

UNIVERSITY ELECTRONIC MANAGEMENT SYSTEM OF DNIPROPETROVSK NATIONAL UNIVERSITY. MAIN PRINCIPLES AND FEATURES

© S.V. Chernyshenko, G.V. Baranov, A. Degtyarev, V.S. Chernyshenko

Dnipropetrovs'k National University (Dnipropetrovs'k, Ukraine)
E-mail: svc@a-teleport.com

The main target of the SMOOTH international project of EU Tempus-Tacis programme was designing an automation computer system of university management for Ukrainian and Russian University. The Ukrainian team was responsible for the system design for Dnipropetrovsk National University.

Enterprises and institutions automation systems typically provide such key possibilities as: standardized data sources and information inputs; data storing and representation; standardized visual documents and forms interface; managed universal directories, collections and lists of ordered data that can be searched and used freely in the system; analytical data processing and report builders; flexible navigation between a parts of the system; security and safety; scheduled automatic processing.

The main stages of projection of enterprise automation systems include: enterprise business model developing and analyzing; enterprise business logic formalization; application platform, engine and key technologies selection; application architecture projection; application code designing and testing; application operation and maintenance.

1. Computer network of Dnipropetrovsk National University, Ukraine

Dnipropetrovsk National University (DNU) is one of the leading universities in Ukraine according to the number of students (about 15000) and staff (about 1500, including 600 professors). There are 18 faculties and institutes with 104 departments in the university structure. The university provides education in 78 specialties covering all main fields of human activity. DNU has 18 studying buildings and 7 student hostels; its principal territorial distribution is represented in Fig. 1.

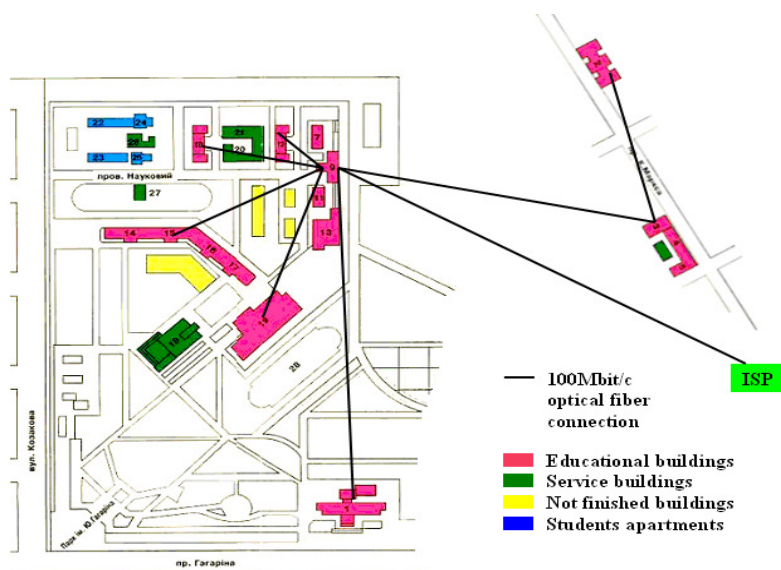


Fig. 1. The university campus and its computer network

DNU is quite a difficult object for the automation both from territorial and organisational sides. Taking into account the complex management structure, the main idea of improving university management in DNU is considered as creation of a full-scale computer network and optimization of information flows in it. A big distance between university buildings creates a possibility to design multi-server hierarchical structure of the network. DNU built special fibre-glass connections between the main buildings; local networks are based on the twisted-pairs technology. The computer network has only one Internet portal, DNS, firewall and so on. The principal scheme of the network is shown in Fig. 2.

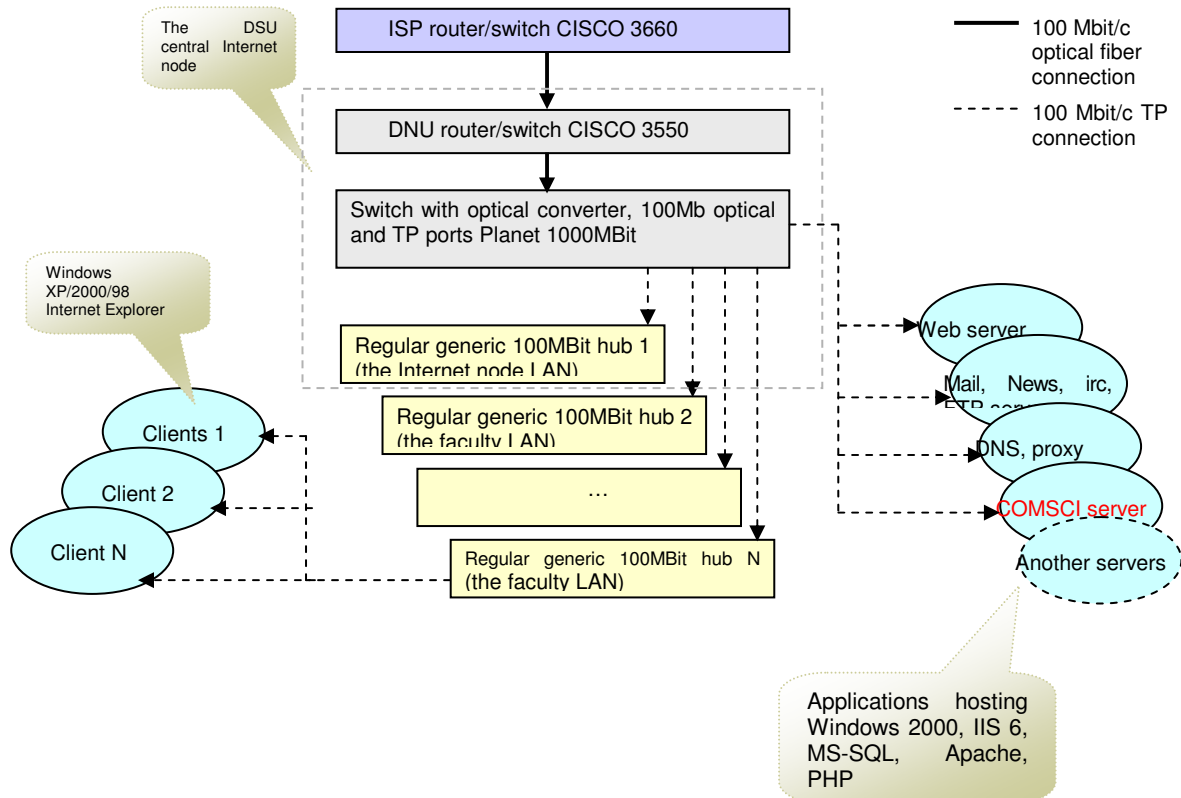


Fig. 2. The key principles of the DNU network topology

The architecture of the DNU network is distinguished by the following characteristics:

