

U P O R A B N A

I N F O R M A T I K A

2004 STEVILKA 4 OKT/NOV/DEC LETNIK XII



Testni centri ECDL

ECDL (European Computer Driving License), ki ga v Sloveniji imenujemo evropsko računalniško spričevalo, je standardni program usposabljanja uporabnikov, ki da zaposlenim potrebno znanje za delo s standardnimi računalniškimi programi na informatiziranem delovnem mestu, delodajalcem pa pomeni dokazilo o usposobljenosti. V Evropi je za uvajanje, usposabljanje in nadzor izvajanja ECDL pooblaščen ustanova ECDL Foundation, v Sloveniji pa je kot član CEPIS (Council of European Professional Informatics Societies) to pravico pridobilo Slovensko društvo INFORMATIKA. V državah Evropske unije so pri uvajanju ECDL močno angažirane srednje in visoke šole, aktivni pa so tudi različni vladni resorji. Posebej pomembno je, da velja spričevalo v več kot osemdesetih državah, ki so vključene v program ECDL. Doslej je bilo v svetu izdanih že več kot tri milijone indeksov, v Sloveniji okoli 1700 in podeljenih okoli tisoč spričeval. Za testne centre ECDL so se v Sloveniji usposobile organizacije, katerih logotipi so natisnjeni na tej strani.

ISER



MELES - ICES



EUROCOM
Simply logical



ISA
INFORMATIJSKE
TEHNOLOGIJE



KOPA



Micro Team

Much

spin



U P O R A B N A I N F O R M A T I K A

2004 ŠTEVILKA 4 OKT/NOV/DEC LETNIK XII ISSN 1318-1882

Uvodnik

Razprave

Markus Alekxy, Ingo Bayer, Tilo Dickopp, Axel Korthaus:

IB-CUE – Theoretical Concept and Implementation Aspects for a Software-based Information System for Innovative Budgeting in a Competitive University Environment

177

Pascal Aubry, Vincent Mathieu, Julien Marchal:

ESUP-Portal: open source Single Sign-On with CAS (Central Authentication Service)

187

Jorma Kajava, Rauno Varonen:

Information Security e-Learning – from Practice to Theory

199

Milan Ojsteršek, Dejan Dinevski, Tomaž Klojčnik:

Portal Univerze v Mariboru

206

Jaka Lindič, Aleš Popovič, Jurij Jaklič, Mojca Indihar Štemberger:

Vpliv spletnih portalov na kakovost visokošolske izobraževalne institucije

218

Lea Bregar, Margerita Zagmajster, Marko Papić:

Portal kot sredstvo za razvoj e-izobraževanja v slovenskem visokem šolstvu

228

Viljan Mahnič, Marko Požnenel:

Uporaba podatkovnih skladišč v univerzitetnem okolju: primer Univerze v Ljubljani

237

Nataša Žabkar, Viljan Mahnič:

Obvladovanje tveganj pri uporabi informacijske tehnologije v univerzitetnem okolju

248

Nove knjige

K razvoju telecentrov v Sloveniji

259

Koledar prireditev

259



ISSN 1318-1882

Ustanovitelj in izdajatelj:

Slovensko društvo INFORMATIKA
Vožarski pot 12
1000 Ljubljana

Predstavniki

Niko Schlamberger

Odgovorni urednik:

Andrej Kovačič

Uredniški odbor:

Marko Bajec, Vesna Bosilj Vukšič, Dušan Caf, Aljoša Domijan, Janez Grad, Jurij Jaklič, Milton Jenkins, Andrej Kovačič, Tomaž Mohorič, Katarina Puc, Vladislav Rajkovič, Heinrich Reinermann, Ivan Rozman, Niko Schlamberger, John Taylor, Ivan Vezočnik, Mirko Vintar, Tatjana Welzer - Družovec

Recenzenti prispevkov za objavo v reviji Uporabna informatika:

Marko Bajec, Tomaž Banovec, Vladimir Batagelj, Marko Bohanec, Vesna Bosilj Vukšič, Dušan Caf, Srečko Devjak, Aljoša Domijan, Tomaž Erjavec, Matjaž Gams, Tomaž Gornik, Janez Grad, Miro Gradišar, Jože Gričar, Jozsef Györkos, Marjan Heričko, Jurij Jaklič, Milton Jenkins, Andrej Kovačič, Iztok Lajovic, Tomaž Mohorič, Katarina Puc, Vladislav Rajkovič, Heinrich Reinermann, Ivan Rozman, Niko Schlamberger, Ivan Vezočnik, Mirko Vintar, Tatjana Welzer - Družovec, Franc Žerdin

Tehnična urednica

Mira Turk Škraba

Oblikovanje

Bons

Prelom

Dušan Weiss, Ada Poklač

Tisk

Prograf

Naklada

700 izvodov

Naslov uredništva

Slovensko društvo INFORMATIKA
Uredništvo revije Uporabna informatika
Vožarski pot 12, 1000 Ljubljana
www.drustvo-informatika.si/posta

Revija izhaja četrtletno. Cena posamezne številke je 4.500 SIT. Letna naročnina za podjetja 17.800 SIT, za vsak nadaljnji izvod 11.900 SIT, za posameznike 5.900 SIT, za študente 2.800 SIT.

Revijo sofinancira Ministrstvo za šolstvo, znanost in šport RS.

Revija Uporabna informatika je od številke 4/VII vključena v mednarodno bazo INSPEC.

Revija Uporabna informatika je pod zaporedno številko 666 vpisana v razvid medijev, ki ga vodi Ministrstvo za kulturo RS.

© Slovensko društvo INFORMATIKA

Navodila avtorjem

Revija Uporabna informatika objavlja izvirne prispevke domačih in tujih avtorjev na znanstveni, strokovni in informativni ravni. Namenjena je najširši strokovni javnosti, zato je zaželeno, da so tudi znanstveni prispevki napisani čim bolj poljudno.

Članke objavljamo praviloma v slovenščini, prispevke tujih avtorjev v angleščini.

Prispevki so obojestransko anonimno recenzirani. Vsak članek za rubriko Razprave mora za objavo prejeti dve pozitivni recenziji. O objavi samostojno odloča uredniški odbor.

Prispevki naj bodo lektorirani, v uredništvu opravljamo samo korekturo. Po presoji se bomo posvetovali z avtorjem in članek tudi lektorirali. Prispevki za rubriko Razprave naj imajo dolžino do 40.000, prispevki za rubrike Rešitve, Poročila do 30.000, Obvestila pa do 8.000 znakov.

Naslovu prispevka naj sledi ime in priimek avtorja, ustanova, kjer je zaposlen, in elektronski naslov. Članek naj ima v začetku do 10 vrstic dolg izvleček v slovenščini in angleščini, v katerem avtor opiše vsebino prispevka, dosežene rezultate raziskave. Abstract se začne s prevodom naslova v angleščino. Članku dodajte kratek avtorjev življenjepis (do 8 vrstic), v katerem poudarite predvsem delovne dosežke.

Pišite v razmaku ene vrstice, brez posebnih ali poudarjenih črk, za ločilom na koncu stavka napravite samo en prazen prostor, ne uporabljajte zamika pri odstavkih.

Revijo tiskamo v črno-beli tehniki s folije, zato barvne slike ali fotografije kot originali niso primerne. Objavljali tudi ne bomo slik zaslonov, razen če niso nujno potrebne za razumevanje besedila. Slike, grafikoni, organizacijske sheme ipd. naj imajo belo podlago. Po možnosti jih pošiljajte posebej, ne v datoteki z besedilom članka. Disketi z besedom priložite izpis na papirju.

