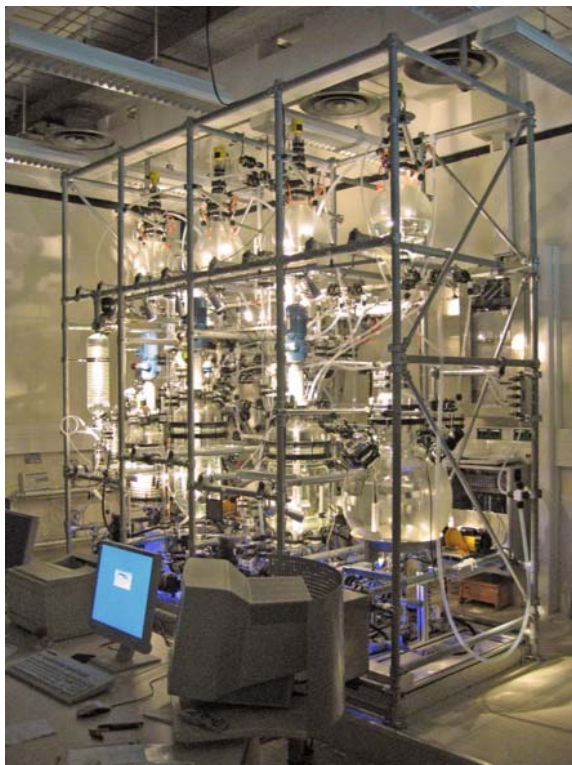
Fig. 11. From a virtual laboratory idea to a UoL.

Fig. 12. Chemical plant VERA at Ruhr-Universität Bochum, Germany.

A chemical plant is described by its construction model and physical model. The construction model defines parts of the model such as a pump or a valve and the physical model defines how they interact with each other and describes their dynamical behaviour by differential and algebraic equations. As shown in Figure 13 the thermo-fluid subsystem consists of four storage tanks T1, T2, T3 and T4, two main tanks TM and TB, one reaction tank TS, waste tank TW, heaters, two pumps and several valves all together connected by a pipe system.
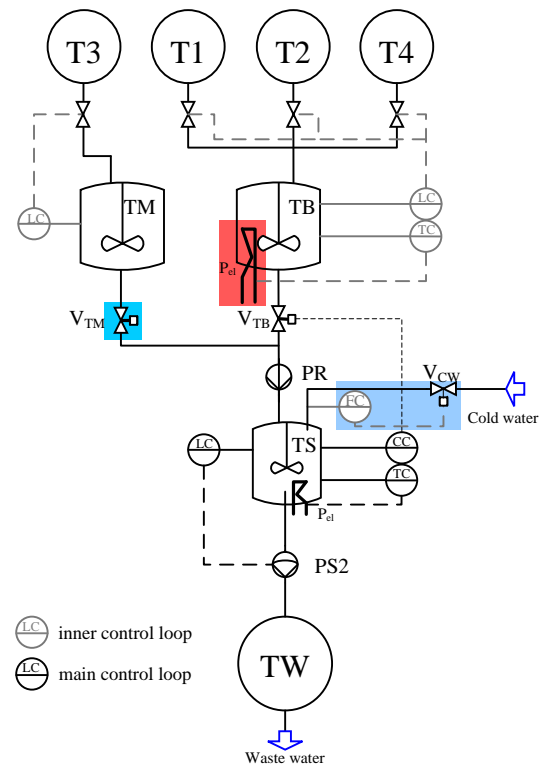

Fig. 13. Flow chart of the thermo-fluid subsystem.

The thermo-fluid subsystem including the control system and the physical constraints is described by a simulation model. It is built based on principles and laws for all the components as a fluid process (laminar, Torricelli flow, …) for tanks and pipes, valves and pumps. It is a dynamical system described by a system of differential and algebraic equations.

The VCLab authoring tools are used for specifying the entire simulation model, to generate the visual objects of the 3D model, to design the assessment and finally to compose the entire UoL. Figure 14 shows the screenshot of a learning session composed of several basic LOs. An active situation with a laboratory scene LO, several Scope LOs and an assessment LO with symbolic facilities are here used for exploring the dynamical behaviour of this process and to control it. The experiment flow is influenced by the assessment. Depending on the learner answers the scenarios can vary. An ontology used for building the UoL allows the easy navigation between the learning contents and assures their delivery in the right order according to the learning model from Figure 2.