- TCP/IP-oriented LAN and WAN network technologies.
- Virtual LAN IP addresses and connection to the Internet through the proxy server only.
- One Internet service provider (ISP).
- One Central Internet node.
- Independent connection of each university building.
- Optical fibre connections of all buildings.
- One data server per each faculty or department
- The main border gateway (router/switch/converter) under Red Hat Linux.
- The e-mail, news, ftp, irc server, user accounts server under Free BSD Unix.
- The main web server under Red Hat Linux, apache.
- DNS and proxy servers under Free BSD Unix.
- COMSCI faculty server under Windows 2000 server, Active directory Domain controller, IIS, ASP, MS-SQL server, Net framework, Apache, PHP, Java server, users account and faculty LAN services server software.

- SMOOTH project server under Windows 2003 server, IIS, ASP, MS-SQL server, Net framework, Apache, PHP, Java etc...

2. University Electronic Management System

On the basis of the organised network a special managerial computer system has been elaborated under assistance of the SMOOTH project. We called it "University Electronic Management System" (UEMS).

The main objectives of UEMS are as follows: storage of all essential university data at the central server; operative providing decision makers with all necessary information; optimisation of information flows in the network; automatic preparation of the formal documentation. The main requirements to the system composition are the following: web-interface and remote access possibility; distributed data storage; simple modernisation of the system; high level system security; possibility of user profiles generation.

We selected the web interface as the main software technology, which is characterised by such positive peculiarities as minimal requirements to client hardware; no additional software needed (internet browser and text processor only); server-side software updates absolutely transparent for users; worldwide access.

The modern Internet environment was selected also for internal network environment, so the system is based on Intranet. We used client-server principles, the World Wide Web interface, and other technologies, associated with the Internet: web-servers; universal web clients; SQL database servers; Internet network communications; open data representation technologies, protocols and formats; open data processing libraries and network tools.

The key features of the World Wide Web technology, which can be effective in the enterprise automation engineering:

- Internet Application Architecture (IAA) and Web Application Architecture (WAA) as a standard.
- Open script sources and libraries.
- Powerful interfaces for all known Internet client-server systems including data storing, communications, searching and archives.
- Powerful libraries like network communications, operating systems, user interfaces controls and classes, raster and vector graphic, streaming data processing, encrypting, and others.
- Universal browser client's technologies.
- Universal open data representation and structural formats like HTML, XHTML, XML, DML, VML, UML, open binary documents formats like PDF, and so on.
- Full client support under all most popular and well known operating systems.

The internal architecture of Multi-user Enterprise Management System (MEM), on the basis of the IAA and WAA technologies, consists of three main parts:

- Input (generalization of the data inputs, formats and data pre-processing, like HTTP methods GET and POST, cookies, environmental and files sources).
- Processing (all operations with data, including database, file, network and other operations and the output content generation).
- Output (sending content header and data to the client and closing the network connection)

Code subdividing architectures include procedural, module, and mixed approaches.

Distributed database architecture of WAA engines and solutions will be used:

- Data storing systems (SQL servers, file and other data sources like ODBC drivers).
- Data schema structure (location and relations of databases, tables, fields and stored procedures).
- Subdividing the application on central or main server and local remote office parts, probably with different data storing engines (for example SQL server and file data source or ODBC can be used in one system corresponding in the main server and local office server).

- Data synchronization and load balancing principles and technologies (mirroring and clustering).
- Database schema and data security on the application level (application level developer database access secured API)

Mirroring, a most popular data synchronization and load balancing architecture, is selected (Fig. 3).

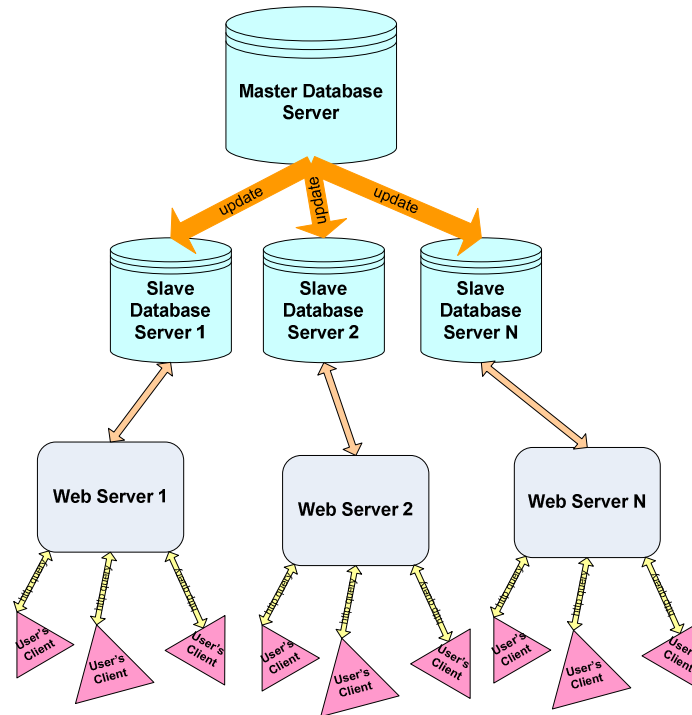


Fig. 3. The scheme of mirroring

The selected software instruments:

- Server platform (operating system) – MS NT-based OS (NT, XP, 200x).
- Web server – MS IIS web server (supplies with OS).
- Database engine – MS Jet (Access) for local servers and MS SQL server for central one
- Server side script language – ASP server-side technology.
- Client side script language – JavaScript as client-side script language.

