

Learning Grid Technologies in a Project-Based Computer Architecture Course

Guillermo Vega-Gorgojo, Yannis A. Dimitriadis, Eduardo Gómez-Sánchez, Miguel L. Bote-Lorenzo
School of Telecommunications Engineering, University of Valladolid
Camino Viejo del Cementerio s/n, 47011 Valladolid, Spain
{guiveg,yannis,edugom,migbot}@tel.uva.es

Alejandra Martínez-Monés
School of Computer Engineering, University of Valladolid
Camino Viejo del Cementerio s/n, 47011 Valladolid, Spain
amartine@infor.uva.es

Bartolomé Rubia-Avi, Iván Jorrín-Avellán
Faculty of Education, University of Valladolid
Paseo de Belén 1, 47011 Valladolid
{brubia,ivanjo}@doe.uva.es

Abstract

This paper describes the integration of the grid topic within a Computer Architecture engineering course. Students are engaged in a project of design and evaluation of computing systems for a specific customer/market sector. The contents of the course and the case studies have been adapted in order to motivate the study of the grid. In particular, two case studies have been analyzed: one concerns an Internet-based music distribution enterprise, while the other is related to the assembly of genome sequences. A preliminar analysis of this experience shows that most of the students chose to evaluate the suitability of the grid for their case study, and were able to argue on the benefits and drawbacks of this and other alternatives. In the future, we plan to set up a grid-based learning environment so the grid is not only the object of study, but also a tool for learning.

1. Introduction

The computational grid, a large-scale geographically distributed infrastructure that provides resource sharing for dynamic multi-institutional virtual organizations [3], is achieving increasing interest both from the scientific community and the industry. As a result of a certain degree of technological maturity, standards such as the Open Grid Services Architecture [10] have been proposed attempting

to engage enterprises to develop products related to the grid. Indeed, vendors such as IBM [16], Sun Microsystems [27] or Oracle [22] are investing seriously in the topic.

However, since grid is an emerging technology it demands great research efforts to be performed by a qualified workforce, and this need contrasts to the lack of courses where the topic is studied at universities. For example, the IEEE/ACM Computing Curricula 2001 [1] does not consider any course for learning the grid in Computer Science disciplines, though related issues such as middleware or cluster computing appear as advanced topics. Besides, new courses are not always easy to launch at universities, especially if the body of knowledge to be studied must comply to national laws, as it occurs in Spain.

Therefore, grid computing could be introduced as a new topic in some existing courses. For example, in a course on Distributed Systems the grid can be presented as a new kind of middleware that enables the use of resources from third-party organizations, and related to the peer-to-peer approach [23]. Furthermore, the grid can also be introduced in a Computer Architecture course, together with other parallel computing alternatives such as shared-memory multiprocessors, massively parallel multiprocessors and clusters.

However, in order to achieve effective learning students should not be limited to a presentation of the different features of this new technology, but they should rather develop a critical view of computing strategies, finding out benefits and drawbacks on their own. This would help the genera-

tion of a creative workforce that may spread the use of the grid. To meet these objectives it is convenient to follow IEEE/ACM Computing Curricula recommendations. The 1991 edition [28] emphasizes the use of recurrent concepts, and proposes the use of system design/evaluation projects as well as case studies. The 2001 edition [1] preserves this idea, while remarking that graduate students need to develop some abilities like abstraction, efficient teamwork or adaptability.

Therefore, project-based learning and case study seem adequate approaches so that students achieve a mature, critical knowledge of the grid technology, while developing important skills. These pedagogical strategies engage students in solving a real problem. Thus, they must understand it and propose a solution by discussing and evaluating possible alternatives. In this process, students practice abilities such as analysis, knowledge application and decision-making, while their motivation is strongly increased compared to traditional learning scenarios. Moreover, it has been seen that students reach a deeper knowledge if they are put to collaborate in groups [17]. The main benefit of this approach is that the teaching/learning process is focused on the student, rather than on the teachers, and therefore students become active in constructing their own knowledge. This pedagogical model stems from the theories of social constructivism [29] and implies a strong change in curriculum design and practice in all educational levels.