Prispevke pošiljajte po elektronski ali navadni pošti na naslov uredništva revije: ui@drustvo-informatika.si, Slovensko društvo INFORMATIKA, Vožarski pot 12, 1000 Ljubljana. Za dodatne informacije se obračajte na tehnično urednico Miro Turk Škraba.

Po odločitvi uredniškega odbora o objavi članka bo avtor prejel pogodbo, s katero bo prenesel vse materialne avtorske pravice na Slovensko društvo INFORMATIKA. Po izidu revije pa bo prejel nakazilo avtorskega honorarja po veljavnem ceniku ali po predlogu odgovornega urednika.

Spoštovane bralke, spoštovani bralci,

tematska številka revije *Uporabna informatika*, ki jo imate pred sabo, je posvečena uporabi informacijskih in komunikacijskih tehnologij v visokem šolstvu. Povod za njeno izdajo je bila 10. konferenca EUNIS (European University Information Systems Association), ki je pod geslom "IT Innovation in a Changing World" potekala na Bledu od 30. junija do 2. julija letos. Konferenco je organizirala Fakulteta za računalništvo in informatiko Univerze v Ljubljani, na njej pa so avtorji iz 29 držav Evrope, Amerike in Avstralije predstavili 96 prispevkov (referatov, kratkih referatov in posterjev), v katerih so prikazali najnovejše dosežke pri uporabi informacijskih in komunikacijskih tehnologij na naslednjih področjih:

- e-izobraževanje,
- sodobni visokošolski informacijski sistemi,
- obvladovanje informacij in zagotavljanje kakovosti,
- e-knjižnice,
- računalniška in komunikacijska infrastruktura.

Da bi približali omenjeno problematiko tudi bralcem revije *Uporabna informatika*, smo zaprosili avtorje nekaterih najzanimivejših prispevkov, da jih pripravijo za objavo v reviji. Objavljene razprave pokrivajo različne vidike rabe informacijskih in komunikacijskih tehnologij v visokem šolstvu, od naprednih univerzitetnih informacijskih sistemov in e-izobraževanja do izgradnje portalov, podatkovnih skladišč in obvladovanja tveganj pri uporabi informacijske tehnologije.

V prvi razpravi z naslovom *IB-CUE - Theoretical Concept and Implementation Aspects for a Software-based Information System for Innovative Budgeting in a Competitive University Environment* je opisana zasnova informacijskega sistema za načrtovanje univerzitetnega proračuna, ki ga razvijajo na univerzi v Mannheimu. Članek je zanimiv tako zaradi modela proračunskega načrtovanja, ki poleg financiranja osnovne dejavnosti spodbuja tudi doseganje optimalnih rezultatov in inovacij, kot zaradi uporabljenega pristopa k realizaciji informacijskega sistema, ki temelji na uporabi UML modelov, izdelavi prototipa in njegovi nadaljnji evoluciji v smeri porazdeljene aplikacije.

V drugi razpravi avtorji Aubry, Mathieu in Marchal opisujejo splošno rešitev za problem enkratne prijave, ki jo uporablja francoski konzorcij ESUP-Portail in ki omogoča, da se izognemo večkratnemu vnašanju gesla za dostop do različnih aplikacij. Rešitev temelji na uporabi odprte kode in zagotavlja integriran dostop do vseh storitev in informacij na univerzah in visokih šolah tako študentom kot zaposlenim.

Tretja razprava sodi na področje e-izobraževanja. Kajava in Varonen opisujeta sistem za računalniško podprto učenje na področju varnosti podatkov, ki ga je razvila Univerza Oulu na Finskem. Opisani so najpomembnejši faktorji, ki so vplivali na načrtovanje vsebine in realizacijo ustrezne računalniške podpore, kot tudi izkušnje z uporabo sistema v manjšem telekomunikacijskem podjetju. Na podlagi teh izkušenj bodo sistem prilagodili za uporabo v univerzitetnem okolju.

Kar tri razprave slovenskih avtorjev obravnavajo problematiko izgradnje portalov. V prispevku z naslovom *Portal Univerze v Mariboru* je predstavljen portal, ki omogoča zaposlenim in študentom dostop do različnih vsebin, ki jih pripravljajo posamezne fakultete in rektorat, dostop do univerzitetnega informacijskega sistema in podporo izobraževanju na daljavo. V drugi razpravi na to temo pa Lindič in sodelavci opisujejo, kako lahko primerno oblikovan spletni portal vpliva na kakovost visokošolskih izobraževalnih institucij, in to pokažejo na primeru portala Ekonomske fakultete Univerze v Ljubljani. To skupino člankov zaključuje razprava z naslovom *Spletni portal kot informacijski vir za spodbujanje razvoja in raziskovanja e-izobraževanja v Sloveniji*, v kateri je podrobneje opisana vloga izobraževalnih portalov in njihov vpliv na nadaljnji razvoj študija na daljavo pri nas.

Razprava *Uporaba podatkovnih skladišč v univerzitetnem okolju: primer Univerze v Ljubljani* opisuje projekt izgradnje enotnega podatkovnega skladišča za področje študijske informatike, ki je za ljubljansko univerzo pomemben zato, ker omogoča integracijo podatkov, ki so se doslej obdelovali ločeno v okviru informacijskih sistemov posameznih članic.

Zadnja razprava posega na področje obvladovanja tveganj pri uporabi informacijske tehnologije, ki je bilo doslej prisotno predvsem v finančnih ustanovah, z vse večjo uporabo informacijskih in komunikacijskih tehnologij v visokem šolstvu pa postaja vedno bolj pomembno tudi v univerzitetnem okolju. Prikazuje pristop k obvladovanju tveganj, ki je primeren za uporabo na univerzi, združuje pa rešitve, ki so predlagane v ogrodju za upravljanje informacijske tehnologije COBIT, angleškemu standardu za obvladovanje tveganj in standardu BS 7799.

Upava, da objavljeni prispevki predstavljajo vso raznolikost uporabe informacijskih in komunikacijskih tehnologij v visokem šolstvu, obenem pa omogočajo primerjavo stanja v Sloveniji s podobnimi projekti v tujini. Za sodelovanje se zahvaljujema avtorjem in recenzentom, še posebej pa odgovornemu uredniku prof. dr. Andreju Kovačiču, ki je z veliko mero razumevanja in naklonjenosti sprejel našo pobudo za pripravo te številke.

Gostujoča urednika
Viljan Mahnič in Boštjan Vilfan

Slovensko društvo INFORMATIKA

zbira na podlagi 53. člena statuta in pravilnika o priznanjih predloge za

PRIZNANJA SLOVENSKEGA DRUŠTVA INFORMATIKA

1. Priznanje se lahko podeli posamezniku ali pravni osebi za:
 - dosežke na področju uporabne in znanstvene informatike ter vidne prispevke na področju razvoja informacijske družbe in razvoja novih načinov in tehnologij dela na področju informatike
 - dolgoletno uspešno delo v društvu ali v drugih društvih, ki so sodelovala z društvom pri programskih vprašanjih
 - razvoj mednarodnega sodelovanja in izmenjavo dosežkov na tem področju
 - izjemne dosežke na področju razvoja konceptov, programskih orodij, naprav in tehnologij v zvezi z informatiko
 - uspešno sodelovanje z društvom
 - publicistično delo na področju informatike in informacijske družbe in
 - izjemne dosežke na področjih, ki zadevajo vprašanja informatike.
2. Predlog mora vsebovati:
 - podatke o prejemniku priznanja
 - opis dosežka
 - predlagano priznanje
 - dokazila o dosežku
 - podatke o predlagatelju.