3. Logical structure of UEMS

The hierarchical structure of UEMS is determined not only by territorial distribution, but also by organisational structure of the university management, by “administrative distance” between university divisions. Some corresponding principles are illustrated in Fig. 4.

The organisational structure should be reflected both in the database and programme interface. At first general question, appeared of the system, is the organization of the database, which one could put on either in general, or for every faculty own, local ones. We propose a special approach, when during the day faculties work with their local copies in the faculty servers, but during the night they are synchronised with the main university database in the central server (Fig. 5).

The level of security of the central server is essentially higher than that of the local ones, so the limitation of the direct work with it guarantees a reliable storage of the data. During the night the network traffic is low, and software environment was optimized so access to the server becomes faster.

UEMS is organized as an entire set of relatively independent modules and each of the modules performs a definite task or a set of similar tasks.

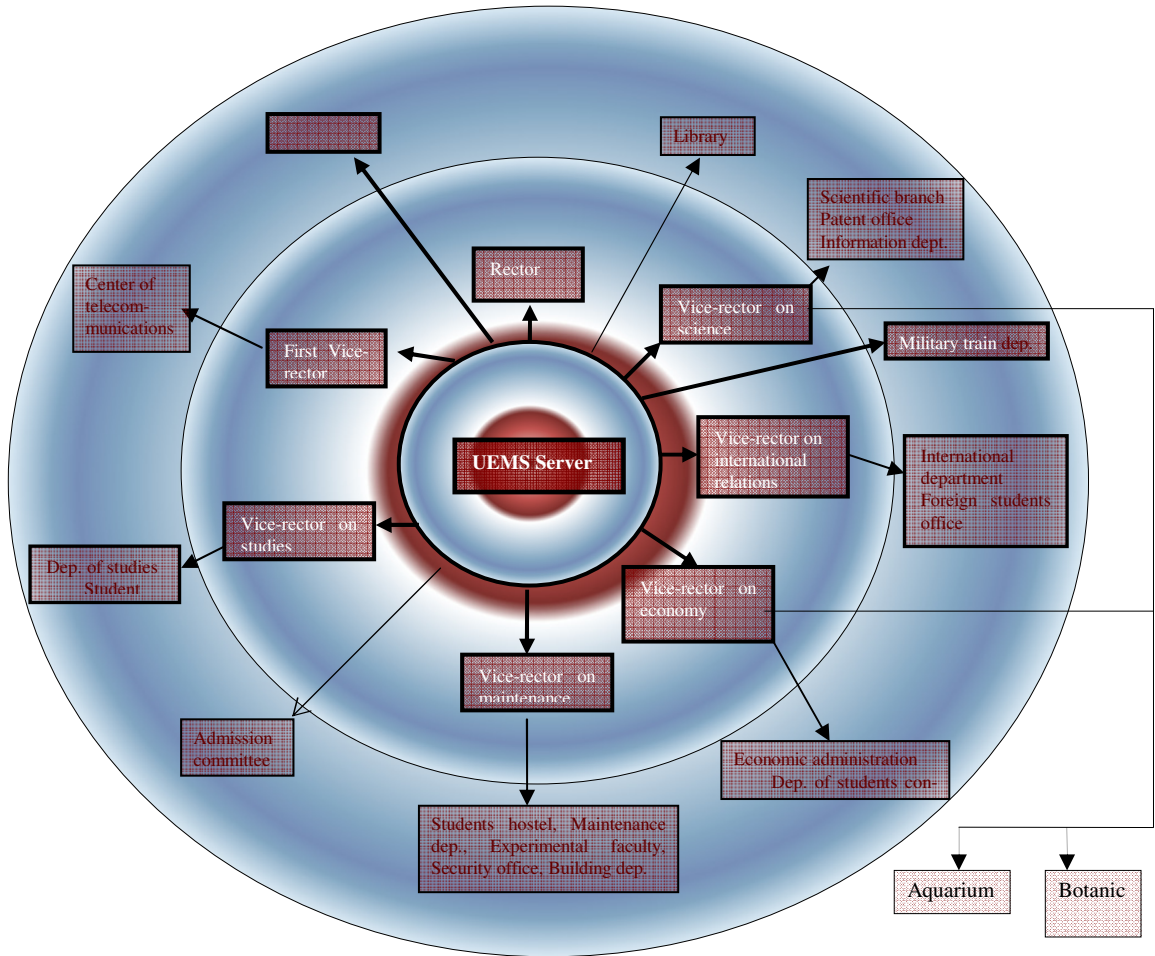


Fig. 4. Organisational structure of the DNU network

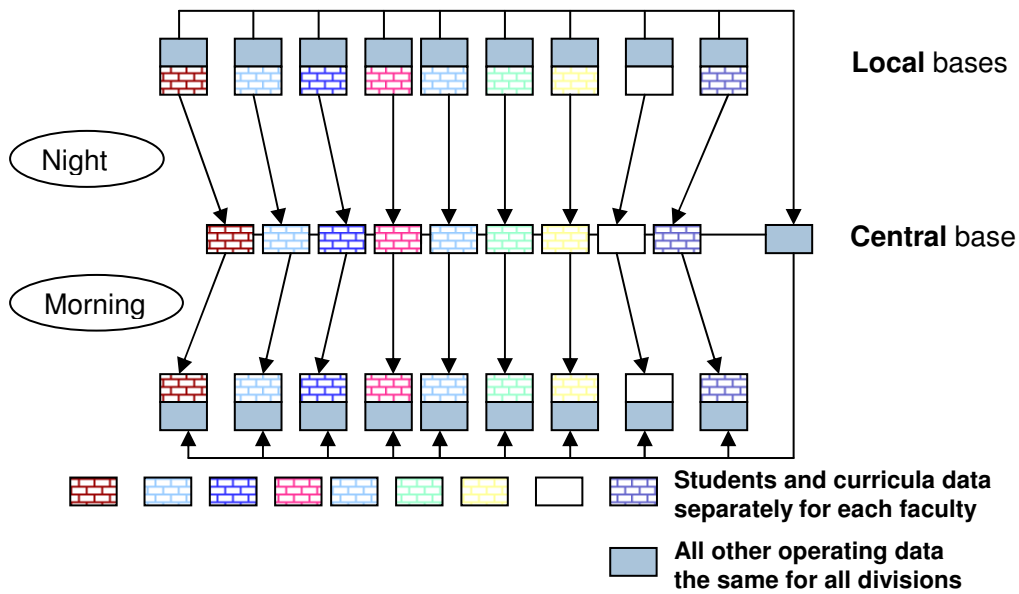


Fig. 5. Database storing concept

Internal logical structure of the system can be represented by several programmes: student database and interfaces; staff database and interfaces; “curricula” database and interfaces; retrieval information system; system of database presentation at the university site; “Virtual University”, an interactive system of the Internet presentation of teaching materials.

The programme interface reflects the annual cycling character of the university life. Correspondingly, the UEMS has the following separate interfaces (“steps”): 1. new academic year (summer-autumn); 2. new semester; 3. semester; 4. new academic year preparation (spring); 5. examinations; 6. retrieval sub-system. An example of the interface is represented in Fig. 6.

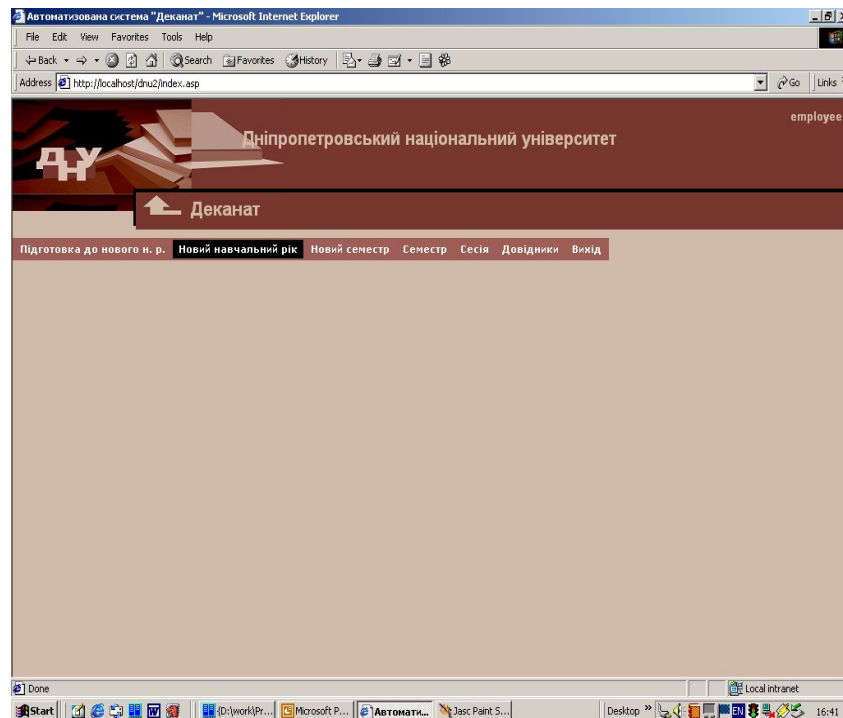


Fig. 6. Example of the general program interface (step 1)

Let us characterise sub-steps of the main steps.

Step 1. New academic year (summer-autumn) – orders of moving up into the next year of study or admission to the 1st year of study; forming of groups; students personal data input; generation of the lists of new students; preparation of graduation documents; practical training reports.

Step 2. New semester – redistribution of the teaching load; data preparation for timetable composition; studying timetable composition.