In this paper, we describe the integration of the grid topic in a course on Computer Architecture at university level. The course is project-based, with different case studies, involving the design and evaluation of computational systems for different customers in real market sectors, some of them requiring supercomputation capabilities. In [6] a description of the objectives, methodology and design of the proposed educational project is provided, while [20] describes how collaborative abilities and attitudes were shown to improve learning. This paper will address how the grid topic is introduced in the course, and discuss the reactions of the students to their customer problem, concerning the use of a grid.

The rest of the document is organized as follows: in section 2 we provide a description of the proposed educational project in Computer Architecture. In section 3 we describe the actions adopted in the integration of the grid and we discuss the experience gained along current academic year. Finally, we draw the main conclusions of the study and show the current research work.

2. Educational organization of the Computer Architecture course

This section describes the educational organization, as well as the pedagogical strategies that we use in the Com-

puter Architecture course in our University. This course is part of the core body of knowledge in the Telecommunications Engineering curriculum, in Spanish universities. The course takes place in the fall semester of the fourth year (out of five), concluding a branch on computing topics, that includes programming fundamentals and operating systems. Therefore, students expect to face somewhat real-world problems, in order to apply all this knowledge. Besides, the course comprises 30 hours for theory explanation and 60 hours for practice, that can be spent in the laboratory. These facts make this course suitable for project-based and case-study learning, as reported in the literature by several research studies for similar learning scenarios [7]. Indeed, design and evaluation of computer systems is a task in which different opinions and solutions can be valid, teamwork is necessary to efficiently perform such a complex task, and technical and commercial information as well as simulators are available in the public domain. Significantly, a very popular textbook on Computer Architecture [14] promotes a quantitative approach to the study of the field, which is especially suitable in this context.

Therefore, we consider the following general learning objectives for the course:

- With respect to contents, students must know and understand the common organization of a computer, the different types of processor architectures, the relationship between different subsystems, and the techniques applied to improve performance in each one of the subsystems. In summary, we care for many of the learning objectives proposed by the IEEE/ACM Computing Curricula 2001 [1] in units AR3 through AR9 (some of the remaining topics are covered in other elective courses). Besides, students are offered a brief review of different approaches in operating systems.
- In addition to the exposure to these topics, students must construct their own knowledge and apply it to propose innovative solutions to real problems, without an explicit application roadmap prescribed by the teacher.
- Moreover, there is a number of general skills that students are expected to develop by their graduation. Therefore, though not a specific objective of the course, we designed activities that also promote some skills, such as information selection, collaboration, information sharing, discussing and technical report writing.

With these objectives in mind, the course is based on a design and evaluation project of computing systems structured in subprojects. The 13 week-long semester corresponds to three subprojects of about four weeks each. During each subproject different subsystems and techniques

such as analytic models, real machine benchmarking or simulations, are studied.

Students are organized in pairs and take different roles within the project. First, they play consultants of a consultancy firm that must study some market sector as well as the existing computing technology, in order to assist a customer to purchase a computing system for his business problem. Besides, students play engineers of a computer manufacturer which must design and evaluate the different subsystems. Teachers play the roles of the customer, the director of the consultants' team, or the CEO (Chief Executive Officer) of the manufacturing company. Instead of proposing only one customer for all pairs, five different case studies are considered each year, but each pair of students deals only with one given customer. At the end of each subproject every group has to produce a formal technical report. The last report is collaboratively produced among all groups that deal with the same case study in a session (about three or four groups), though they are not limited to collaborate with students in the same session (the same time slot to attend the laboratory), but they can also share information and discuss with groups in other sessions through asynchronous computer supported collaborative tools. Therefore, the structure of the class is as shown in Figure 1, where the different levels of collaboration are remarked.