Podrobni pogoji so navedeni v pravilniku na naslovu <http://www.drustvo-informatika.si>

Predloge pošljite do vključno 30. januarja 2005 na naslov:

Slovensko društvo INFORMATIKA
Vožarski pot 12
1000 Ljubljana

z oznako "PRIZNANJA 2005"

Predloge bo v skladu s pravilnikom obravnavala komisija za priznanja in jih s svojim mnenjem posredovala izvršnemu odboru društva. Priznanja bodo javno podeljena na otvoritvi posvetovanja Dnevi slovenske informatike aprila 2005.

DNEVI SLOVENSKE INFORMATIKE 2005

Portorož, 13.–15. aprila 2005

Popravek

Obveščamo vas, da je bil v tretji številki revije Uporabna informatika pomotoma objavljen napačen datum posvetovanja Dnevi slovenske informatike 2005.

Posvetovanje bo potekalo **od 13. do 15. aprila 2005** v Kongresnem centru Grand Hotela Bernardin v Portorožu.

Več informacij o posvetovanju najdete na spletni strani posvetovanja DSI 2005: www.dsi-2005.org,
lahko pa nam tudi pišete na naslov: dsi@drustvo-informatika.si.

Vabimo vas, da se nam pridružite!

Slovensko društvo INFORMATIKA

IB-CUE – Theoretical Concept and Implementation Aspects for a Software-based Information System for Innovative Budgeting in a Competitive University Environment

Markus Aleksy¹, Ingo Bayer², Tilo Dickopp¹, Axel Korthaus¹
 University of Mannheim, Schloss, L 5.5, 68131 Mannheim, Germany
¹Department of Information Systems, ²School of Business Administration, Dean's Office
 {aleksy|dickopp|korthaus}@wifo3.uni-mannheim.de
 bayer@bwl.uni-mannheim.de

Abstract

In our paper, we first describe a new conceptual approach to budget planning and allocation on the schools' level, which promotes autonomy and goal-orientation and provides incentives in order to optimize the performance and innovation outputs. Subsequently, we focus on the design and implementation of a software application system, which realizes computer support for this approach. An object-oriented UML model and the discussion of selected design decisions provide background information about a stand-alone prototype implementation. We provide an analysis of different aspects concerning a layered architecture for the final software product and the evolution of the prototype to become a distributed application.

Keywords: university budgeting information system, distributed object-oriented systems, object-oriented analysis and design.

Povzetek

IB-CUE: teoretična podlaga in različni vidiki realizacije informacijskega sistema za podporo inovativnega proračunskega načrtovanja v tekmovalnem univerzitetnem okolju

V članku najprej opišemo nov pristop k proračunskemu načrtovanju (načrtovanju denarnih pretokov) ter razdelitvi sredstev na ravni posameznih šol (fakultet). Pristop podpira avtonomijo in ciljno usmerjenost ter vsebuje spodbude za doseganje optimalnih rezultatov ter inovacij. Nato se osredotočimo na programsko opremo za računalniško podporo takega sistema. Prototip informacijskega sistema je realiziran kot izoliran (stand-alone) sistem, njegove lastnosti pa opišemo na podlagi objektno usmerjenega UML modela ter posameznih razvojnih odločitev. Sledi analiza različnih vidikov razslojene arhitekture končnega programskega produkta ter evolucija prototipa v smeri porazdeljene aplikacije.

Ključne besede: informacijski sistem za načrtovanje proračuna univerze, porazdeljeni objektno usmerjeni sistemi, objektno usmerjena analiza in načrtovanje.

1 Introduction

Today, state-owned German universities are facing major challenges due to dramatic changes in their environment. The devastating status of public finances as well as intensifying competition in all fields of education forces them to rethink their strategies and to adapt swiftly to these challenges, because only those who are able to "ride before the wave" of forced reforms imposed by the government have a chance to define their own future ([18], [19]). Therefore, they have to assure that their declining financial resources are not only allocated in an effi-

cient way, but also in a way that advances the achievement of the university's goals as a whole. In our paper, we sketch the core ideas of an innovative budgeting model on the level of a school inside a university, because this is actually the place where the university's resources are transformed into teaching and research as the two major outputs of the university [2]. An in-depth discussion of design and implementation considerations for a software application system implementing the new approach follows in the second part of the paper.

2 Concepts for an innovative budgeting system

The budgeting system is based on an internal school organization, in which the chairs of the school belong to different departments, as, for example, Department of Accounting, Department of Management, and so on. The departments are responsible for teaching and research in their field of specialization, and the heads of the departments negotiate contracts with the dean's office about the output that has to be delivered to the school. There is a mutual understanding that every department has to carry a part of the basic burden the school has to carry for the university as a whole. In addition to this basic output, every department is also expected to strive for innovations in research and teaching, which will help the school to excel as a competitor for public and private funds [7].

Thus, the budgeting model is based on the assumption that it should be useful in a threefold way. Firstly, it should enable the faculty members to carry out the basic workload the school is obliged to deliver in general. Secondly, it should reward those faculty members who perform in a way that enables the school to reach its more ambitious goals, and, last not least, it should boost innovation inside the school. Along these different goals, the budget model consists of three major pillars, which are:

- a Basic Budget,
- a Performance Budget,
- an Innovation Budget (cf. Figure 1).

The *Basic Budget* enables the school to meet basic operational requirements and provides resources for the general infrastructure, staff rooms etc. It can also

reflect different performance levels of departments and be used to counteract the cementation of these differences due to different initial resources.

The *Performance Budget* rewards the faculty in accordance with the percentage of the overall workload of the school the individual faculty member or the individual department carries. This budget is distributed based on performance indicators that reflect the performance indicators on the university level, which are directly responsible for the budget volume of the school as a whole. In addition, performance budget distribution can also account for indicators that reflect the strategic goal settings of the school, so that goal-orientation is strengthened within the whole school.

The *Innovation Budget* has an "incubator function" for the school. It is used to stimulate innovative projects, which are in congruence with the school's strategy, like research projects, new teaching models etc.

To be as efficient as possible, administrative tasks such as budget planning activities have to be supported by adequate software-based information systems. In the remainder of the paper, we outline some basic aspects of the development of a software application system that can support the new budgeting approach explained above. IB-CUE ("Innovative Budgeting in a Competitive University Environment") is the acronym we chose as a name for this information system, which is currently being developed at the University of Mannheim. However, a working prototype of the application already exists.

Since the focus of this paper is to describe this prototype and to present further considerations for implementing a fully-fledged software solution to support our theoretical approach, we do not go into more detail about the budgeting methodology here. Readers who are interested to learn more about the business administrative concepts behind our approach are referred to [1].

3 Considerations for the development of the IB-CUE software application

When we planned the development process of IB-CUE, we identified two core milestones that could help to enforce an incremental approach that supports separation of concerns and step-wise refinement:

1. implementation of a stand-alone prototype providing the business logic, and
2. migration to a distributed software system.

