Design of a Distributed and Asynchronous System for Remote Evaluation of Students' Submissions in Competitive E-learning

L.M. Regueras[1]*, E. Verdú*, J.P. de Castro*, M.A. Pérez* and M.J. Verdú*
* University of Valladolid, Valladolid, Spain

Index Terms: Remote Evaluation, Competitive Learning, E-learning, Moodle.

Abstract: This paper presents the design of a distributed asynchronous system to allow remote evaluation of students' submissions. The system is being developed in the context of the *Edujudge* project. The main goal of the *Edujudge* project is to provide the *UVA On-line Judge*, which is an on-line programming trainer that counts with an automatic evaluation engine, with a greater pedagogic character. The aim is to adapt the *UVA On-line Judge* for its use in formal educational environments, such as the official courses regularly offered by different types of educational institutions including Universities, secondary schools or Vocational Education and Training (VET) centers. More specifically the *Edujudge* project pursues the integration of the *UVA On-line Judge* into the open source e-learning platform *Moodle* and into the *QUESTOURnament* module. *QUESTOURnament* is a competitive e-learning tool that allows the development of contests within the classroom, or on a remote basis, and that has been implemented as a module integrated into the Learning Management System (LMS) *Moodle*. Both systems, the *UVA On-line Judge* and *QUESTOURnament*, have currently an important number of users which provides the design and development of the distributed asynchronous system proposed, and the whole *Edujudge* project, with an important added value.

1. Introduction

During the last years, programming competitions have risen in terms of number of contests, participants, size and scope. Relevant examples of programming competitions are the *ACM International Collegiate Programming Contest (ICPC)*[2], the *International Olympiad in Informatics (IOI)*[3], *Imagine Cup*[4], *Topcoder*[5] or *Google CODE JAM*[6], which have more participants every year. Programming contests and on-line judges are very often used, not only for the competition itself, but for the previous phase of training with a view to face the competition.

Programming competitions have traditionally been used by teachers and students to increase learners' programming skills. However they are currently being used by other stakeholders of the computing sector, different than the education institutions, as for example, companies of the sector that are making use of programming competitions as a means to recruit the best professionals in the programming field.

In learning contexts, the management of training sessions and competitions among the students and its integration in the whole learning process is not an easy task and requires specific learning systems.

This paper presents the design of a distributed asynchronous system to allow remote evaluation of students' submissions. More specifically, this system is being developed in the context of a project called "Integrating Online Judge into effective e-learning", with "Edujudge" as short name, a project funded with the support from the European Commission in the frame of the *LifeLong Learning Programme (LLP) 2007-2013*. The *Edujudge* project has as starting point two elements with evident synergies: the long time established on-line programming trainer *UVA On-line Judge*[7] and the competitive e-learning tool

---

[1] E-mail: luireg@tel.uva.es
[2] http://icpc.baylor.edu/icpc/
[3] http://www.ioinformatics.org/
[4] http://www.imaginecup.com/
[5] http://www.topcoder.com/
[6] http://www.google.com/codejam/
[7] The *UVA On-line Judge* - http://online-judge.uva.es/ - has been developed under the management of Dr. Miguel A. Revilla.

QUESTOURnament[8], which has also been developed in the University of Valladolid. The *Edujudge* project started on January 1st, 2008 so that the work is still in an early phase of development, in spite of which this paper expects to provide a clear view of its current state, and more important, of the work to be developed.

2. The *UVA On-line Judge*

The *UVA On-line Judge* is an on-line programming trainer created in 1995 with the aim of training users who participate in worldwide programming competitions. Currently, the *UVA On-line Judge* has more than 70,000 users from an important number of different countries (see Fig. 1) and more than 2,000 problems[9]. Some users are really active and more than 10,000 users have submitted more than 100 problems to the system.