Step 3. Semester – blank templates printing; preparation of orders; table of the students' list change; teachers' work with students' personal data; preparation of the academic certificate.

Step 4. New academic year preparation (spring) – perspective curricula for bachelors, specialists, masters; working curricula; reports of departments readiness; assignments on teaching load.

Step 5. Examinations – exam timetable preparation; generation of lists of students for an exam; input of exam results; summary register of lists of students for groups; statistical analysis of exam results; orders of exclusion, for scholarships.

4. Curriculum preparation and correction

Let us consider, as examples, a few special parts of UEMS, and start from the programmes for preparation and correction of a study curriculum.

UEMS should serve as an efficient tool for the arrangement of the curriculum. If earlier this was put together by means of Excel and different other aids, now a web-based application should be responsible for this. As it had been expected, we confronted with different problems and difficulties connected with planning and production of the software. There were several solution attempts. Be-

sides, till present new ideas have come up, so that we must be ready constantly to integrate the changes.

Our purpose is modelling of the curriculum and on its basis modelling of the outgoing working plan. Besides, should become systematically gone forward to maximize the working efficiency. Alexander Rostilov and Leonhard Detzel, SMOOTH participants from the University of Koblenz-Landau, took part in rearrangement of the modules on ASP.NET and C#, as well as the export of the data in Excel for the wide treatment and for printing out of the curriculum. The corresponding graphic screen form is shown in Fig. 7, and more traditional table presentation is in Fig. 8.

The production of the plan is finished with the help of the web forms, which the user fills according to the previously agreed rules. Thus insists for him on selecting the possibility from the available disciplines, or on providing a new discipline. At the same time the inputs should be checked, corrected if necessary, or be pretended. The programming of the web forms, which should allow the production of the curriculum to the user of the system, included the definition and the integration of certain rules after which should be approached. Thus, for example, the user first of all, should determine the number of hours per semester, which will be maintained by the production of the curriculum. If this number falls short or is crossed, the system reacts with an error message. Validation should also provide for the error messages if a wrong value is given, e.g., text instead of a number.