The fact that different pairs of students may deal with different customers enriches the learning process by the need of contrasting customer requirements and proposed solutions. In fact, they tend to develop critical positions towards different technologies, instead of absolute preferences for a particular one. To meet these objectives, the proposed case studies correspond to different businesses that require different solutions, such as the assembly of genome sequences (supercomputing), an Internet-based music distribution company (multiprocessing high-end servers) or a museum tour (wireless devices). Besides, each year case studies are renewed introducing hot topics in technology and business worlds.

The project starts with the customer formulating the requirements on the computing system for his business (in fact for the market sector in general), but in terms of the business domain. Therefore, the consultancy firm must generate a ranked list of technical requirements, and model the estimated workload for them. To do so, the consultants can choose between using existing benchmarks (such as Spec [26]) or designing new ones. Next, they analyze the performance of some existing machines (those in the laboratory), and consider their cost as well as possible performance loss by inadequate integration of the different subsystems (CPU, memory hierarchy, I/O, operating systems, software applications) for this particular business. In this sense, they could make a recommendation, but based only on the existing machines. Nevertheless, the engineers of the computer manu-

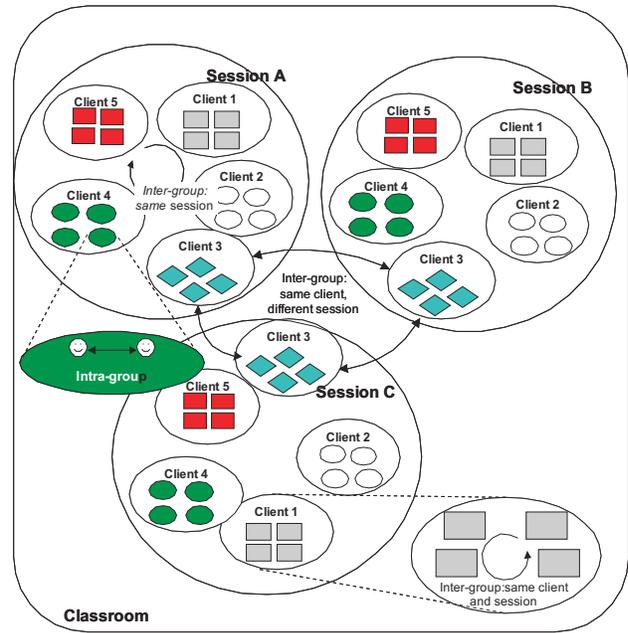


Figure 1. Structure of the class showing the levels of expected collaboration: within a group, two students collaborate; generally, all pairs with the same customer (client) in the same session discuss face-to-face issues concerning their customer; they also discuss with students having other customers about their differences or about technology; finally, students in separate sessions collaborate asynchronously through computer supported tools.

facturing company can propose and evaluate new designs based on the educational computer architecture DLX [13], meeting the demands of the consultancy firm. Finally, the consultants must propose and evaluate a global solution for the customer. The solution can range from just a scalar single-processor machine to any type of distributed system such as a cluster or a grid. Appropriate marketing and financial strategies are also prescribed both for the producer and for the customer.

In order to improve the pedagogical strategies, this learning process is systematically evaluated. A detailed description of this evaluation is covered in [20]. Briefly summarized, the information used in the evaluation is originated from heterogeneous sources: data logs from the cooperative computational tool BSCW [12], technical reports and regular student questionnaires evaluating the educational project, systematic ethnographic observations collected in the diary of an external observer, focus group sessions held with a group of volunteers, as well as grades and observations made by the teachers.

3. Learning the grid in this educational setting

In the previous section, the educational project of a Computer Architecture course was described. As already mentioned, the increasing research and industrial investments on the grid make this topic suitable to be integrated in this course. Since the course will not be utterly devoted to the grid, the focus will be set on the design and evaluation of computing systems for high level parallelism workloads. Hence, grid middleware details will not be explored thoroughly. In summary, we consider the following learning objectives concerning the grid:

- Learning the foundations of grid computing, distinguishing it from other parallel systems like shared-memory multiprocessor machines or clusters, as well as from other resource-sharing approaches as peer-to-peer networks.
- Determining its suitability for a number of real-world problems, in a performance-cost comparative to other alternatives.