4.7 Learner perspective of the grid implementation

From the learner perspective, the grid implementation simplifies the delivery process of the resources, in the sense that it removes the burden on the user of possessing and handling demanding software and hardware. The required hardware is always accessible; the higher complexity of the resource is compensated by the choice of a suited execution node, which is performed automatically by the grid infrastructure. The software for calculations and simulations is also always up to date because it is located on the server. Because of the server-sided execution the delivered content may be richer in details. Each LO of a UoL being executed may run on a different server. Therefore, the UoL can contain several LOs executing in parallel, what would not be possible in case of a conventional client-sided execution model. This property
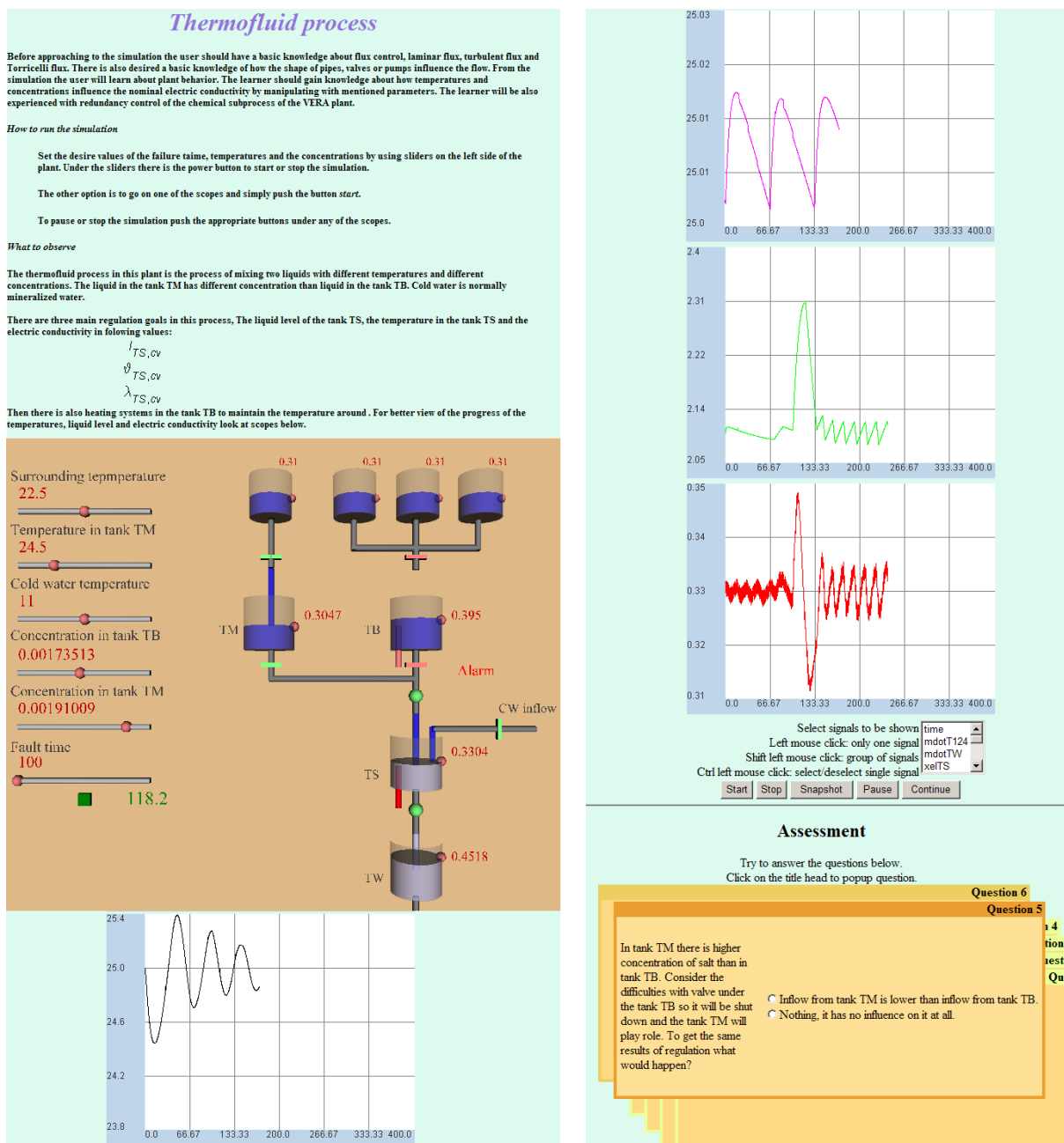

Fig. 14. Screenshot of an example session.

is especially important in case of Collaborative Learning described in detail in the following section.

