Deploying University Study-Oriented System at Polish Universities

Janina Mincer-Daszkiewicz

Faculty of Mathematics, Computer Science, and Mechanics,
Warsaw University, Poland
jmd@mimuw.edu.pl

Abstract

University Study-Oriented System (USOS) is an integrated student management information system for academic structures and student affairs at Polish universities. The system has been produced in-house at the Faculty of Mathematics, Computer Science and Mechanics of Warsaw University for 17 Polish universities — participants of the New Educational Tools Tempus JEP project. The results of the project had been reported at Eunis 2002. The purpose of this paper is to describe further activities involved in development and deployment of USOS. The most positive aspect of the story is that the Tempus project helped to initiate a cooperation of Polish high education institutions which established a consortium for coordinating, supporting and conducting common IT tasks.

Keywords: Student Management Information System; USOS — University Study-Oriented System; UCI — Consortium of Polish high education institutions

1 Introduction

University Study-Oriented System (USOS) is an integrated student management information system, produced in-house at the Faculty of Mathematics, Computer Science and Mechanics of Warsaw University for 17 Polish universities — participants of the New Educational Tools Tempus JEP project JEP-14461-1999 (carried out between November 1999 and December 2001, cf. [4, 7, 8]). The results of the project had been reported at Eunis 2002 ([6]). The purpose of this paper is to describe further activities involved in development and deployment of USOS. There are four main subjects we want to discuss:

1. organizational structure, aims and tasks of UCI — consortium of Polish high education institutions, similar to Swedish LADOK [3] — established in 2002 with the purpose to support further development of USOS and its gradual deployment at the participating institutions,
2. directions of USOS development, in particular funtionality of its new modules and integration of USOS with various other on-line services,
3. methods, techniques and tools of software engineering involved in the project and in particular the role of USOS in teaching Computer Science students a practical software engineering,
4. odds and ends of USOS deployment at Warsaw University and other institutions cooperating within UCI.

At the end of the Tempus project the question being asked by the Tempus partners was: what are the possible scenarios for USOS and further cooperation, is any local initiative possible without euro-money? The most probable — as often happens with projects of that kind — was that the cooperation ends with USOS deployed at few departments as a local application. The other possibility was that at least home institution of USOS — Warsaw University — would deploy it at the university level. The most challenging was that the cooperation which emerged during Tempus days would continue finding other organizational framework and helping to support further maintenance and development of USOS and deploy it at at least a few universities. At the beginning of 2003 it seems that the last scenario has many chances to succeed. The most positive aspect of the story is that the Tempus project helped to group institutions in higher education in Poland willing to work together on IT tasks. This process is in progress.

2 UCI — consortium of Polish high education institutions

At the end of the Tempus project many questions had risen concerning the status and future of USOS? Who owns it? Who should be responsible for or interested in its further maintenance and development? Who should pay for it? Who should be in charge? Fortunately Polish universities were not the first to be confronted with such questions. Having in mind an example of LADOK — the consortium of Swedish high education institutions, which was one of the EC partners in the Tempus project — a decision had quickly been made to try to follow the Swedish solution. It took a year to work out all legal details and finaly at the end of 2002 the consortium called UCI (an English equivalent of University Center for Computerization) has been established. The aims listed in the signed agreement were to coordinate, support and initiate activities such as:

1. designing and developing computer based information systems,
2. working out regulations concerning automation of various administrative procedures,
3. gathering common funds for IT projects,
4. representing partner institutions in talks with the Ministry of Education,
5. negotiating with companies selling IT software and hardware.

The administration structures of UCI consist of:

1. A Council. Each participating institution is represented in the Council by rector's plenipotentiary responsible for IT or other authorized person. Each institution has the same voting strength. The Council is responsible for acceptance of the annual report of the board, acceptance of the fiscal balance, election of the board and its chairman, acceptance of new members of the consortium, approval of the budget for the next fiscal year.

2. A Board of Directors. It consists of 5-11 members. Chairman of the Board of Directors is the chief executive officer of the consortium. The Board is responsible for deciding on the overall and principal direction of activities, budget proposals, production of the annual report and fiscal balance, signing agreements with other companies, supervising Project Commisions etc.

3. Project Commisions. A group of UCI members may launch a common project. Projects are supervised by Project Commisions consisting of representatives of the institutions belonging to the group. The Project Commision decides about the project budget, hires a Working Team which is responsible for carrying the project, appoints a project leader, sets task and priorities for the team.

USOS is one of such projects (the main one at the moment). To clear legal status of the software the ownership of USOS had been transferred from the Tempus partners to the consortium. The USOS Commision consists of 7 representatives of universities participating in the USOS project. The USOS Working Team has been located at the Faculty of Mathematics, Computer Science and Mechanics of Warsaw University which is the home for USOS from the beginning. The consortium entrusts the Faculty with the maintenance and further development of USOS as stated in the agreement between the consortium and Warsaw University.