To accomplish these objectives, in the current year a two-fold effort has been made. First, the grid topic is covered in theory lectures, with the restrictions mentioned above. Second, the teachers select some of the case studies from real-world problems for which a grid would be a reasonable solution, although it could be argued that other approaches like clusters could also be valid. This way, students must confront the alternatives and learn the differences among several technologies, getting a critical insight into their benefits and limitations.

As for the grid foundations provided in theory lectures, the topics considered are as follows:

- A definition of the grid [4] is provided, relating it to other distributed, resource-sharing systems, and pointing out the differences [11].
- The types of applications supported by the grid are discussed, as by [8]: distributed supercomputing, high-throughput computing, on-demand computing, data-intensive computing and collaborative computing. This relates the grid concept to many of the case studies carried out in the course practice, and therefore puts the grid as one of the alternatives to be considered by the students.
- The Open Grid Services Architecture [10] standard is just mentioned, recalling what students know about web services from a previous course on middleware for distributed systems.
- Finally, the challenges for the grid technology are pointed out, following [3], and relating them to other

courses in the curricula. For example, scheduling is related to operating systems, toolkits to software engineering, messaging to distributed systems...

Some basic bibliography is provided, like introductory papers [8, 2] and well-known books [9, 3]. Besides, selected press clipping complements this material offering a non-academic vision of this topic.

While theory lectures provide the baseline to understand the grid, a deeper insight can be gained through the work with case studies where the grid is a suitable solution. As previously stated, students are organized in pairs and must provide a computing solution to a customer, but not all student pairs serve the same customer. Rather, five different customers promote the exploration of different market sectors as well as different computing technologies. Therefore, it is expected that discussions arise in two ways, enriching the learning process: on one hand, students with the same customer but proposing different solutions must argue on their benefits and drawbacks, becoming critical on technology; on the other hand, students with different customers discuss to learn how the benefits of one technology apply to one given business, but not in general. In this sense, neither it is convenient that the grid (or any other particular technology) suits all case studies, nor that it is an obvious solution to any of them. Rather, several of the case-studies *may* require a grid, but other solutions can be arguably defended as well. In summary, all these discussions help students have a broader view of technologies, including the grid, but related to their own experience and that of their classmates.

As educators, to state that this educational approach is effective and efficient, we have been carrying a systematic evaluation [20] for several years. However, since the grid topic has been introduced this semester (fall 2003), we cannot provide here a complete evaluation. Nevertheless, we have employed data currently available to draw a preliminary analysis of the accomplishment of the learning objectives. By the time of writing this paper, the second subproject had just finished, so we have used intermediate reports from the first two subprojects and questionnaires proposed to the students. In this paper we will illustrate the results obtained from these data sources, using two of the case studies proposed this semester.

3.1. Case study 1: Internet-based music distribution business

A brief description of the first case study is as follows: The fictitious company MUSICA is a new business venture which aims at establishing a new model in music distribution. Indeed, traditional means of distribution are being challenged: first, CD sales are dropping due to illegal copies; besides, peer-to-peer applications [23] provide an

easy way to share music files between users without control by music enterprises. MUSICA has been recently established with the aim to provide an Internet-based music distribution business in the Spanish market. They plan to reach the European market in the near future. The increasing number of homes with Internet access and the improvements in computer networks make the new model of music distribution intended by MUSICA suitable. Its subscribers will access two services: the first allows downloading MP3 files, while the second is an audio streaming service [19]. MUSICA has signed agreements with several record companies to distribute their music.

This case study models a high-scale Internet business. It is not a supercomputation problem, which is the traditional domain of grids. Instead, MUSICA must serve many independent light-weight tasks. This task-level parallelism may translate to a parallel architecture, but not directly to a grid. Other relevant properties are the large amount of data that must be stored, calling for data-scalability to support the storage of new songs. Besides, for both downloading and for streaming services, data location and network bandwidth are critical to provide an acceptable quality of service (QoS). This issue and the fact that users will be geographically distributed suggest the suitability of a data grid such as in [18], with replication and migration capabilities to manage data storage.