After entire filling of the forms and based on this inputs, first of all a graphic representation of the simplistic curriculum is indicated, which shows the lecture types and contains information about lab hours, a training period, lectures and a kind of exam. Besides, it explains the fissures of the table, which was originating in this manner, the semesters in which the before selected lectures are put down. The lecture types are shown to the better overview in a colour code so that all lectures which belong to the same lecture type are indicated in their own colour.

Furthermore, the graphic representation (Fig. 9) shows the possibility of the correction and the balance of the lecture hours about all semesters, as well as the registration of the new disciplines. This occurs with the help of the web form which is directly about the graphic representation and also allows selecting the disciplines which were stored before in a .csv-file.

	A	B	C	D	E	F	G	H	I	J	K
1											
2		C	U	R	R	I	C	U	L	U	M
3											
4		1 year		2 year		3 year		4 year		5 year	
5		1	2	3	4	5	6	7	8	9	10
6		Physical training									
7		ts									
8		ts									
9		ts									
10		History of Ukraine	Business	Philosophy	Sociology	Polittology	History of religion	Ukrainian constitution	Ukrainian & international culture	Environmental	P P
11		ex	Ukrainiang	ex	ex	ex	ex	its	its	protest.civil def.	R R
12		Principles of law		Physics		Psychology & pedagogics		The fundamat of ecology	Safety human activity	Psychology of managem.	E A
13		its		its		its		its	its	its	- K
14						Students are allowed to choose:					D T
15						MILITARY TRAINING or PEDAGOGICAL COURSES					I I
16		Foreign language				1.Methods of teaching of mathematics and informatics at schools		2.Pedagogical training		Business foreign language	P C
17		its				ex		ex		its	L E
18						Differential equations		Equations of mathematical physics		Control theory	O
19						its		ex		ex	A
20		Discrete mathematics				Functionality analysis		Stochastic processes		Math.modellirny and system analysis	System and methods of decision making
21		its				ex		its		ex	D
22						Mathematical analysis		Data analysis		Modeling of econom.,eool, social sys.,its	Mathematical economics
23						ex		ex		ex	I
24		Algebra & geometry				Theory of probability and mathematical statistics		Numerical methods		Methods of optimization	Mathematical economics
25		ex				ex		ex		ex	L
26						ex		ex		ex	O
27		Algebra & geometry				Computer graphics		Numerical methods		Special programming languages	Multimedia technologies or Distributed data bases
28		its				ex		ex		ex	M
29						C programming		Data bases and inform. system		Computer networks	Insurance business
30		ex				ex		ex		ex	M
31						International econ.relation		System programming		Computer technologies in banking	I
32						ex		ex		ex	M
33						ex		ex		ex	N
34						ex		ex		ex	G
35						ex		ex		ex	
36		Base software of PC				Theory of accounting		Economic analysis		Financial economics	Computer technologies in banking
37		ex				ex		ex		ex	
38		The principles of economical theories				ex		ex		ex	
39		its				ex		ex		ex	
40						ex		ex		ex	
41						ex		ex		ex	
35						ex		ex		ex	

Fig. 7. Graphic presentation of a curriculum

№ п/п		Назва дисципліни і видів навчальної роботи студента		План навчального процесу													Кількість годин на тиждень по курсах, семестрах і чвертях																																		
				Обсяг роботи студ. (в годинах)													1 курс							2 курс							3 курс							4 курс													
				Індивідуальні завдання			Підсумковий контроль			Шкільні дисципліни			Курсові роботи (проекти)			Кількість кредитів ECTS			1 семестр		2 семестр		3 семестр		4 семестр		5 семестр		6 семестр		7 семестр		8 семестр																		
аудиторні заняття	всього	лекції	практичні (семінари)	лабораторні роботи	індивідуальні заняття	всього	лекції	практичні (семінари)	лабораторні роботи	всього	лекції	практичні (семінари)	лабораторні роботи	1 чверть / 2 чверть	2 чверть / 4 чверть	3 чверть / 4 чверть	4 чверть / 6 чверть	5 чверть / 6 чверть	7 чверть / 8 чверть	9 чверть / 10 чверть	1 чверть / 12 чверть	2 чверть / 12 чверть	3 чверть / 12 чверть	4 чверть / 12 чверть	5 чверть / 16 чверть	6 чверть / 16 чверть	7 чверть / 16 чверть	8 чверть / 16 чверть																							
1. Цикл загальноуніверситетських нормативних навчальних дисциплін																																																			
1	1	Історія України	ІУ	3	3	р	30	2,0	81	36	36	0	0	45																																					
1	2	Українська мова (за професійним спрямуванням)	УМ	1	1	нр	7	1,5	54	36	0	36	0	18	0	4	0	2,0																																	
1	3	Культурологія	КУ	1	8	р	39	1,5	54	36	36	0	0	18																																					
1	4	Філософія (філософія, логіка, естетика, релігієзнавство)	ФІ	6	5	р	8	5,0	189	72	36	36	0	117																																					
1	5	Психологія	ПС	7	7	нр	29	2,0	81	36	18	18	0	45																																					
1	6	Іноземна мова (за професійним спрямуванням)	ІМ	2	8	3,5								126	0	2	0	1,0	0	2	0	1,0	0	2	0	1,0	0	2	0	1,0	0	2	0	1,0	0	2	0	1,0													
1	7	Економічна теорія	КТЕ	2	1	нр	8	3,0	108	54	36	18	0	54	2	0	2,0																																		
1	8	Фізичне виховання*	БФВ	2,4	8,8					8,0	288	288	0	288	0	2	0	2*	0	2	0	2*	0	2	0	2*	0	2	0	2*	0	2	0	2*	0	2	0	2*													
1	9	Політологія	ПО	1	13	р	8	2,0	81	36	36	0	0	45																																					
1	10	Правознавство	ЮП	1	10	нр	18	2,0	81	36	36	0	0	45																																					
1	11	Соціологія	ДС	1	12	нр	39	2,0	81	36	36	0	0	45																																					
1	12	Екологія	БГ	1	12					1,5	54	36	36	0	18																																				
1	13	Безпека життєдіяльності	БЖ	1	8					1,5	54	18	18	0	36																																				
1	14	Охорона праці	БЖ	1	13					1,5	54	18	18	0	36																																				
Усього за дисциплінами першого циклу													42,5	1530	882	342	540	0	648	2	6	0	5,0	4	4	0	6,0	2	6	0	7,5	2	6	0	9,0	0	2	0	0,0	0	2	0	0,0	6	2	0	9,0	0	2	0	0,0
2. Цикл загальних нормативних навчальних дисциплін одного напрямку підготовки																																																			
2	1	Математичний аналіз	ТМ	1	13,6					1,2	2нр	7,17	15,0	540	288	144	144	0	252	4	2	0	3,0	4	2	0	2,0	2	2	0	9,0																				
2	2	Алгебра та геометрія	ТМ	1	4					3,4	2нр	30,39	10,5	378	144	72	72	0	234	2	2	0	6,5	2	2	0	6,5																								
2	3	Дискретна математика	ПМ	2						1,2	2нр	8,19	6,0	216	90	54	36	0	126	2	2	0	7,0																												
2	4	Диференціальні рівняння	ТМ	1	11					10	нр	18	4,5	162	108	54	54	0	54	4	2	0	7,0																												