Users can send programs which solve the different problems to the *UVA On-line Judge* and the system automatically provides feedback indicating if the code is correct or not. Besides being a training tool, it allows to organize on-line programming contests. Most of the current users of the *UVA On-line Judge* are university students and teachers but there are also some users who are students and teachers from secondary education.

At the moment, the *UVA On-line Judge* is oriented to programming contests and training, but it can become an innovative education tool, mainly for the field of programming, but also for other fields like mathematics. In fact, the current users of *UVA On-line Judge* are demanding a greater pedagogic character for this tool. For example, a lot of programming teachers would like to use the *UVA On-line Judge* for activities organized in the frame of the official courses that they teach. This would require, at least, the possibility of managing courses and users, both students and teachers, as well as an extension of the current functionalities of the *UVA On-line Judge*. The new judge should provide, for example, the possibility to offer a gradual evaluation or to classify the problems according to their level of difficulty.

On the other hand, the set of problems is continuously being incremented with the contributions of the users. Therefore, it is necessary to give the problems an adequate and common structure, adding metadata and creating a search engine so that the problems are more accessible for the community of teachers and students who want to benefit from them during a learning process.

There are different on-line competitions at international level like the ones mentioned in the Introduction section of this paper and also different on-line judges, most of them promoted by China, like the *Zhejiang University Online Judge*[10], and Russia, like the *Saratov State University Online Contester*[11] or the *Timus Online Judge* managed by the *Ural State University*[12]. Other examples are the *Sphere Online Judge (SPOJ)*[13] in Poland and our *UVA On-line Judge* in Spain. However, all these on-line judges analyzed at international level lack of the demanded pedagogical character, reason that turns into a priority to provide the *UVA On-line Judge* with that pedagogic character.

Fig. 1. Degree of participation in the UVA On-line Judge at international level. [2]

[8] The *QUESTOURnament* competitive e-learning tool has been developed by the research group Infraestructuras, Tecnologías, Aplicaciones y Servicios de Telecomunicaciones (ITAST) - http://www.itnt.uva.es/ and http://itastserver.tel.uva.es/formacion/.
[9] http://icpcres.ecs.baylor.edu/onlinejudge/
[10] http://acm.zju.edu.cn/
[11] http://acm.sgu.ru/
[12] http://acm.timus.ru
[13] http://www.spoj.pl/

3. The *QUESTOURnament* tool

QUESTOURnament [1] allows teachers to configure both individual and group work environments with a set of intellectual challenges to be solved by the students in a time-constrained way. In most traditional environments, the questions and problems are only proposed by teachers. However, in *QUESTOURnament*, challenges can be also submitted, and the corresponding answers pre-evaluated, by the students. When a challenge is proposed by a student, the teacher must approve it before it is presented to the rest of students. The student will be rewarded depending on the quality of his/her work, including both the adequateness of the proposed challenge and the assessment of answers submitted by the other students to the challenge proposed by him/her.

One of the most interesting characteristics of *QUESTOURnament* is the scoring system, which plays a very important role, since the sessions are presented as a contest. Once submitted, the answers to the challenges are rewarded by means of a variable scoring system that intends to stimulate participation and to give an incentive to the effort. As seen in Fig. 2, the *QUESTOURnament* system shows all the time the current summarized ranking with a direct access to a detailed scoring board. The classifications can be individual and per team, in case teams have been organized among the students. The main area of the screen shows all the challenges proposed by teachers or students. Although, as it has been explained before, the challenges proposed by students must be approved and scored by the teacher, it is possible to configure the system to allow automatic approvals too. In fact, an important design premise has been the flexibility: the teacher has a lot of ways to use the system (individual contents or by teams), the maximum and minimum score can be set, the number of answers to be accepted can also be configured, etc.