These are some of the expected arguments that may appear in this case study that involve the grid, and help to know its relevant features. Significantly, nine out of the 12 student pairs serving MUSICA have considered and evaluated the suitability of a grid for this problem, four of them finding it the most adequate solution.

Moreover, whether students favor the use of a grid or not, they point out important features of the grid. Students that discard its use have different arguments. One group argues that security is a critical requirement for MUSICA and they are concerned with such issue in the grid. Another group discards the grid because delegation on third-party organizations is prone to error. Costs associated to middleware development are pointed out by two other groups. Finally, one group indicates that high-bandwidth requirement negatively affects the acceptance of a grid.

However, groups supporting the grid as a solution also have their arguments. The need for data scalability and the geographical distribution of users are the main issues in favor of the grid. Four groups also pointed out that the expected growth in the number of users, and the planned expansion to the European market make the grid an scalable, easy to adapt solution.

Interestingly, one group expressed the following assessment: "The whole system need not be grid-enabled, but while the file service could be supported by a data grid, other independent systems may perform tasks such as user

logging and payment management. Since the file service is the most time-consuming, response time could be greatly reduced if music files were distributed and replicated across multiple sites. In addition, migration would allow dynamic adaptation to the regional tastes of subscribers, if any". This argumentation neatly fits the expected benefits of a grid applied to this customer.

3.2. Case study 2: Genome sequence assembly

The second case study deals with a bioinformatics laboratory. The following is a brief description: The fictitious laboratory called GENOMA is devoted to the assembly of genome sequences [24]. It receives assignments from other research laboratories to assemble fragments that comprise a genome sequence. The typical length for a genome sequence ranges from millions to billions of base pairs. The available fragments, called inserts, are from two to ten thousands of base pairs long. The sequence assembler (a computing program) pieces together the overlapping inserts and reconstructs the original sequence. In a final stage, the sequence is validated by biology experts, with the aid of scaffolding computational tools. Scientific collaboration is required to obtain key genomic data, such as Sanger databases [25], and to promote knowledge sharing.

This case study covers a computing-intensive problem which is the main application field for the grid. A computing program must deal with huge amounts of data to generate a very long sequence. Sequencing algorithms, such as that described in [24], can be parallelized in many loosely coupled tasks. A grid system could be used to harvest the computing power of available resources. In this case, scheduling grid resources is a critical issue [21]. A grid federation of scientific organizations would enhance collaboration and ease the management of the growing amount of genomic data [15].

This is the discussion we expected with respect to the grid for this case study. Six out of the nine groups engaged with GENOMA have taken the grid into account and, curiously, only two think of it as the most suitable solution.

We were concerned with the low acceptance of the grid for such a customer, which apparently called for the grid so clearly. Qualitative analysis showed that a misleading interpretation of the grid had spread out, as one group expressed: "Grids are large-scale geographically distributed systems so they are non adequate for an autonomous laboratory like GENOMA". Most of the groups discarded the grid due to a similar argumentation. It seems that we did not emphasize the requirement of scientific collaboration, while students somehow misunderstood the initial specifications, getting to assume that GENOMA should be totally autonomous.

Nevertheless, two groups highlighted the distributed su-

percomputing nature of the problem and related it to the grid. They suggested it to harness the computational power of the resources available. Interestingly, one group proposed the constitution of a grid by federating several laboratories to maintain a critical pool of resources.

With respect to the choice between grids and clusters, students pointed out the wide area nature of grids in contrast to clusters, as well as the easiness for the use of heterogeneous resources in the grid. Finally, most of the groups focused on network requirements and criticized grid dependence on high-bandwidth wide area networks in the context of GENOMA.

3.3. Discussion

Following a project-based approach, the grid topic has been introduced in a Computer Architecture course. We must challenge ourselves questioning if the proposed educational project achieves the intended objectives concerning this topic, and how can we improve this educational setting. However, we should mention that these results are obtained using the documentation from the first and second subprojects, and therefore are not definitive and need to be completed after the last subproject is over.