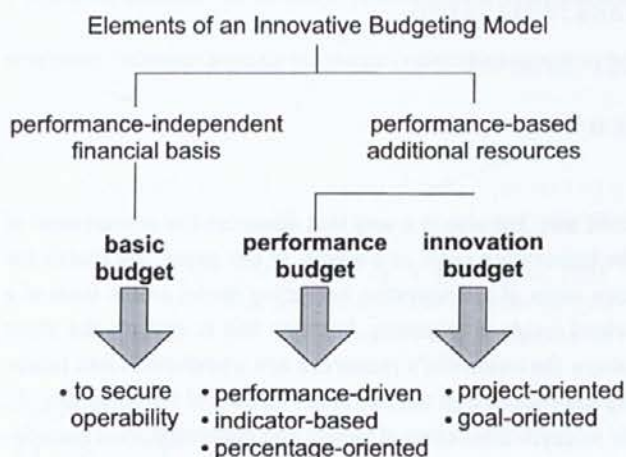


Figure 1: Elements of an innovative budgeting model

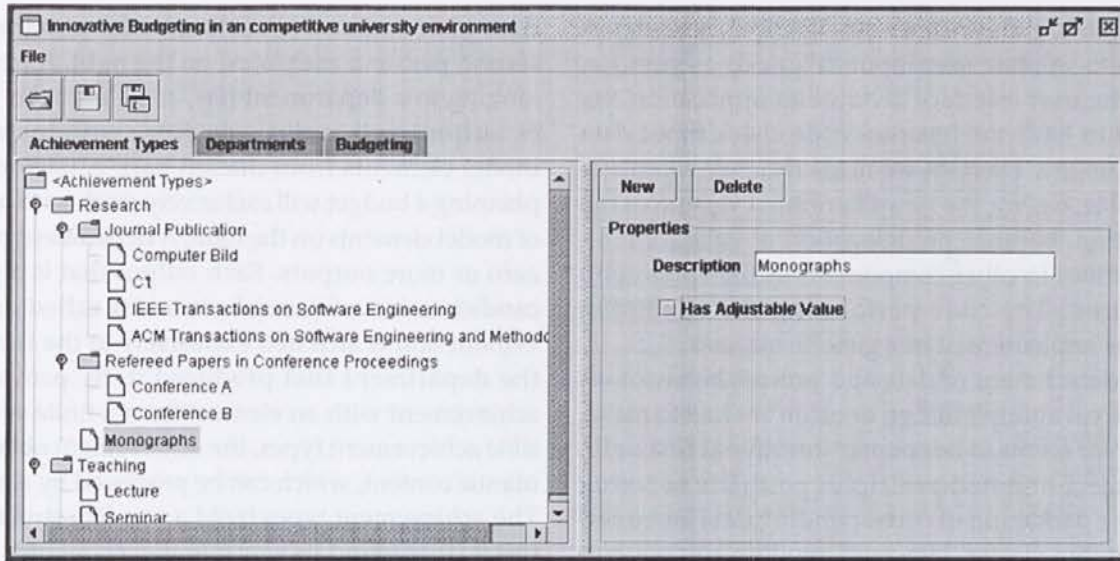


Figure 2: Screenshot of the prototypical IB-CUE software application

Starting with a prototypical implementation that only focused on the business logic of the budgeting application and deferred technical considerations until later had the big advantage, that many of the complex questions and design decisions of software development could be disregarded at first. By initially focusing on the functional aspects only, it became easier to provide a correct mapping of the requirements (defined by the concepts of the budgeting model) to the program code implementing the business logic. However, when designing the prototype, we kept in mind to encapsulate the business logic using clearly defined interfaces in order to allow for a layered approach during the evolution of the system. This software engineering method is called “Design by Contract” [10]. Thus, it will be possible to reuse the core business logic together with different middleware technologies for distributed communication or different clients for user interaction, for example. As mentioned before, this first step was already completed, and a Java-based prototypical implementation of the budgeting approach is available since December 2003. Figure 2 shows a screenshot of the prototype’s GUI.

A core task of the second phase of the development process will be the consideration of distribution aspects for the application. Since a school and its departments have a distributed organization structure, and faculty members should be able to enter their performance achievements into the system by themselves, it is mandatory that all people who are in-

involved in the budgeting process must be able to work with the application simultaneously, without having to leave their offices. This requirement, which is crucial for user acceptance of the software, cannot be met by a monolithic stand-alone version of the application. For this reason, a suitable middleware technology has to be selected that provides the technical foundation for the collaboration of distributed application components.

4 Design and Implementation Aspects of the IB-CUE Prototype

In order to capture the functional requirements of the IB-CUE prototype, we performed object-oriented analysis and design activities using the Unified Modeling Language (UML) ([12], [13]). The resulting UML models served as a foundation for the implementation of the prototype in Java. During analysis and design, we had to make several decisions with respect to potential architectural options. For example, we attached great importance to the encapsulation of the functional business logic code by applying a set of design patterns and best practices [6], such as the “façade pattern” or “single point of access” paradigm. This modularization facilitates the reuse of the business logic and the distribution of the application in the next evolutionary step.

Another design decision referred to the question of where user input data is to be checked for correctness with respect to the business rules, before it is fed into

the budgeting algorithms etc. While it is common practice to validate user input as quickly as possible, i.e., in the user interface layer of an application, we decided to have the business code check input data for plausibility. Thereby, we made the code completely reusable, e.g. for the new distributed version of the application, because no delegation of validation responsibilities to other components of the application is required. The code performing the validation checks is implemented in a generic manner.

The detachment of data and process behavior represents yet another design decision we had to make. This choice seems to be counter-intuitive at first sight, since object-oriented principles postulate cohesion and close packaging of corresponding data and operations within classes. However, modern component-based approaches like Enterprise JavaBeans [5] etc. with their session-oriented and entity-oriented components recognize the benefits of separating data and behavior on a higher level of abstraction. Common design patterns such as the “value object pattern” or the “transfer object pattern” also promote the factoring out of data-centric classes. By separating these data classes, which are used by the business logic and also by the user interface (or the network layer in a future version of IB-CUE), we guarantee that new data format requirements resulting from new application components do not affect the code implementing the core business processes and rules of the system. E.g., since varying marshaling capabilities of different middleware technologies might result in different data type requirements, our approach can minimize the impact of switching the middleware technology.

In the remainder of this section, we would like to provide a detailed description of the design of our prototype, which was implemented in Java. Since the business logic was proven to be correct, other layers of the planned software architecture for the final software product will be implemented using the AspectJ technology [9]. For an overview of the architectural layers required for this system see section 5.

The first prototype mainly realizes the business logic layer, which will be reused in the implementation of the final software product. Therefore, we should have a detailed look at how this layer was designed. The business logic model consists of two parts, as can be seen in Figure 3: Classes that represent departments and their output make up the first part and are placed on the left side of the figure. Classes that deal

with planning and allocating budgets make up the second part and are placed on the right. A person belonging to a department (i.e., someone who produces output) will exclusively work with instances of model elements from the left part. A person who is planning a budget will exclusively work with instances of model elements on the right. A department produces zero or more outputs. Each output that is a possible candidate for a financial reward is called an achievement and is attached exclusively to the instance of the department that produced it. By providing an achievement with an element from a finite set of possible achievement types, the achievement obtains a semantic content, which can be processed by a machine. The achievement types build a tree-like structure, but this feature found its way into the model only for convenience considerations and is not mandatory. The well-known “Composite” design pattern has been used to model this structure [6]. As the researcher is responsible for determining the type of an achievement, the instances of the AchievementType subclasses should be named in accordance with concepts that he can understand, e.g. “publication in journal XY”, and not “publication in an A class journal”.