The cost of system maintenance and development is covered from the fees of the member institutions based on the project budget. Fees are low and devoted for basic system maintenance. Fees can be adjusted every year. Consortium may also apply for external funds from possible sponsors in addition to routine annual operating budgets.

The USOS Commision meets once every few months and functions as a forum for information exchange between the users and the USOS Working Team. It is responsible for stating and ranking maintenance duties for the team.

The USOS Working Team coordinates all design and programming work in accordance with the directives of the USOS Commision, debugs the system, issues new USOS distributions, takes responsibility for development and maintenance of interfaces to other systems, supports help-desk, prepares system documentation and manuals, assists new members with converting data from old databases to the USOS database, suggests tasks and ideas for further development, informs other parties about USOS. The annual budget of the project should generally cover basic daily system maintenance. Any special tasks concerning development of brand new modules may need special funding. New modules may be either developed by the team, or delegated to commercial software companies and conducted under the supervision of the team. In particular projects run by students (cf. p. 4) should be approved by the team, which is also responsible for integrating the project results with the system. On the other hand local system deployment and usage is the responsibility of the institution, who pays for system hardware and software infrastructure and local support, i.e. it is not conducted by the USOS maintenance team.

17 institutions out of 17 taking part in Tempus joined the consortium, 14 joined the USOS project. Consortium is an open organization in a sense that high education institutions which had not participated in Tempus but would like to join can be accepted under the condition that they would somehow pay back an investment cost. The possible way of doing this consists in funding a new USOS module. Two new institutions joined the USOS project.

3 USOS — achieving new quality by integration of services

USOS offers basic functionality, like handling students' and teachers' personal data, study programs, requirements of degree certificates, course catalog, course registration, class schedules etc., but many more functions are needed. It can be observed that the more system delivers, the more is expected and asked for by the growing number of its users. It looks like the new needs emerge out from the simple reflection that most of the data IS already there at our finger tips and only a new interface should be built to view it from a different perpective.

University Payment System and Speed Collect

Various larger and smaller projects have been carried out in a year 2002 in order to meet new demands and changed prerequisites. The most important one, called USP (an English equivalent would be *University Payment System*) is the system for handling students' payments. In Polish public universities there are generally two kinds of payments. Some students, so called **day-time students**, don't pay regular tuitions, but are charged for failed courses which have to be retaken, failed promotion to the next stage of study, missing credit points etc. So called **evening-time students** pay regular tuitions. The amount of the tuition may be divided into 1, 2, or more parts which should be payed in due dates. Delayed payments are increased by extra penalty rates.

The university signed an agreement with one of Polish banks (a member of an international group) offering a service called SpeedCollect. Each student of the university (possible debtor) is assigned a unique number. This number is used as an extension to the account number of the department which is an extension to the account number of the university. Students transfer money they are due to those accounts. Every day information about receivables is delivered by the bank to the department electronically in a form of a structured file. That file is imported into USOS, interpreted on-the-fly and all students' due payments are automatically reconciled with receivables which are easily matched with the students by the account numbers.

All involved parts profit from this system of electronic data processing and transmission.

The department has the money credited to the account promptly, knows immediately who has payed and can eliminate delays in updating of accounts receivables. Additionally manual input of data into the financial/accounting system is eliminated. It also becomes possible to take a quick action against students who have not paid receivables on the due date, eg. by sending automatic e-mail with the warning message or omitting them on examination lists.

Students have immediate and remote access on the web ([5]) to personalized information about due payments, due dates, account numbers, penalties for not paying on time etc. They may transfer money electronicaly from their accounts in electronic banking system by simple copy and paste from one web page to another to avoid any misspelling or at least print money transfer forms with bar codes (with all necessary data, like the amount due and the account number, already there). They may also open a special electronic account called **virtual wallet**, preload it with some amount of money and then electronically transfer money from the wallet to the university account (or use it to pay for on-line purchases through the Internet). Such payment is authorized on-line both by the bank and by USOS. A full transaction history is also available on-line.

The bank makes money charging a fee for every transaction.

It is also possible to accept cash directly at the counter and enter the information manually to the system.

USP delivers all kinds of reports, like the list of students who have due payments or students who have to pay extra charges for delayed payments.

Financial aid and scholarships

The other large module designed and implemented during 2002 is the module for handling financial aid and scholarships.

Students apply for financial aid, delivering information about the average income in the family per capita. The global amount of money reserved for the financial aid is also specified. Then that money is distributed among students according to one of many various algorithms implemented in USOS. Information about money transfers from the university account to individual student accounts are delivered to the bank on files. Students getting best grades apply for scientific scholarships. Money for that purpose is also distributed among applicants according to one of the predefined algorithms, on the basis of the calculated grade averages.