The analysis performed shows that most of the students have considered this technology and argued on its suitability for a real problem, as Table 1 illustrates for MUSICA and GENOMA. It was an explicit objective to put the grid in a real context, so this is a major achievement. However, only a small set of groups proposes a grid-based solution. This fact was expected for customer MUSICA, as it was discussed in section 3.1. Anyway, students have pointed out interesting grid benefits for this case study such as large scale or increased QoS, relating technology properties to the particular aspects of a real problem. The main alternative to the grid is a cluster, as it is supported by seven out of twelve groups. Arguments in favor of the cluster are cost-effectiveness, scalability and suitability for high-level parallelism loads like that of MUSICA. In fact, three of the groups proposing a grid also remarked the suitability of a particular grid in which some of the nodes were clusters, in order to serve requests properly.

The most disappointing result is the low rate of acceptance of the grid for GENOMA, as only two groups propose a grid system. Indeed, groups assigned to this customer have shown less interest on the grid than those corresponding to MUSICA. Although GENOMA was designed to match a grid, we seem to have transmitted the misleading requirement of GENOMA autonomy, which refrained students from proposing a grid. Nevertheless, both groups favoring or against the grid were able to support their election, showing certain knowledge of the topic. In this case, there is not a clear option. Due to the computing-intensive

Table 1. Measured levels of grid acceptance and proposed systems for MUSICA and GENOMA case studies. MPP stands for Massively Parallel Processor, SMP for Symmetric MultiProcessor and DSM for Distributed Shared-memory Multiprocessor.

Action	MUSICA	GENOMA
Considering a grid	9/12 (75.0%)	6/9 (66.7%)
Proposing a grid	4/12 (33.3%)	2/9 (22.2%)
Proposing a MPP	0/12 (0.0%)	1/9 (11.1%)
Proposing a SMP	1/12 (8.3%)	2/9 (22.2%)
Proposing a DSM	0/12 (0.0%)	1/9 (11.1%)
Proposing a cluster	7/12 (58.3%)	3/9 (33.3%)

nature of this problem, students have chosen some form of parallel machine ranging from a symmetric multiprocessor to a massively parallel processor. The arguments provided were well-proven technology for symmetric multiprocessors, high number of parallel nodes for distributed shared-memory multiprocessors and cost-effectiveness for clusters, although cost is not a critical aspect for GENOMA. Finally, we withdraw some experience from practice that will help us to design case studies more carefully for the next semester.

Concerning the effectiveness of project-based learning, this issue has been exhaustively evaluated in the general context of the course. We expect that the benefits reported in [20] will also apply to the learning of the grid. We have observed a considerable interest in this technology, supposedly due to the relevance of the grid in the media, as well as the challenging problems grid are applied to. Besides, students learn by collaborating and actively deepen their knowledge as they feel the need to know more. This way, some of the student arguments reported in subsections 3.1 and 3.2 show that they draw non-evident conclusions and take well founded decisions. Most of the students have made an effort to put the grid in the context of their customer and arrived to some interesting discussions, as observed by the teachers. However, it seems that some groups have strictly limited to map the requirements of the customer onto the general properties of the grid. Indeed, we miss a broader discussion among computing alternatives and a deeper analysis in some cases. As the course has not yet finished, we expect an improvement in this final stage.

In addition, one of the problems we are facing is that grid technologies embrace a very wide horizon and the focus of the Computer Architecture course is somewhat restricted. Nevertheless, the grid could be gradually integrated in other courses with related topics. This way, in an Operating Systems course the problem of resource scheduling could be

extended to resource scheduling in a meta-system like the grid. Similarly, OGSA grid services could be presented as a new technology of middleware in a Distributed Systems course.