Fig. 2. Extracts of the main screen of QUEST system in which challenges proposed by students and teachers are shown as well as a summarized ranking

Since the time when a challenge is created until the end of the process, the challenge goes through different phases and states. On one hand, the life-cycle of a challenge is as follows:

- Stationary phase: score remains as proposed by the teacher during a period of time to allow students to understand and to take in the task.
- Inflationary phase: score grows to adjust the reward to the difficulty level of the challenge. It is assumed that a lack of correct answers means that the difficulty is higher than the reward.
- Deflationary phase: once a challenge is correctly answered, the score starts decreasing so that the student who is the first to answer gets the maximum score.

On the other hand, challenges can be on a set of states as follows:

- Approval pending: the challenge has been proposed by a student but a teacher has not approved it yet.
- Start pending: each challenge has a start date and an end date. Answers can be received only during the intermediate period. Besides, before the start date, only who proposed the challenge and/or the teacher can access it.
- In progress: the challenge is fully active, answers are received and scoring is varying.
- Closed: the time to answer is over, and no more answers are allowed. The students can read all submissions.

As a result, *QUESTOURnament* is a dynamic and changing environment in which the students are content generators and participate in the learning process in an active way.

QUESTOURnament is available in eleven different languages and allows the development of contests, individual or by teams where the members cooperate and compete with each other. As supported by

telematics, *QUESTOURnament* can be used within the classroom or on a remote basis to support a distance learning process. The system has been implemented as a module which can be integrated into the open source e-learning platform *Moodle*[14].

4. The *Edujudge* Project: Integrating the *UVA On-line Judge* and the *QUESTOURnament* tool

Previously to the Edujudge project, the *QUESTOURnament* system only allowed the launching of questions or challenges with an open answer format, but it can be adapted to allow other kind of challenges. The types of problems which are used in the *UVA On-line Judge* are among those that match perfectly with the philosophy of the *QUESTOURnament* tool.

Through the integration of the *UVA On-line Judge* into the *QUESTOURnament* tool, which runs on the *Moodle* e-learning platform, some of the pedagogical functionalities demanded by the current *UVA On-line Judge* users would be provided by the e-learning plattform. The distributed nature of the project will alleviate the bottleneck of user management and authentication of users who are provided with a broader access to problem's details. Other functionalities demanded by the current *UVA On-line Judge* users, such as gradual evaluations or the categorization of the problems according to their level of difficulty, will require a complete new development.

The main goal of the *Edujudge* project is to give a greater pedagogic character to the *UVA On-line Judge* and to adapt it to the needs of a formal educational environment. More specifically, the *UVA On-line Judge* will be used in the frame of courses regularly delivered in educational institutions such as Universities, secondary schools or Vocational Education and Training (VET) centers.

From a technical point of view, the tasks to be developed include:

- An improvement of the accessibility and usability of the problems by defining metadata and creating a remote problem repository. In this way, problems will be more accessible for the community of teachers and the reuse of resources will be promoted.
- Turning the *UVA On-line Judge* into a more intelligent system that is able to classify problems according to their level of difficulty and to undertake a gradual evaluation.
- The development of *a distributed* system for remote and/or automatic evaluation so that the application field of the system can be extended.
- The effective integration of the *UVA On-line Judge* into *QUESTOURnament* and into other services available in the open source e-learning platform *Moodle*.
- A decentralized management of courses and students by means of installing the e-learning platform *Moodle* with the *QUESTOURnament module* locally at every institution that uses the *UVA On-line Judge* with a multilingual User Interface (UI) translated into different European languages.

5. A Distributed System for Remote and/or Automatic Evaluation

The need for externalizing the evaluation of different learning activities has arisen in order to make educational systems independent from the agents which undertake the evaluation. In *Edujudge*, the evaluation agent is the *UVA On-line Judge* engine, which provides automatic corrections.

Fig. 3 shows the Deployment Diagram of the distributed asynchronous system proposed for remote evaluation. The system consists of an UI properly integrated into *Moodle* and into the *QUESTOURnament* module, a server prototype with catalogue and grading services for the *UVA On-line Judge* and a control subunit in charge of orchestrating and refreshing the interactions of the former parts. The UI will offer two services: general interaction with students and teachers and management of statistics of use and achievements.