Incoming modules

Among the incoming modules are:

1. a system for registering students for language courses. The rules of registration are different than registration for regular courses, so the new module for that purpose had to be designed.

2. document management system supporting automatic delivering and flow of applications. Since we want to archive all student applications the idea is to post special web forms for editing such applications and sending them to the Student Office. Students should have the possibility to check that the application has been accepted and observe its further flow to the deputy dean for students' affairs, and back to the clerks at the Student Office. They should also see status of the application at each stage and the final decision.

Other plans

The general direction of development is to integrate various information handling services in one coherent system, available through a common unified interface, authorized by a common authorization layer. Of course their is still a lot to be done in this respect, for example university payroll and human resources systems are separate applications. But from the very beginning it is assumed that those university-wide databases should some day be integrated on-line so at the moment at least various off-line connections are worked out. Such policy should help to achieve a new quality in administrative software.

4 USOS — a testbed for practicing software engineering

From the beginning the USOS project has been carried out at the Faculty of Mathematics, Computer Science, and Mechanics of Warsaw University by students supervised by academic teachers and Ph.D. students. Software was designed and developed on software engineering and database courses, and on master seminars. Whenever we describe that scheme of system development everybody doubts that it can work in a long run. Many negative examples are given of projects carried out in a university environment which failed without delivering any useful outputs ([8] - private talks). To our own surprise in our case the situation is opposite. The longer project runs the more stable, mature and diverse in character are sub-projects carried out by students. It seems that the large application like USOS, which is totally under control of one project team, brings out new ideas which may be worked on by students on various platforms, from different perspectives, using a plethora of tools and methods.

The early student projects were generaly focused on producing particular functional modules. Examples of such projects were those for handling course registration, study programs, requirements of degree certificates, class schedules. Design was a responsibility of academic teachers and Ph.D. students. Some of the projects involved Oracle programming, others PHP and MySQL programming. Recent student projects are more diverse, focused on design and methodology, more software engineering oriented. The examples are *Describing USOS in the Unified Modeling Language*, *Calibrating and maintaining large scale databases*, *Design of user-friendly interfaces*, etc. Software reengineering, data maining, database profiling, software modeling — these are the main domains.

The most important difference between usual academic projects and USOS projects is anormous: the latter HAVE to deliver a final product and the author HAS to take part in its deployment. Students build teams, practice new software engineering methodologies, validate various software development techniques, play various roles in the project (programmers, sub-project leaders, system administrators etc.) They have to work in a strict time regime, use software version management systems, adopt standards, prepare and issue software distribution packages, use bug-tracking systems, feel responsibility for the final product, use flexible solutions (the produced code will be maintained and reworked by somebody else). We teach by example and practice. If a student cannot understand what "user-friendly interface" really means just made him sit for a day at the help-desk for administration officers! Or make him responsible for training end-users how to use Oracle forms! Next time he will really make the best effort to design a simple, self-explained and intuitive interface. The code is often refactored when old solutions are replaced with new better ideas. Some of the students fail (even if they produce software, it is not attached to the official version of USOS), majority succeed — all learn "programming in the large" and practical software engineering. Software companies producing commodity software rarely can afford experimenting with various techniques, methodologies and tools in the same project. We can — USOS becomes a testbed for such experiments. What a year or two ago might seem to be a risky didactic experiment is now an established multilevel process.

In [6] Oracle CDM Fast track (cf. [2]) was described as the leading methodology used in the project. Last year due to growing scale of the project we decided to start using **UML** and **Rational Rose**. Oracle design tools play a main role in design and implementation phases but do not support sufficiently the requirement analysis process. The Unified Modeling Language can be applied to various areas of software development, such as data modeling, enhancing practitioners' ability to communicate their needs and assessments to the rest of the team. UML can also be used to describe the complete development of databases from business requirements through the physical data model. In USOS use cases modeled in UML become standard tools used during talks between developers and clients for working out system requirements.

Practices applied in USOS in many respects match the rules of **extreme programming** (cf. [1]). The most important ones in USOS are:

1. plan in a short run and in a long run,
2. conform to standards,
3. make it simple,
4. work closely with the clients,
5. integrate constantly,
6. often deliver prototypes.

There is also another aspect of students' involvement in the development of USOS. To cut costs universities want to deploy and suport the system in a local environment with the help of local experts. However the best way to get such experts is first to train them and then hire. Looking for such local experts and planing future recruitment other UCI members started offering Master seminars or other USOS-oriented courses.

Of course the development of USOS by student groups have to be supervised by members of the USOS working team hired by UCI. Sometimes in one sub-project work together hired-programmers and student-programmers ("hired-programmer" may also mean a student hired to develop a particular piece of software). There is always a clear distinction between those two ways of involvement. For some students that may even be an extra motivation — they may get an opportunity to earn money if they prove their value first.