Fig. 8. Table presentation of the curriculum

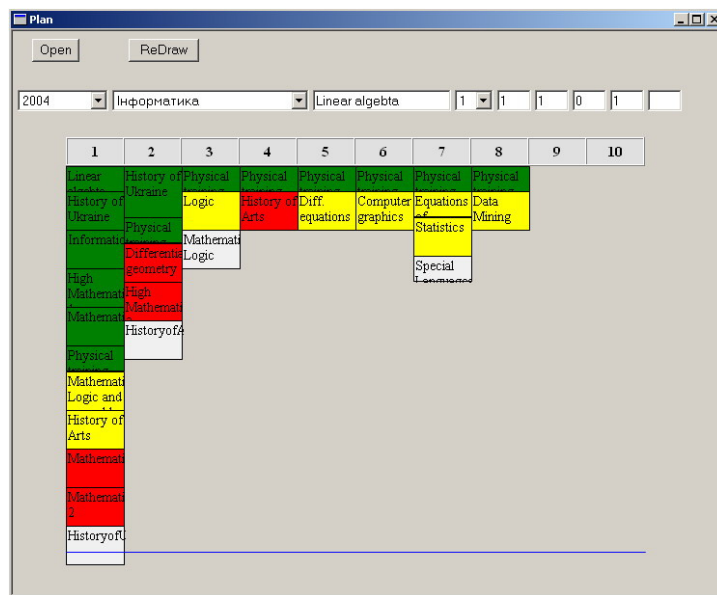


Fig. 9. Process of curriculum formation

Now a certain sort of algorithm which distributes the given disciplines to semester is used. Besides, the priority values which are determined for every field especially are selected from the database. The distribution should automatically take place on all 8 semesters, so that a certain current value per semester is not crossed.

The correction of the curriculum in the graphic representation takes place through clicking the respective discipline. As a result this is displayed in the web form, with the actual settings for the selected lecture, about the table. Now there is a possibility to determine the duration of the discipline, as

well as the semester in which this should take place as well as to adapt the remaining information (lab hours, training period, exam kind), or to change.

Thus, the semester hours number is the sum of all lecture hours in a semester which is identical for all semesters shown with a horizontal line under the table, so that one also has an optical border by which one can orientate himself by the timetable production or correction in the graphic representation.

For the graphic representation we also plan the introduction of two buttons in every discipline, with which one can extend or shorten the duration of the respective lecture. In addition, an announcement with the actual semester hours number is planned, so that the user will still have another further control, as well as a better overview by the planning of the hours. Perhaps, an error message will occur if too much or not enough semester hours are chosen and ask the user to change this setting.

A previously planned function with the help of which one could adapt the hourly number of the respective discipline by the drawing with the mouse, was rejected for the time being, because from our viewpoint the introduction of the buttons would be completely sufficient and also a simpler and better alternative for this.

Of course, one has a possibility to return to any mode at any time. With the help of the present inputs, as well as the data being in the database the final curriculum is provided. On the basis of the curriculum the working plan for the current year is automatically provided. This is not changeable. Should the working plan be changed, nevertheless, this can happen only about the change of the curriculum.

5. Searching and statistical subsystems

An important additional part of UEMS is the information, searching and statistical module. Examples of module interfaces are represented in Fig. 10, 11. Let us have a look at the group of the statistic modules, or the modules of statistics, as they are also called. The purpose of every statistic module is getting some statistic data that suit a set of given conditions in a definite output format.