The budgets are divided into different groups along three dimensions:

The “assignment dimension”

Budgets may or may not be assigned to a (not empty) set of departments. Unassigned budgets function as aggregate budgets and, therefore, they must have children, which either are assigned to departments or have children themselves. This is another application of the aforementioned “Composite” design pattern [6]. In a correct model, every budget has to be attributable to one or more departments either directly or indirectly through its child budgets. Furthermore, the planner can assign budgets to departments either manually, or they are assigned to a department automatically, because the department fulfils certain goals, i.e., it has achievements of certain types, which trigger the assignment of this budget. Typically, basic and innovation budgets will be assigned by hand, whereas performance budgets will be assigned automatically according to generally defined achievement goals.

The “amount type dimension”

Each budget has an associated rule that is used to calculate the amount for this budget. It should be noticed

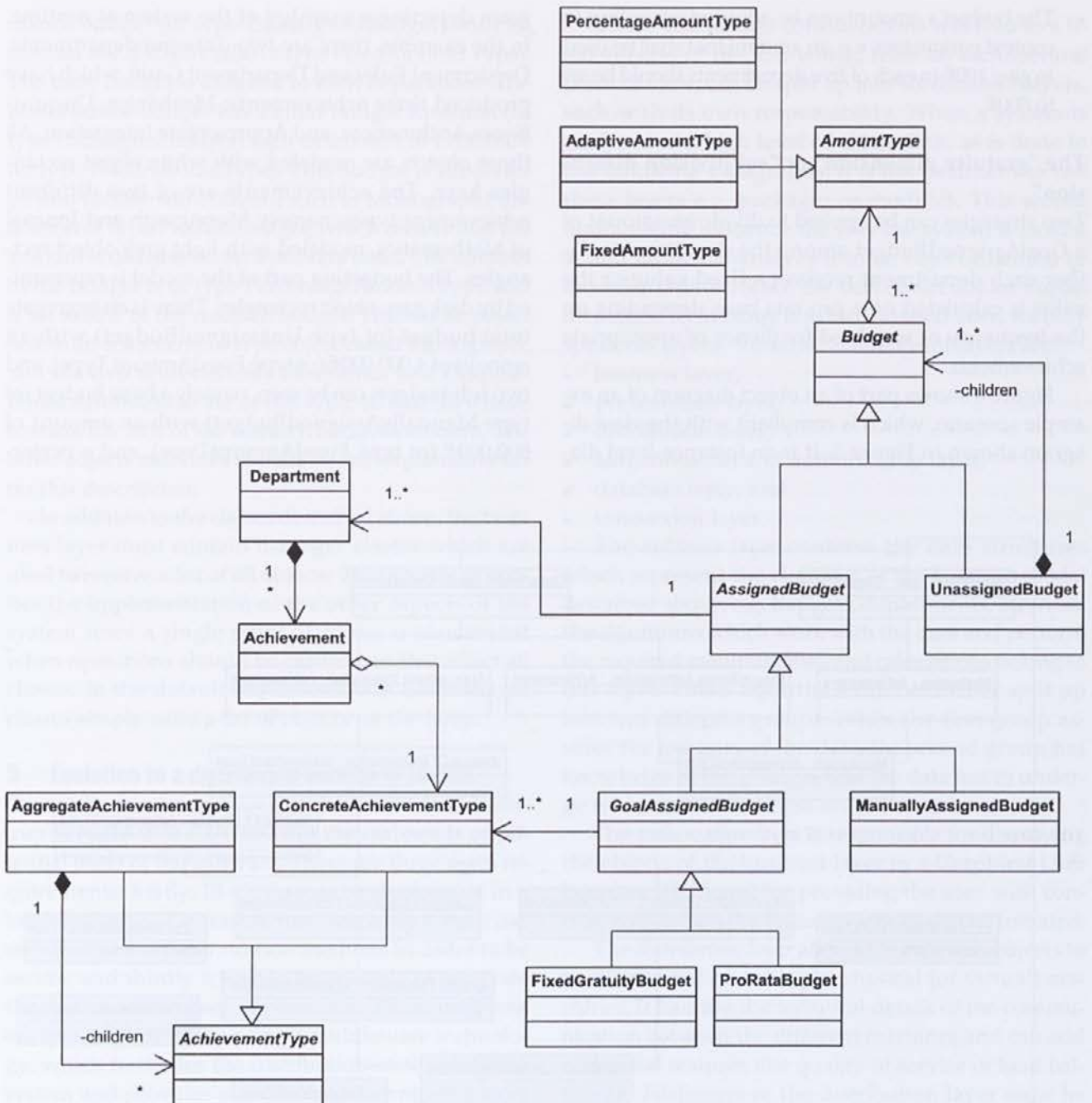


Figure 3: UML class diagram of the budgeting system problem domain

that more than one department might share the amount of a budget, so that an adequate partitioning scheme must determine the amount granted to each department. There are three kinds of rules which can be used to achieve this:

- The budget's amount can be a fixed value. The root budget, which is the aggregate of all budgets, may only be set based on this kind of rule.
- The budget's amount can be a fraction of the amount of its parent budget, specified in percent.

- The budget's amount can be adjusted according to context parameters, e.g. an amount that shall be used to give 100€ to each of five departments should be set to 500€.

The “gratuity dimension” or “subdivision dimension”

Two strategies can be applied to divide an amount of a GoalAssignedBudget among the departments. Either each department receives a fixed value or the value is calculated on a pro rata basis depending on the frequency or weighted frequency of appropriate achievements.

Figure 4 shows part of an object diagram of an example scenario, which is compliant with the class diagram shown in Figure 3. It is an instance-level dia-

gram depicting a snapshot of the system at runtime. In the example, there are two different departments, Department Euler and Department Gauß, which have produced three achievements: *Mechanica*, *Disquisitiones Arithmeticae*, and *Approximate Integration*. All those objects are modeled with white object rectangles here. The achievements are of two different achievement types, namely *Monograph* and *Journal of Mathematics*, modeled with light grey object rectangles. The budgeting part of the model is represented by dark grey object rectangles. There is an aggregate total budget (of type *UnassignedBudget*) with an amount of 4.000.000€ (of type *FixedAmountType*), and two sub-budgets can be seen, namely a basic budget (of type *ManuallyAssignedBudget*) with an amount of 500.000€ (of type *FixedAmountType*), and a perfor-