It is especially important that the Learning Grid offers a personalized approach of learning in form of an activity. The learner profile carried together with the requests to the grid makes all experiences suited to the learners needs. The delivered content depends on the courses already taken and on the general advancement of the learner in a particular topic. Preferences, like language of content and its form: text, audio, video or their combination may be taken into account during delivery. As the progress made by the learner is hold in his profile, he may access the learning activities from any computer without caring for the synchronization of the context. The learner is not anymore bound to a particular workstation and his profile follows him.

Not only is the choice of the presented content affected, but also the overall quality of service. The right selection of the nodes for operation on an appropriate geographical location with respect to the learner results in faster response times. This is particularly important in case of animation, e.g. for real time experiments. Equipping the nodes with different kind of services allows a better user balancing and scalability. It is common that the enrolment to particular courses changes over time and even during semesters. The grid can adapt itself automatically to the demand for services.

5 COLLABORATIVE LEARNING IN GRID ENVIRONMENT

An important aspect of learning in virtual environments is Collaboration, which is a micro phase in the experiment learning model of Figure 6. It is actually a scenario in which learning activities may benefit mostly from the Learning Grid. It is hard to implement rich collaborative environments capable of running several simultaneous experiments and supporting several groups of learners working together in parallel using the classical server centred approach. Even fastest server would very soon reach its limit in case of a large amount of simple simulations or in case of executing several complex ones. Distributed services on the grid remove these bottlenecks. This is addressed in the following.

5.1 *Architectural perspective of grid supported Collaborative Learning*

Collaborative e-learning applications start the execution querying the Grid Catalogue for a driver service that can perform collaborative experimenting, see Figure 15. The catalogue provides this application with the node hosting the service, which is suited for it and which can sustain the amount of potential users and embedded experiments offered by the environ-

ment. The corresponding Driver Service takes over the execution. It contacts the Grid Catalogue for instantiating the other Driver Services for the embedded Simulation LOs. A good granulation of the solution can be observed, mimicking the property of a conventional grid of splitting the tasks in small slices and assigning them to nodes spread on a grid.
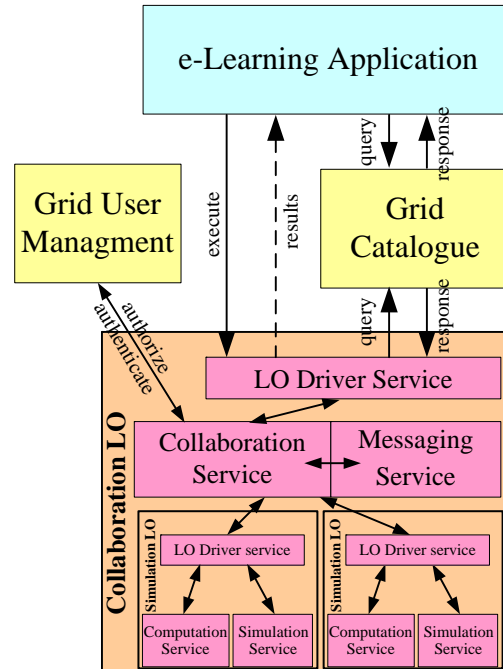

Fig. 15. Collaborative environment execution architecture.

For each instantiation of the Collaboration LO its supporting Simulation LOs may be delivered from different hosts depending on the current state of the grid and on the availability of the hosts. From the learners perspective there is no difference in learning experience what is assured by the quality of service constraints.

The Collaboration Service benefits from the authentication and authorization facilities of the grid infrastructure by the Grid User Management. Looking up the profile of the learner it may determine his role within the environment.