[14] http://moodle.org/

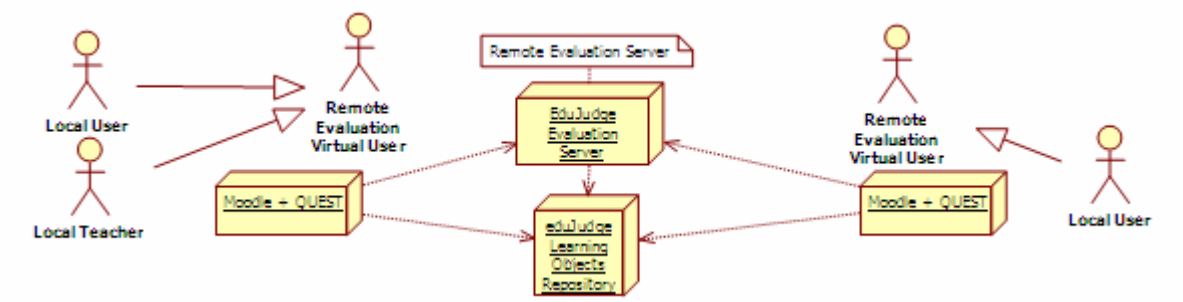

Fig. 3. Deployment Diagram of the Distributed System for Remote and/or Automatic Evaluation.

The management of the different type of users will be done locally at the sites where the system will be used (such as Universities, secondary schools, etc.), where *Moodle* and the *QUESTOURnament* module will be installed. A remote evaluation virtual user will act as a holder for every user in the local e-learning platform. This way, the evaluation server must only check permissions on a per-site basis.

Although the *UVA On-line Judge* is focused on programming learning, the distributed system for remote and/or automatic evaluation is being designed to be able to extend its use to other fields like mathematics or language learning, for example. Actually, the system can be applied to any kind of problem and in different education levels which evaluation can be outsourced. In fact, it should be possible to access not only to different evaluation servers including automatic evaluation engines like the *UVA On-line Judge* or a database automatic evaluation server, but also to human evaluators such as experts in a subject, native speakers, translators, etc.

Moreover, all services will be standard-oriented and publicly exposed so that other platforms and systems can interact with the *EduJudge* components. As every piece of *EduJudge* is modeled as a component, every part is replaceable with alternative or multiple components (i.e. one *EduJudge Evaluation Server* can interact with many Learning Object Repositories).

Problems are to be modeled as "Learning Objects" and the resulting "Learning Object Model (LOM) for algorithms with automatic evaluation" will be implemented and published. In this way standard e-learning platforms and tools will be able to use the existing problems, provide new ones to the *EduJudge* repository or to generate this kind of objects from their current standard LOM repositories. In addition, the *EduJudge* repository is planned to receive feedback from clients (on *Moodle*) and evaluation servers in order to implement an automatic classification of problems according with that feedback.

The evaluation engine will be a stateless component with a thin layer of transaction control exposedas a web service. As evaluation can be a heavy weight process and evaluation servers can be quite busy, the grading process should be asynchronous and, hence the evaluation server should retain some temporal knowledge of the client's activities.

The implementation of the client part of the system will be focused on reutilization. Hence, a platform-neutral Application Programming Interface (API) will be developed to allow easy deployment on other PHP-based platforms. Also, all code will be made available to the community in order to facilitate the dissemination and implementation of this tool in different Universities, secondary schools, etc.

6. Current State of Work

As commented in a previous section, the *Edujudge* project just started in January of the present year, which means that the work is still at an early phase of development. More specifically, regarding the distributed system for remote and/or automatic evaluation, the current work is focused on defining the specifications of the system.

Moodle uses an abstraction of the questioning and grading processes by means of what is called "question engine" by which many question types can be plugged-in into the platform. This API is the natural way to enhance the functionality of the questioning modules as *Quiz* and, of course, *QUESTournament*.