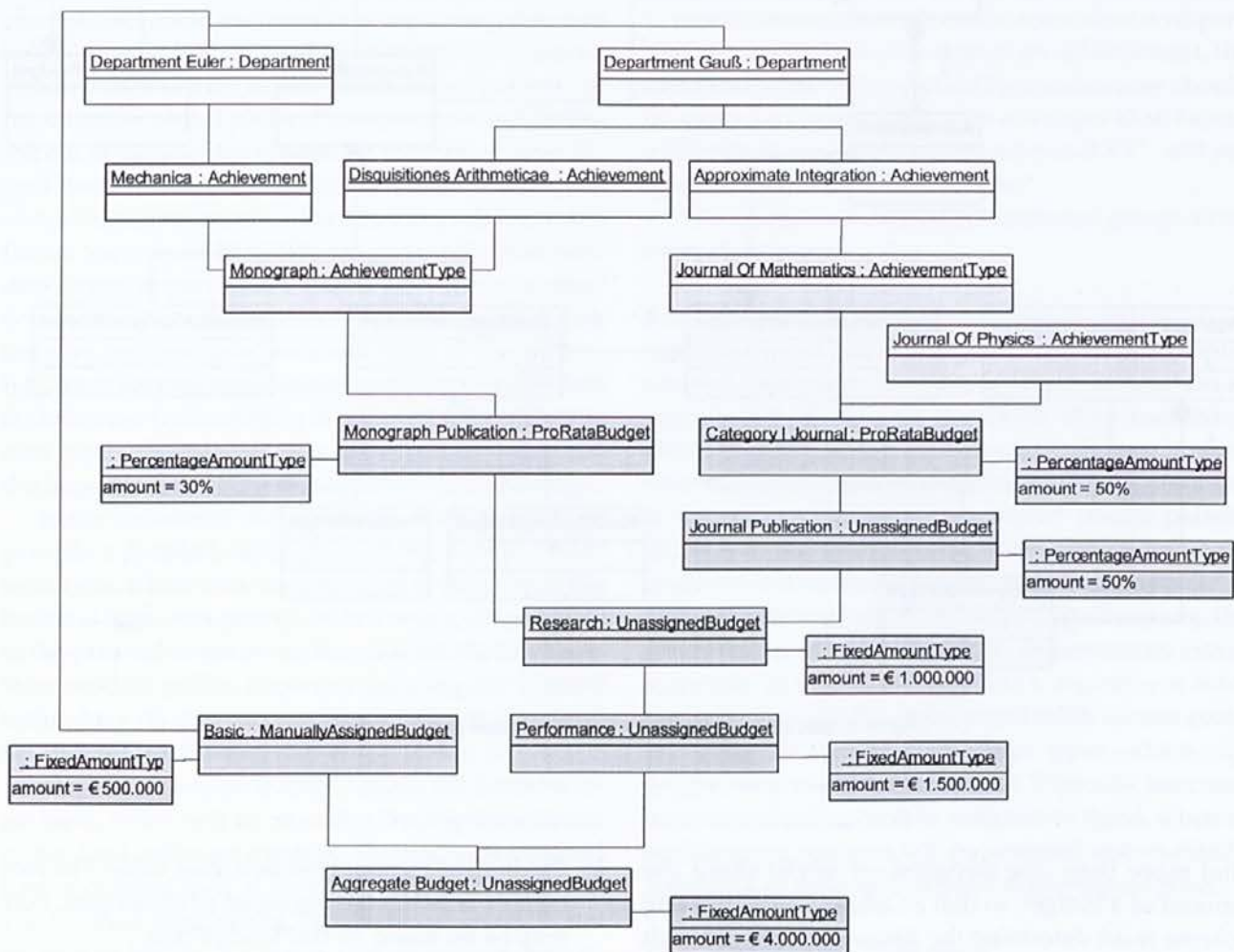


Figure 4: UML object diagram of an example scenario

mance budget (of type `UnassignedBudget`) with an amount of 1.500.000€ (also of type `FixedAmountType`). The basic budget is assigned to each department. The performance budget has a child budget `Research` (of type `UnassignedBudget`) with an amount of 1.000.000€ (of type `FixedAmountType`). This budget is subdivided into further sub-budgets, such as `MonographPublication` of type `ProRataBudget`, which means that the amount is distributed on a pro rata basis. The amount of the budget is of type `PercentageAmountType` and is set to 30% of the research budget. There is an object link to an achievement type object called `Monograph`, and the two achievements `Mechanica` and `Disquisitiones Arithmeticae` are of this type, so that they have to share the 30% of the research budget's amount. The other objects and links should be self-explanatory after this description.

In addition to the classes described above, the business layer must contain manager classes which are used to receive a list of all objects. This greatly simplifies the implementation of the other aspects of the system since a single point of access is convenient when operations should be carried out that affect all classes. In the default implementation the manager classes simply store a list of objects on the heap.

5 Evolution to a distributed version of IB-CUE

Currently, we are working on a distributed, multi-user version of IB-CUE to meet the real needs of potential users of the software. There are three main requirements: firstly, IB-CUE must be deployable in a heterogeneous IT infrastructure, secondly it must use standardized communication methods in order to be secure, and thirdly it should be possible to integrate the system with legacy applications. These problems are typically solved by using a middleware technology, which facilitates the distribution of the software system and provides standardized technical services for common infrastructure problems. It is obvious that the correct choice of this technology is crucial to the overall robustness and maintainability.

In order to be able to use a middleware technology at all, we encapsulated the business code into well-defined components. So, whatever method is chosen for distribution, it will result in network components functioning as proxies [6] for the business components, which do the actual work. This way, the code which is proven to be correct (cf. above) does not need to be changed.

Those and further considerations will lead to a final version of IB-CUE which, from an architectural point of view, can be split up into six different layers, each with its own responsibility. When a system is analyzed at a high level of abstraction, as is done in the following paragraphs, it is not pertinent to call these layers e.g. packages or modules. This would undoubtedly influence the way the system is looked at and induce to analyze it in an object-oriented or modular fashion. Since the best choice for an implementation technology is to be discussed later, we only speak of "layers" for now. The relevant layers are:

- business layer,
- presentation layer,
- distribution layer,
- authentication and authorization layer,
- database layer, and
- transaction layer.

The *business layer* contains the data structures which represent the elements of the business model described above, e.g. budgets, departments, etc. Also, the algorithms which work with the data and perform the required manipulations and calculations belong to this layer. These algorithms can be further split up into two different groups: While the first group assures the integrity of the data, the second group has knowledge of the changes that the data has to undergo when certain business actions are taken.

The *presentation layer* is responsible for displaying the objects of the business layer in a Graphical User Interface (GUI) and for providing the user with controls with which the business actions can be initiated.

The *distribution layer* allows the business objects to be distributed on multiple physical (or virtual) machines. It handles the technical details of the communication between the different machines and can add additional features like quality of service or load balancing. Furthermore the distribution layer must be able to communicate with legacy systems in order to integrate data which is hold there into the application.

The *authentication and authorization layer* governs a list of users who have access to the system. The permission to perform certain actions can be granted to or removed from a user at this layer.

The *database layer* is responsible for long-term persistence of the business objects in a database.

The *transaction layer* protects the system from damage which could arise from inconsistent data due to concurrent use of business data.

Although the described functionality can be implemented in several ways, in this case an approach must be chosen which has minimum impact on the business layer. Otherwise the testing and verification of the business algorithms which has already been done would have been futile. This means the interdependencies between the layers have to be designed in a way that allows every layer to interact with the business layer while the business layer itself exists in a clean room environment without any outgoing dependencies. Figure 5 shows which layer must have knowledge of which other layers. A layer must have knowledge of another layer only if it needs to influence the mode of operation of that other layer.

In the following subsections the dependencies between the layers are described briefly by giving examples of how a layer influences the operation mode of another layer. This will be the basis for the discussion on how the problem is to be addressed. The order of description of these dependencies matches the numbering of the arrows in Figure 5.

Dependency between transaction and database layers

The transaction layer needs to coordinate transactions with the database layer. Since a database management system (DBMS) usually comes with its own transaction processing monitor (TPM), in some cases the TPM of the DBMS can also be used to manage transactions outside of the DBMS, but in most cases it is the responsibility of the transaction layer to nest the database transactions in the application transactions [20].