Though this paper has dealt with learning the grid as the *object* of study, we are also addressing the use of the grid as a *tool* for learning. In this sense, we are currently working on the development of a Grid Collaborative Learning Environment (GRIDCOLE) based on the standard OGSA. We intend to produce an application which integrates the tools used in the course (benchmarking, simulation, document sharing) and new ones (collaborative edition, discussion), that shall be offered as OGSA-compliant grid services by third-party providers. This way, GRIDCOLE will help teachers in the definition of the learning activities and will enable the use of external resources. Therefore, the course can be enhanced with new features such as the incorporation of third-party simulators or real benchmarking of computing systems not available in the laboratory, or even benchmarking of the actual grid. This way, students would experience the use of an application run over a grid infrastructure, feeding back the cognitive loop: they would study the appropriateness of a grid for a real problem using computational resources forming part of a grid. This work is reported in [5].

4. Conclusions

The introduction of a course devoted to grid computing is not always possible, due to difficulties in the design of a curricula that must comply to general laws. Hence, the integration of the topic within existing courses remains as an alternative. However, this is not an easy task, since it must be put in the context of the rest of the course contents. Concerning the grid issue, this paper has presented the effort made to integrate the topic in a Computer Architecture course for engineering students at university level. The course is conceived as a design and evaluation project of computing systems, and therefore the grid had been presented as one of the available alternatives, that must be understood to evaluate its suitability for the case study.

Since the course is organized in a project-based manner, students must provide a solution for the customer in their case study. To motivate the selection of the grid, the case studies are proposed so that the grid could be a reasonable solution to some of them, though other solutions could be accepted as well if supported with valid arguments. To provide students with theoretical foundations, some small changes have also been made in the lectures of the course. With the whole educational setting, we expect students to become active actors of the learning process and acquire a deeper knowledge than that usually achieved with a traditional approach.

The general benefits of this educational project have been systematically evaluated during various years, and reported in [6][20]. Concerning the grid, this technology is appropriate for the proposed case studies MUSICA and GENOMA. A preliminar analysis showed that most of the students had discussed the suitability of a grid for these case studies and compared benefits of this solution with other computing systems like clusters or shared-memory multi-processors. Some of them drew very interesting conclusions and proposed different configurations to adapt the grid to their case studies. In addition, teachers observed students showed a vivid interest in the topic. Thus, we may expect they spread out the use of the grid in the near future.

Another interesting finding was the critical importance of a careful design of case studies. A requirement that was badly expressed caused many students to discard the grid for a problem in which it was very suitable, limiting a further study of this technology. By the time of the conference the semester in which the course takes place will be over, and we shall provide results of a more thorough evaluation.

Acknowledgements

This work has been partially funded by Spanish Ministry of Science and Technology projects TIC2002-04258-C03-02, TIC2000-1054 and Castilla y León Regional Government project VA 117/01. The authors would also like to thank the multidisciplinary group EMIC at the University of Valladolid, and, very specially, all students that avidly participated in this academic program.

References

- [1] ACM and IEEE Computer Society. "Computing curricula", <http://www.computer.org/education/cc2001>, 2001.
- [2] M. Baker, R. Buyya, and D. Laforenza. "The grid: international efforts in global computing". In *Proceedings of the Computer and eBusiness Conference*, I'Aquila, Italy, 2000.
- [3] F. Berman, G. Fox, and A. Hey, editors. *Grid computing: making the global infrastructure a reality*. John Wiley and Sons, Chichester, UK, 1st edition, 2003.
- [4] M. Bote-Lorenzo, Y. Dimitriadis, and E. Gómez-Sánchez. "Grid characteristics and uses: a grid definition". In *Proceedings of the 1st European Across Grids Conference*, Santiago de Compostela, Spain, 2003. Springer-Verlag.
- [5] M. Bote-Lorenzo, L. Vaquero-González, G. Vega-Gorgojo, J. Asensio-Pérez, E. Gómez-Sánchez, and Y. Dimitriadis. "GRIDCOLE: A grid collaborative learning environment". In *Proceedings of the workshop on Collaborative Learning Applications of Grid Technology at CCGRID 2004*, Chicago, IL, USA, Dec. 2003.
- [6] Y. Dimitriadis, A. Martínez-Monés, and B. Rubia-Avi. "Co-operative learning in computer architecture: an educational project and its network support". In *31th ASEE/IEEE Frontiers in Education Conference*, Reno, NV, USA, Oct. 2001.