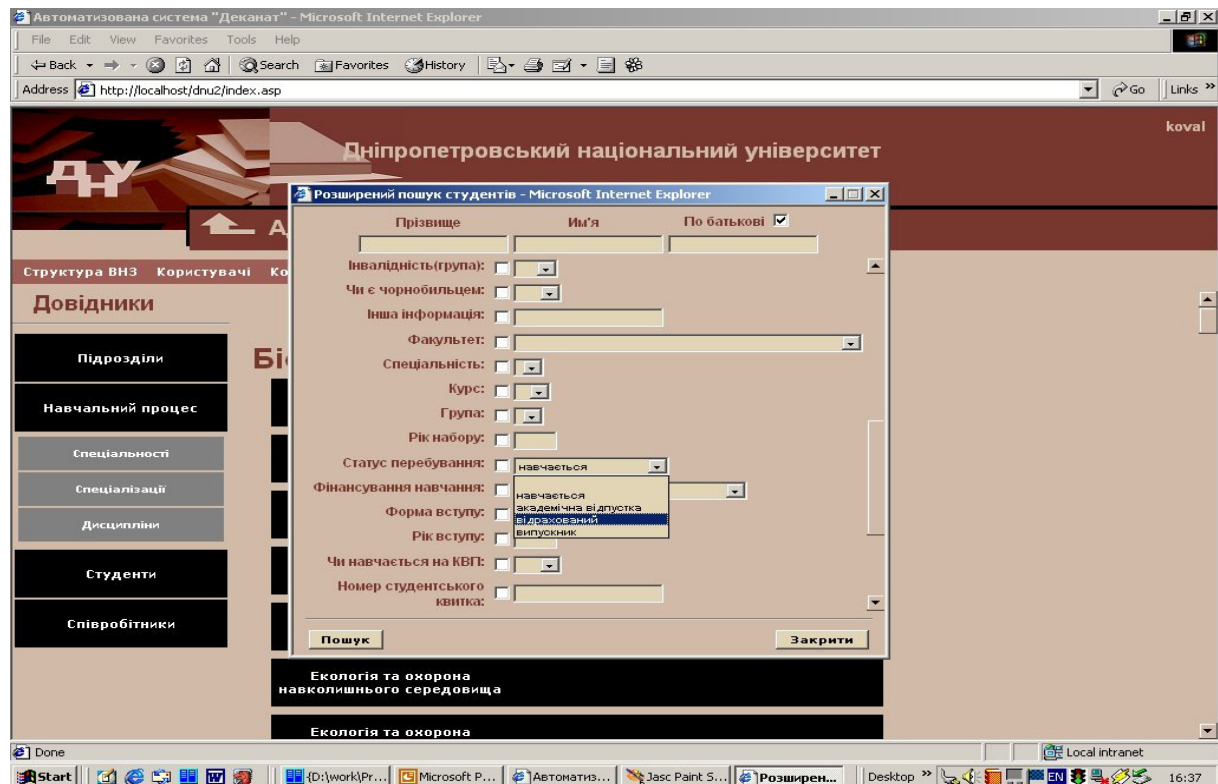


Fig. 10. The interface of the information and searching system

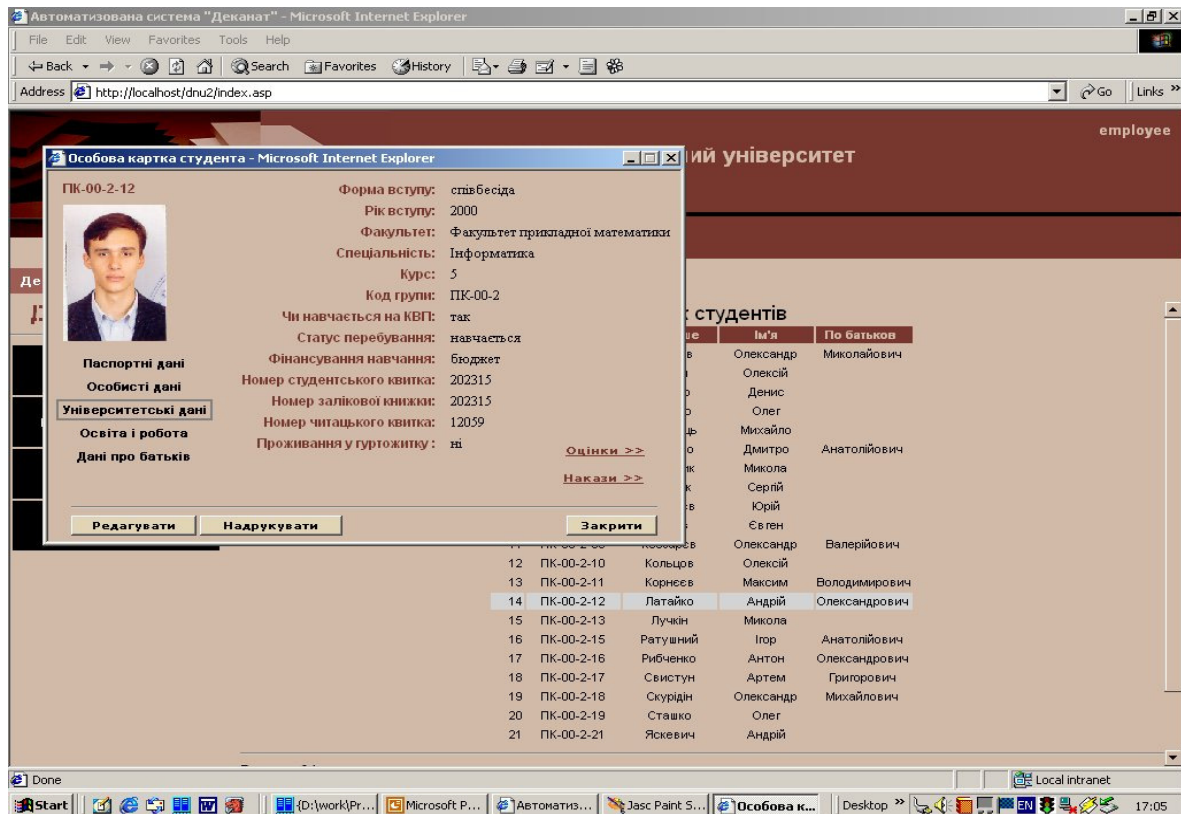


Fig. 11. Personal data of a student (the information and searching system)

The parameters of request to a module for getting responding data, structure and format of output data are determined by a concrete problem. That can be, for example, getting the quantity of every kind of mark got inside the groups of current faculty within pointed examinations, or calculating average progress of that or another group of current faculty within the whole period of studying, or, finally, getting all the marks and total average progress of every student of a stated group within a stated studying period.

As a rule, the format of output data is based on an official form that is used in the work of the dean's office. But in some cases it is necessary to get some additional data in a comfortable format. Most of such cases are also provided: a user is given a possibility to select values of as many parameters as possible according to the data security requirement and the way of formulating the task of the module.

The resulting data can be useful information in making decisions. It means that as a result of further data processing a definite summary can be made and appropriate measures can be taken relatively to the analysed aspect of the work of the dean's office.