Collaborative experimenting using VCLab is based on a collaborative enhanced virtual laboratory scene as shown in Figure 16. The communication between the collaborating learners is based on two different types of information channels supported by the Collaboration Service. There is the 3D scene with the learner avatars, from where the participants can see and hear what the experimenting colleague is currently doing and to which gestures can be submitted as reactions. The other type of information channel is more classical. The Collaboration Service uses the Messaging Service. On default it is a textual chat, which may be replaced by binding another service to it providing audio or even video capabilities. This possibility is drawn from the architectural flexibility of the grid.
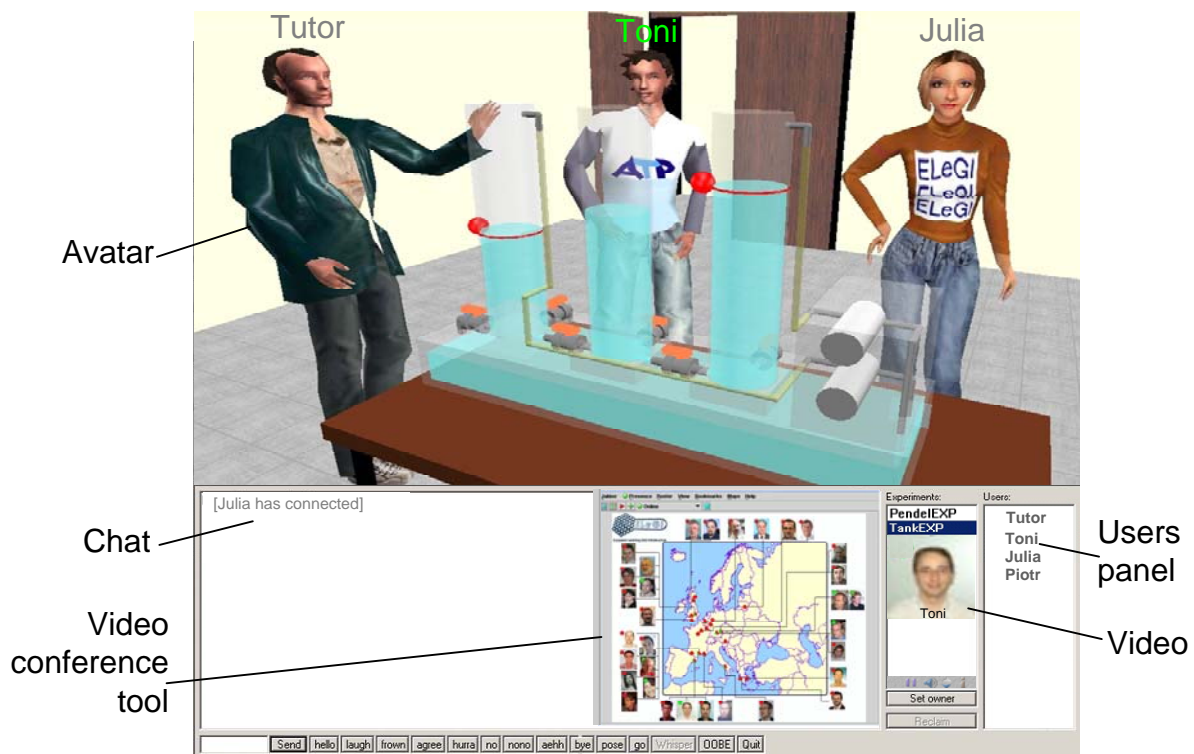
Fig. 16. Screenshot of a collaborative activity.

5.2 *The Learner perspective of grid supported Collaboration Learning*

The VCLab collaboration environment supports the full experiment learning model from Figure 6. It contains a 3D virtual scene including embedded experiments, see Figure 16. Participants of this virtual environment can be students of different universities participating simultaneously in the Learning Grid. They are represented by avatars and they communicate by means as shown above.

One important aspect of defining collaboration in e-learning is the definition of roles of participants and the assignment of these roles to them. The VCLab models four distinct roles within the collaboration environment, see Figure 17. These roles are Author, Tutor, Learner and Experimental Plant. The Author does not directly take part in the collaboration activities. Its tasks are, preparing the environment by means of defining experiments, tasks for learners, instructions, designing graphics and so on. The Tutor is a privileged participant of the collaboration activities. Its tasks are to provide the content to learners, to monitor their progress, to supervise experiments, to give hints, explanations and advice, to answer learners' questions, which may occur during the learning process. The Experimental Plant represents the modelled knowledge, which learners should gain during experimenting with it.