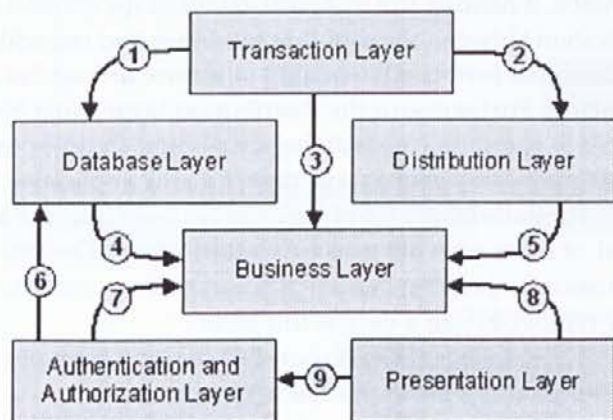


Figure 5: Dependencies between the layers

Dependency between transaction and distribution layers

In cases when several clients access data concurrently over a network or when data is held at many different locations, the transaction layer must be designed with respect to the special situations. One possible scenario is that a transaction needs to be rolled-back when a participant in the network exceeds response time. The example points up that the transaction layer must have knowledge of the details of the distributed system. Further examples can be found in [16].

Dependency between transaction and business layers

One of the most common activities of transaction processing is that resources are locked by the transaction manager – in this case the transaction layer. This can only be done when the transaction layer has access to the resources either directly or through a resource manager, but in both cases profound knowledge of the implementation details of the business layer is essential.

Dependency between database and business layers

The database layer not only needs to read all the values encapsulated in the business layer in order to store them in a DBMS, but it is also responsible for deciding which data needs to be kept on the heap at all. This requires keeping a record of the frequency in which the data in the business layer is accessed.

Dependency between distribution and business layers

In order to marshal data that is to be transmitted over the network, its structure must be known. This means that the distribution layer needs read and write access to the business layer.

Dependency between authentication/authorization and database layers

Similar to the example given above a DBMS usually has its own security system, which in most cases requires a form of authentication [14]. One responsibility of the authentication and authorization layer is to provide the DBMS with data necessary to authenticate the user. While the database login might differ from the login used to gain access to IB-CUE, the authentication and authorization layer need to be able to map one to the other.

Dependency between authentication/authorization and database layers

Not every user is authorized to work with all business data and to initialize every business action. The authentication and authorization layer therefore must have the possibility to disallow certain actions depending on the *identity* of the data and the user – it is not sufficient to control data access by only taking the *type* of the data into account, e.g. when a university member has the permission to view the budget of his school he does not at the same time have the permission to view the budget of every possible school.

Dependency between presentation and business layers

The presentation layer takes the role of the view and the controller in the commonly used model view controller pattern [5]. The relationship between the view and the model can be designed to be a one-way relationship as between the presentation layer and the business layer in IB-CUE.

Dependency between presentation and authentication/ authorization layers

The presentation layer must display a login dialog that provides the authentication and authorization layer with the data necessary for the authentication of the user.

The usual way to implement the described layers is to choose appropriate technologies and design the classes in a way that respects the requirements of the technologies to be used. For example, if the distribution layer should be implemented using Enterprise JavaBeans (EJBs) [5], the classes have to implement certain interfaces and use certain implementation patterns.

The most appropriate approach in this case is, however, to use a technique known as “aspect-oriented programming” (AOP) [8]. The name aspect-oriented programming is a summary for generic techniques which allow the software designer to separate concerns like distribution, transaction processing or authorization and authentication in a different module. The main business classes can still be written in a traditional object-oriented language. This is accomplished by describing the transformation that each class must undergo when a certain layer – or aspect like it is called in AOP – is added. For example the transaction aspect, when implemented in a naïve way, might make it ne-

cessary to add a flag to every class which tells the TPM if the class is currently in use and therefore access operations should be locked. Since transaction processing is not part of the core business logic the code should also be kept in a different module. This is exactly how AOP works: the code of the various aspects is woven into the core business code previous to the compilation of the system.

6 Conclusion

As a means to support state-owned universities in facing deteriorating financial conditions and an intensified performance competition, we have introduced an innovative budgeting system on the schools’ level, which is especially suited as a controlling instrument to guarantee basic operation of the schools, reward outstanding performance and to bring forward innovation. We have discussed design and implementation aspects of our Java-based monolithic software prototype IB-CUE, which implements this approach. Furthermore, we provided background information about the current evolution of this prototype into a distributed application with separate client and server components. We described basic considerations concerning a layered approach to the design of the final software product’s architecture. The next steps in our project will be to select suitable technologies for the implementation of those layers. For example, we will have to analyze the pros and cons of some of the currently most popular middleware technologies, among which are heavy-weight technologies such as the Common Request Broker Architecture (CORBA) [11], Java 2 Enterprise Edition (J2EE) [17], and the Common Object Model Plus (COM+) [15], as well as more lightweight technologies like Web Services [3], which have become very popular recently.

References

- [1] M. Aleksy, I. Bayer, T. Dickopp and A. Korthaus. Innovative Budgeting in a Competitive University Environment – Theoretical Concept and Design Considerations for an Information System. Proceedings of the International Workshop on Business and Information (BAI 2004), Taipei, Taiwan, March 26-27, 2004, chapter II-4, (2004).
- [2] I. Bayer. “Strategische und operative Führung von Fakultäten - Herausforderungen durch Autonomie und Wettbewerb”. Dissertation, Universität Mannheim, Hemmer Scientific, Frankenthal, pp. 4-7, (2002).
- [3] D. Booth et al. “Web Services Architecture”, W3C Working Group Note, (2004).

- [4] T. Cormen, C. Leiserson, R. Rivest. "Introduction To Algorithms", *MIT Press and McGraw Hill*, (1994).
- [5] L.G. DeMichiel. "Enterprise JavaBeans™ Specification, Version 2.1", Final Release, Sun Microsystems, (2003).
- [6] E. Gamma, R. Helm, R. Johnson, J. Vlissides. "Design Patterns: Elements of Reusable Object-Oriented Software," *Addison-Wesley*, (1995).
- [7] E. Kappler. "Die Universität kann Autonomie lernen". In: Tischer, S., Winckler, G., Biedermann, H. et al.: *Universitäten im Wettbewerb – Zur Neustrukturierung österreichischer Universitäten*. Rainer Hampp Verlag, München, pp. 297-330, (2000).
- [8] G. Kiczales, J. Irwin, J. Lamping, J.-M. Loingtier, C. V. Lopes, C. Maeda, and A. Mendhekar, "Aspect-oriented programming". *European Conference on Object-Oriented Programming 1997 (ECOOP 1997)*, LNCS 1241, pp. 220-242. Springer, (1997).
- [9] R. Ladd. "AspectJ in Action", *Manning Publications*, (2003)
- [10] B. Meyer. "Object-Oriented Software Construction", Sams, (1997).
- [11] Object Management Group. "Common Object Request Broker Architecture: Core Specification", OMG Adopted Specification, Tech. Doc. No. formal/04-03-12, (2004).
- [12] Object Management Group. "UML 2.0 Infrastructure Specification", OMG Adopted Specification, Tech. Doc. No. ptc/03-09-15, <http://www.omg.org/docs/ptc/03-09-15.pdf>, (2003a).
- [13] Object Management Group. "UML 2.0 Superstructure Specification", OMG Adopted Specification, Tech. Doc. No. ptc/03-08-02, <http://www.omg.org/docs/ptc/03-08-02.pdf>, (2003b).
- [14] G. Pernul. "Database Security", *Advances in Computers*, Vol. 38, *Academic Press*, pp. 1-74, (1994).
- [15] D.S. Platt. "Understanding COM+", *Microsoft Press*, Redmond, WA, (1999).
- [16] M. Rangarao, A. Vogel. "Programming with Enterprise JavaBeans, JTS and OTS, Building Distributed Transactions with Java and C++", *John Wiley & Sons*, (1999).
- [17] B. Shannon. "Java™ 2 Platform Enterprise Edition Specification, v1.4", Final Release, Sun Microsystems, (2003).
- [18] B. Sporn. "Adaptive University Structures - An Analysis of Adaptation to Socio-economic Environments of US and European Universities". *Jessica Kingsley, London*, (1999).
- [19] W.-D. Webler. "Qualität der Lehre als Gegenstand staatlicher Steuerung". In: Neusel, A., Teichler, U., Winckler, H. (Hrsg.): *Hochschule-Staat-Politik*, Christoph Oehler zum 65. Geburtstag, Frankfurt, New York, pp. 235-256, (1993).
- [20] G. Weikum, G. Vossen. "Fundamentals of Transaction Information Systems: Theory, Algorithms, and Practice of Concurrency Control and Recovery", *Morgan Kaufmann Publishers*, (2001).