The main purpose of 'Statistics of teachers' module is providing the possibility to get full information about marks put by a definite teacher to students of a definite group for a definite discipline within stated examinations. In case of successful authorization in the main module and successful definition the current faculty by this module, there appears the result of starting call of this module in the working frame. The content of the working frame is shown in Fig. 12.

The user is to point wanted examinations with selecting the appropriate academic year and kind of examinations (winter or summer) in the form. In addition to that, the user is given the possibility to choose the categories he wishes to output the statistics in. By default all the checkboxes inside the 'Statistics' field-set are checked, so the statistics in this case will be displayed in all of the three available categories: teachers, groups and disciplines. Pushing the 'Clear form' button will lead to clearing the fields for academic year and examinations kind and setting the default values for the checkboxes.

The screenshot shows a web interface for a statistical module. At the top, there are two main sections: 'Дані про сесію' (Data about the session) and 'Статистика' (Statistics). The 'Дані про сесію' section contains two dropdown menus: 'Учбовий рік' (Academic year) and 'Сесія:' (Session). The 'Статистика' section has three checked checkboxes: 'По викладачах' (By teachers), 'По групах' (By groups), and 'По дисциплінах' (By disciplines). To the right of these sections are two buttons: 'Відобразити статистику' (Display statistics) and 'Очистити форму' (Clear form). Below the filters is a table with the following structure:

№ з/п	Викладач	Група	Дисципліна	Кількість								
				"5"	"4"	"3"	"2"	не з'яв.'	з пов. пр.	не доп.		

Fig. 12. The working frame of the statistical module

If all the required parameters (such as academic year, kind of examinations and at least one category) are stated, pushing the 'Display statistics' button will be followed by making the appropriate request to the server program. Otherwise there will be displayed a message about the absence of required data in the form. If there is at least one record regarding the mentioned examinations at current faculty found in the server database, the responding statistics will be calculated in the mentioned categories and the result will be sent to the client. After that the table in the lower frame will be filled with resulting statistics. Also there will appear the 'Print' button at the bottom of the table – when the button is clicked, the table is printed. If there are no appropriate records in the database, the appropriate message will appear and the content of the lower frame will include only the full caption of the resulting table.

The structure of the caption of the table is quite simple: the leftmost column is the ordinal number of a record, then there are columns for the values of mentioned categories, the further four columns are for the quantities of the kinds of marks (those are fives, fours, threes and twos), and, finally, there are three columns for the quantities of absences without valid reasons, absences for valid reasons and quantities of students who were not allowed to take examinations.

It is worth saying that indicating the categories has an influence just upon the output format: in fact the work is performed with the same data, so the sums of the values in every column for quantities of marks for the same examinations are constant and do not depend on selected categories.

The resulting data can be used to compare progress of groups. In addition to that, basing on statistics of teachers and groups and considering statistics only of groups' user can assess objectivity of that or another teacher. For instance, if a group is totally weaker according to the statistics of groups but according to the results of the examinations that group gets on average higher marks for disciplines taught by some other teacher, than a stronger group, there can be raised a question about the objectivity of that teacher.

Basing on statistics of disciplines we can see which disciplines are studied better by students. This information can be useful when some curricula or timetables are being created. Statistics of groups and disciplines can be utilized in assigning students to special directions.

Results, computed by the program, cannot be the only basis for taking a decision; however, they can be a useful additional tool. Other means and information should be considered additionally to the suggested statistics, and only after a full detailed analysis of all the obtained results the decision can be taken.

Now 'Statistics of students' module is being developed. This module is to simplify getting a list of students who are going to get a diploma with honours. Usually this process is accompanied with raising a lot of examination lists, looking through them for every student and checking whether their marks let them get a diploma with honours. The check includes calculating the average progress, a quantity of "threes" and re-examinations of every student. Performing that task manually by one man requires a high level of concentration and huge amount of time. In addition to that, the probability of

making a mistake is quite high. After considering all those arguments, it appears very useful and necessary to create a module that will fully solve the problem.

The module is to provide the possibility to show marks in all the disciplines and calculate average progress of every student of a definite group or entire faculty. Also there must be provided a possibility to show the data just of those students who can still obtain a diploma with honours by showing the examinations that must be re-taken, and students who cannot obtain a diploma with honours in any case by showing marks which do not meet the requirements for obtaining a diploma with honours.

Both modules are created on the basis of ASP (Active Server Pages) technology. At the beginning the user makes a request to the server so as to get some ASP-page; if provided, the user sets some parameters in the forms on the page. When the server receives the request, it starts interpreting server scripts built in the code of the ASP-page. Those server scripts are mixed with a ready HTML-code. After performing, the scripts are replaced with white symbols and a ready HTML-page is got. Then the processed page is sent back to the client's browser and the user gets the result of processing the ASP-page. All of that is meant to provide the possibility to create Web-pages with dynamic content. The content of the page can depend on data that are taken from the database located at the server when the server scripts are being interpreted and that are processed according to the parameters set in the forms.

In 'Statistics of teachers, groups and disciplines' module the identification parameters of examinations are passed as a set of parameters and the data about the examinations at the current faculty are received from the database located at the server. The database is a MS Access database and work with it is based on ADO technology. Selection of required data is performed by appropriate SQL requests. In ASP technology the default script language is VBScript but for creating the two modules JavaScript was chosen as both server and client script language.

UEMS is a complex rapidly developing system, so we described Now UEMS system is being converted into .NET platform. That includes using ASP .NET as a base server technology and writing all the server programs in compiled programming languages with strict type checking (such as C# or VB .NET). According to the specifications of .NET platform that is to increase the total speed and security level of the system.

Conclusion

only several of its aspects. Thanks to the Tempus programme it was possible to intensify the work and to collect very important EU experience. At the same time the project even during the Tempus project time has demonstrated its sustainability, and we are sure that its development will be still continued after the project ending.

Поступила в редакцию 15 октября 2006 г.