USOS gave us a unique opportunity to integrate education in software engineering with production of commodity software. We are determined to continue that model of software development.

5 Deployment of USOS at Polish universities

USOS is being deployed at universities by local teams. We will present problems, procedures and steps involved in the process.

Generaly the deployment may proceed differently at various institutions according to local priorities and the involvement of resources. Even faculties of the same university may choose different procedures. For example the deployment may be **vertical**: a few faculties deploy majority of USOS modules, or **horizontal**: one or two modules are deployed at all faculties (for example the module for handling student financial aid).

Let us share the experience of the University of Warsaw. From a year long perspective we may identify those conditions and events which positively influenced the deployment process and those which were possible pitfalls. We also list the sequence of steps of the deployment procedure.

The following conditions influenced the process positively:

1. Rector made and announced the decision to deploy USOS at the university level and stop support for old desktop-like systems (e.g. the central financial system will only accept data on student financial aid in a format agreed upon and implemented in USOS).

2. Central deployment team has been established. It groups system developers, university administration officers (student affairs, financial affairs), university computer systems administrators and programmers.
3. Deployment may proceed in stages, according to faculty priorities.
4. University authorities (rector, deans of the departments) and administration officers are strongly motivated to achieve the goal.
5. People seem to understand that if not now then never ...

However there are also possible pitfalls. Departments of Warsaw University vary substantially in size, have high degree of autonomy, and different priorities. The degree of computerization at various departments – comprising hardware, software, and peopleware — is different (often very low). Deployment of a large software system demands high organizational effort and a lot of (extra) work, which demands strong involvement of human resources. But people are busy, salaries are low etc.

In Warsaw University the deployment process proceeds in the following steps:

1. At the university level common dictionaries for USOS and payroll and Human Resources database are defined;
2. The dean of the department officially asks the central deployment team for help with the system deployment;
3. Departamental deployment team is established;
4. Departamental staff personal data are transferred from HR database;
5. Students' data are transferred from old desktop databases or spread sheets (only some faculties have such data in an electronic form);
6. Departamental course and degree catalogs are built;
7. In the meantime USOS is installed on departamental computers and people are trained (at their work places);
8. Roles and filters for administration officers are defined according to requirements;
9. Administration procedures are verified along the way.

In January 2003 the situation is the following: one department of Warsaw University deployed 100% of USOS functionality, one department 60%, one 40%, five departments which started recently about 20%. The first observation is that SUCCESS strongly depends on PEOPLE. Even with the same level of support and help from the deployment team, the progress made varies substantially. The second observation is that the deployment of such a huge software system is like an avalanche — when started, can not be easily stopped.

6 Summary and Future Plans

In January 2003 USOS is used on daily basis at four universities (at pilot faculties), a few others are at various stages of deployment. At Warsaw University we started academic year 2002/03 with USOS at four faculties. At the beginning of 2003/04 all faculties will be using at least one module - Internet registration module for foreign language courses.

We are still in the process of working out details of the optimal organizational framework for cooperation of participating universities within UCI consortium. Transfer of knowledge between institutions becomes a fact when the system is running. This factor may be one of the most profitable for UCI members. This pays off by doing things the right way from the beginning and not repeating others' mistakes. A group of institutions has better possibility to influence government and funding organisations by their mutual efforts as a discussion partner. With the co-operative system development approach, the result is a highly standardized system with its major components available for the owners.

A few software companies are interested in integrating their (commercial) products with USOS and offering deployment services.

Last but not least the number of USOS Master and Bachelor Theses grows steadily — there have been 15 by now, many more are on the way.

References

[1] K. Beck, *Extreme Programming Explained: Embrace Change*, Addison-Wesley, 2000.

[2] S. Gylseth, *Using CDM Fast Track, Oracle's DSDM Compliant RAD Approach*, Oracle Corporation, 2000.

[3] Home page of LADOK consortium, Umea, Sweden, `http://www.ladok.umu.se`.

[4] Home page of USOS, Warsaw, Poland, `http://usos.mimuw.edu.pl`.

[5] Home page of USOSweb, Warsaw, Poland, `http://usosweb.mimuw.edu.pl`.

[6] J. Mincer-Daszkiewicz, *Student Management Information System for Polish Universities*, Eunis'2002, The 8th International Conference of European University Information Systems, Porto, Portugal, June 19-22, 2002, pp. 271–281.

[7] J. Mincer-Daszkiewicz, *Developing and Deploying Commodity Software in Academic Environment* (in Polish), KKIO'2002, IV Krajowa Konferencja Inzynierii Oprogramowania, Poznan, Poland, October 15-18, 2002.

[8] J. Mincer-Daszkiewicz, *Student Management Information System For Polish Universities*, SAIAC'2002, Joint International Conference on State of the Art in Administrative Computing, Tartu, Estonia, November 18–20, 2002.