The first stage of the project is focused on defining the communication protocols between the presentation module (*Moodle* part) and the server modules (repository and evaluation) by designing some Simple Object

Access Protocol (SOAP) interfaces and a set of orchestrated sceneries or use cases. Also, some work is starting regarding the repository and the evaluation engine components.

The work currently under development aims to:

- Allow teachers to define quizzes with an arbitrary number of questions of different types including those specific of *EduJudge*.
- Allow students to view questions and to submit pieces of code that are potential solutions to the proposed questions.
- Ensure that the system correctly orchestrates the transactional protocol until the quiz is properly graded and conveniently recorded, summarized and notified to interested parties (i.e. client and repository server).

Adapting the current version of *QUESTournament* to this schema is the next stage of the development. This involves refactoring the current code and generalizing the current contest logic in order to be more extensible and to allow different work flows to be deployed following a plug-in pattern with the greatest reusability of common services such as teaming, grading, scoring, etc. As result a "contest toolkit" will be designed and developed.

The modeling of the Learning Objects for remote evaluation is an open issue. These items should contain all the information needed to allow users to understand the problems, and evaluation servers to automatically evaluate and grade a giant number of possible algorithms with an open set of grading methods. On the other hand, the repository should be able to automatically collect activity information and to enrich the classification of the Learning Objects with this information. This activity is also under intense work at present.

Automatic evaluation of the submitted codes is not a simple task. Current implementation heavily relies on strictly formatted input-output sets, which imposes serious limitations to the kind of problems that can be evaluated. In this project, a more versatile and open method to test and to grade the accurateness of the submissions will be developed. Moreover, some imaginative ways of testing programs can be implemented easily, i.e. interaction between different users programs (cooperation, combat, dialogue, etc). In addition some scalability and extensibility issues are to be addressed.

7. Conclusions

The deployment of a distributed system for remote and/or automatic evaluation in the frame of the *Edujudge* project will produce a number of benefits.

The *UVA On-line Judge* counts already with an important number of users worldwide mainly in Universities that will benefit from the improvement of the pedagogic character of the system. Moreover, the development of the *Edujudge* project will contribute to extend the use of the *UVA On-line Judge* to other scenarios such as secondary schools or VET centers.

Through a distributed and loosely coupled architecture of the system, many deployment sceneries can be possible. Multiple e-learning sites, repository servers and evaluation servers could be arranged and mixed easily, increasing the range of problem's variety, evaluation methodologies and, hence, raising the possibilities and potential utility of this system. Aditionally, resulting API and protocols would cover the general problem of remote evaluation, so many grading tools and services, far from programming questions, can be connected to Edujudge-abled sites.

Besides, as a result of promoting and extending the use of the *UVA On-line Judge* beyond the limits of programming subjects, teachers and students of other fields will be able to benefit from the system.

Currently, walking towards the European Higher Education Area (EHEA) convergence, Universities are working quickly to introduce innovative educational methodologies into classes and to incorporate Information and Communication Technologies (ICTs) into the educational processes. At the moment, a priority is to give students a more active role in their learning process and systems like the one proposed, based on the *UVA On-line Judge* and the *QUESTOURnament* tool, can contribute to support this process. It is also important to notice that the *Edujudge* system can contribute to make European education and training systems a world quality reference by 2010, as it is proposed in Education and Training 2010 European Policy.

8. Acknowledgement

The *Edujudge* project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

REFERENCES

[1] E. Verdú, L. M. Regueras, M. J. Verdú, M. A. Pérez and J. P. de Castro. "Improving the Higher Education through Technology-based Active Methodologies: A Case Study", *WSEAS Transactions on Advances in Engineering Education*, vol. 3, no. 7, 2006, pp. 649–656.

[2] S. Manzoor. "Analyzing, 'Programming Contest Statistics". Southeast University, Dhaka, Bangladesh. http://acm.uva.es/p/13_Manzoor_rev.pdf