The learner role is described by a learner model. The learner is allowed to interact with the experiment in active (master learner) and in passive mode. An experimenting group may have a tutor and must have one student who is the master. The master is actively experimenting while the others are passive. The master's role can be passed to each of the participants to control the plant. Such a virtual meeting place could be the seed of creating a virtual learning community within a VO.
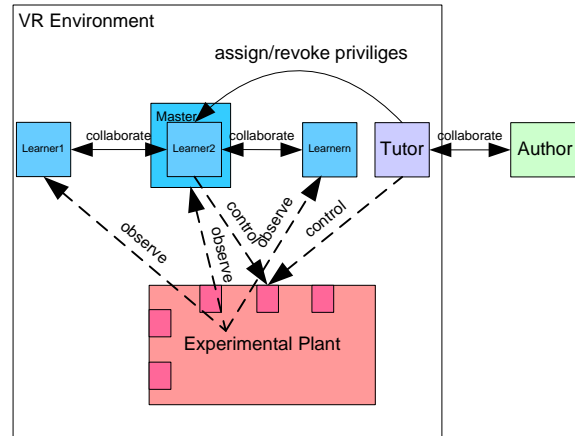

Fig. 17. Collaboration roles.

Another feature of this environment is that besides simultaneously running different experiments, a learner can dynamically plant its own experiment into the environment that he is working on. So it can be observed by fellow students and by the tutor. The Tutor can take over the control and make a demonstration, which changes will be reflected in the original learner experiment. After leaving the collaboration micro phase, the learner may continue its session regarding to the learning model. This example of dynamical binding is only possible due to the dynamic nature of the grid.

6 CONCLUSIONS

Learning Grids contribute to the achievements of the objectives given in the introductory chapter to this article through the definition of the learning services concept and their deployment through grid technologies. Learning services will be consumed in dynamic virtual communities based on communications and collaborations where learners, through direct experiences, create and share their knowledge in a contextualised and personalised way. This way of learning using grid resources can become now more open to learners in the engineering domain. From the 3D visual representation the learner can get the information about the plant more effectively than only from 2D scopes or only from a textual representation. The presented solution is complete and may be regarded as a proof of the concept for a Learning Grid.

The topic that is equally important but not discussed here is the authoring of the content of the grid supported Virtual Environments. VCLab provides a set of graphical authoring tools, which are self grid applications. They create 3D scenes, simulation models, the composition of them with supporting elements like Java applets, HTML text, and automating it with online assessments to produce addressed situations. The creation of collaborative environments is supported by composing previously created experiments with static elements of the environment and defining the roles of participants.

The application of grid technologies in education is of course a much wider topic than presented in this article and by the practical example of a virtual laboratory. Nonetheless the most important aspects of utilizing service-oriented grids in distance learning for engineering education are presented.

6   REFERENCES

Capuano N., A. Gaeta, G. Laria, F. Orciuoli and P. Ritrovato (2006). How to use GRID technology for building next generation learning environments. In Proc. of the 2nd International LeGEWG Workshop e-Learning and Grid Technologies: a fundamental challenge for Europe, Paris, P. Ritrovato, S. Salerno and M. Gaeta Eds., Electronic Workshops in Computing, British Computer Society.

ELeGI (2009). *European Learning Grid Infrastructure*. Available: http://www.elegi.org

Foster, I., C. Kesselman and S. Tuecke (2001). *The Anatomy of the Grid: Enabling Scalable Virtual Organizations.* International J. Supercomputer Applications, vol 15, no 3.

HotEqn (2009). *HotEqn – The IMGless Equation Viewer Applet.*

Available: http://vclab.atp.rub.de/software/HotEqn/HotEqn.html

IMS-LD (2009). *IMS Global Learning Consortium.* Available: http://www.imsglobal.org

Juhásová, E. and C. Schmid (2007). A learning unit for a thermofluid process. International Conference on Process Control PC'07, Štrbské Pleso.

OGF (2009). *The Open Grid Forum.* Available: http://www .gridforum.org

OGSA (2009). *The Open Grid Services Architecture, The Globus Alliance*, University of Chicago. Available: http://www.globus.org/ogsa

OASIS (2009). *Organization for the Advancement of Structured Information Standards.* Available: http://www .oasis-open.org

Schmid, C. (1998). The Virtual Control Lab VCLab for Education on the Web, in Proc. of the 17th American Control Conference ACC'98, Philadelphia, pp. 1314-1318.

Vallés, J., T. Dimitrakos, S. Wesner, B. Serhan and P. Ritrovato (2003). The Grid for e-collaboration and Virtual Organisations, in *Proc. of the eChallegnes e-2003 Conference*, European Commission under the Information Society Technologies (IST) Programme.

WSRF (2009). *The Web Service Resource Framework, The Globus Alliance.* University of Chicago. Available: http://www.globus.org/wsrf

WSRF.NET (2009). *The Web Service Resource Framework and Microsoft .NET technologies*, University of Virginia. Available: http://www.cs.virginia.edu/~gsw2c/wsrf.net.html