- [7] E. Gehringer, editor. Special issue on computer architecture education. *IEEE Computer Society Technical Committee on Computer Architecture newsletter*, Sept. 2002.
- [8] I. Foster and C. Kesselman. *Computational Grids*, chapter 2, pages 15–51. In Foster et al. [9], 1st edition, 1999.
- [9] I. Foster and C. Kesselman, editors. *The Grid: Blueprint for a new computing infrastructure*. Morgan Kaufmann, San Francisco, CA, USA, 1st edition, 1999.
- [10] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. *The physiology of the Grid*, chapter 8, pages 217–249. In Berman et al. [3], 1st edition, 2003.
- [11] I. Foster, C. Kesselman, and S. Tuecke. *The anatomy of the Grid*, chapter 6, pages 171–197. In Berman et al. [3], 1st edition, 2003.
- [12] GMD-FIT. “Basic support for cooperative work, v. 4.0”, <http://bscw.gmd.de>, 2003.
- [13] J. Hennessy and D. Patterson. *Computer Architecture: A quantitative approach*. Morgan-Kaufmann, San Francisco, CA, USA, 2nd edition, Nov. 1995.
- [14] J. Hennessy and D. Patterson. *Computer Architecture: A quantitative approach*. Morgan-Kaufmann, San Francisco, CA, USA, 3rd edition, 2002.
- [15] T. Hey and A. Trefethen. *The data deluge: an e-Science perspective*, chapter 36, pages 809–824. In Berman et al. [3], 1st edition, 2003.
- [16] IBM. “IBM Web page - Grid Computing”, <http://www-136.ibm.com/developerworks/grid>, 2003.
- [17] D. Jonassen, K. Pek, and B. Wilson. *Learning with technology. A constructivist approach*. Prentice Hall, Upper Saddle River, NJ, USA, 1999.
- [18] P. Kunszt and L. Guy. *The Open Grid Services Architecture, and data Grids*, chapter 15, pages 385–407. In Berman et al. [3], 1st edition, 2003.
- [19] J. Kurose and K. Ross. *Computer networking. A top-down approach featuring the Internet*. Addison Wesley, Boston, MA, USA, 2nd edition, 2003.
- [20] A. Martínez-Monés, Y. Dimitriadis, B. Rubia-Avi, E. Gómez-Sánchez, and P. de la Fuente-Redondo. “Combining qualitative evaluation and social network analysis for the study of classroom social interactions”. *Computers and Education*, pages 353–368, Dec. 2003.
- [21] J. Nabrzyski, J. Schopf, and J. Weglarz, editors. *Grid resource management: State of the art and future trends*. Kluwer Academic Publishers, Boston, MA, USA, 1st edition, Sept. 2003.
- [22] Oracle. “Oracle Web page - Grid Computing Technology Center”. <http://otn.oracle.com/tech/grid/index.html>, 2003.
- [23] A. Oram, editor. *Peer-to-Peer: Harnessing the power of disruptive technologies*. O’Reilly, Sebastopol, CA, USA, 1st edition, 2001.
- [24] M. Pop, S. Salzberg, and M. Shumway. “Genome sequence assembly: Algorithms and issues”. *IEEE Computer*, pages 47–54, June 2002.
- [25] Sanger Institute. “Sanger Institute Web page”. <http://www.sanger.ac.uk>, 2003.
- [26] SPEC. “SPEC Web page”. <http://www.specbench.org/spec>, 2003.
- [27] Sun Microsystems. “Sun Microsystems Web page - Infrastructure Solutions: Grid Computing”. <http://www.sun.com/solutions/infrastructure/grid/>, 2003.
- [28] J. Turner. “A summary of the ACM/IEEE-CS joint curriculum task force report: Computing curricula 1991”. *Communications of the ACM*, pages 69–84, June 1991.
- [29] L. Vygotski. *Mind in society*. Harvard University Press, Cambridge, MA, USA, 1978.