Dr. Markus Aleksy studied in Management Information Systems at the University of Mannheim, Germany. He holds a doctorate degree from the University of Mannheim. His research interests include analysis, design, implementation, and evaluation of distributed systems, especially based on CORBA.

Tilo Dickopp studied Computer Science and Applied Mathematics at the University of Mannheim. He is currently working on his doctorate degree. His research interests include object-oriented analysis, design, implementation, and evaluation of distributed systems, especially based on J2ME, as well as mobile electronic commerce.

Dr. Ingo Bayer studied Business Administration at the University of Mannheim. He holds a doctorate degree from the University of Mannheim and is currently Managing Director of the Business School at the University of Mannheim. His research interest lies in the transition Process from a state dominated system of higher education to a more market oriented model, especially in the areas of strategic planning, management, finance and budgeting.

Dr. Axel Korthis studied Management Information Systems at the University of Mannheim. He holds a doctorate degree from the University of Mannheim and is currently working on his professorial dissertation. He is the project leader of a project on collaborative, component-based business application software development and gives lectures on OOA/OOD, OODB, and CBSE. His research interests include object-oriented analysis, design, implementation, and evaluation of distributed systems, especially based on J2EE, as well as knowledge management in software engineering.

ESUP-Portail: open source Single Sign-On with CAS (Central Authentication Service)

Pascal Aubry¹, Vincent Mathieu², Julien Marchal²

¹IFSIC, University of Rennes 1, France,

pascal.aubry@univ-rennes1.fr

²University of Nancy 2, France,

vincent.mathieu@univ-nancy2.fr, julien.marchal@univ-nancy2.fr

Abstract

The universality of the HTTP protocol seduced developers for quite a long time; in fact, today, most applications are web-based. LDAP directories may make life easier on our users' brains by making them memorize only one password, but their fingers are still very busy with all the authentications they need to type — in practice, each time they access an application.

Many solutions for Single Sign-On (SSO) are already available. In this article, we describe a free, simple, complete and reliable solution: the CAS (Central Authentication Service), developed by Yale University (New Haven, CT; USA). CAS has been chosen by the French ESUP-Portail consortium, which provides a complete and open solution to Universities and University-level colleges who wish to offer integrated access to their services and information for their students and staff.

Keywords: Single Sign-On (SSO), open-source, authentication.

Povzetek

ESUP-Portail: odprtokodna rešitev problema enkratne prijave z uporabo CAS

Univerzalnost protokola HTTP že dolgo privlači razvijalce tako, da danes do večine aplikacij dostopamo po svetovnem spletu. Direktoriji vrste LDAP nam prihranijo miselni napor, ker omogočajo, da si je potrebno zapomniti le eno geslo, vendar imajo prsti še vedno "polne roke dela", ker je potrebno geslo vnašati večkrat – v praksi vsakič, ko dostopamo do neke aplikacije.

Za enkratno prijavo (Single Sign-On ali SSO) obstaja več rešitev. V pričujočem članku opišemo odprtokodno, enostavno, popolno in zanesljivo rešitev, ki se ji pravi CAS (Central Authentication Service), razvita pa je bila na univerzi Yale (New Haven, Connecticut, ZDA). CAS uporablja tudi francoski konzorcij ESUP-Portail, ki ponuja odprtokodno in popolno rešitev naloge zagotavljanja integriranega dostopa do storitev in informacij na univerzah in visokih šolah s strani študentov in zaposlenih.

Ključne besede: enkratna prijava, odprta koda, avtentikacija.

1 Why do we need Single Sign-On?

Web-based applications (mailers, forums, agendas, other specific applications) have spread widely over our networks during recent years. These applications often require authentication.

Using LDAP directories provides users with a single account, which is obviously a real improvement. However, some issues remain:

- **Multiple authentications:** it is still necessary to give a netId/password for each application;
- **Security:** because user accounts are unique, password stealing is really a critical problem; for this reason, the security of the authentication process is essential. In addition, security concerns indicate that user credentials should no longer be given to applications.
- **Several authentication mechanisms:** there are a variety of authentication mechanisms. LDAP is one widely used standard today, but it can be replaced by other user databases. In addition, some users possess personal X509 certificates [1], which can be used for authentication. In any case, bypassing authentication mechanisms would be interesting because it would permit the use of mixed authentication, for instance.
- **Cooperation:** particularly in the educational community, close institutions would like to be able to share resources and applications. Such cooperation implies that users should be identified by an establishment when only authenticated with another one.

- **Authorization:** applications often need to know user profiles to allow (or deny) the performance of specific actions.

The principle of all SSO solutions is to remove authentication from the applicative code. The goal is then to offer a globally secured software environment:

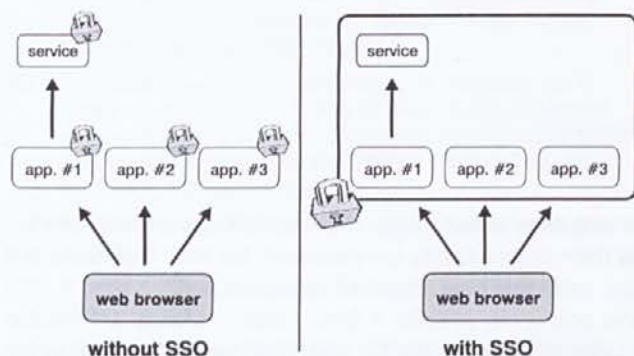


Figure 1: The principle of Single Sign-On

Most SSO mechanisms [2] try to achieve this goal using similar techniques:

- **Authentication is centralized** via an encrypted tunnel in a unique server, which is the only machine to receive user credentials;
- **HTTP redirections** are sent from applications to the authentication server for unauthenticated users, and then back to the applications once a user is authenticated;
- **Information is passed** by the authentication server to applications during the redirections, via cookies [3] and/or CGI parameters.

Among the commercial solutions offered to system administrators and developers, two leaders stand out: Sun One Identity Server [4] and Microsoft Passport [5]. However, because the ESUP-Portail project is based on open-source software only, neither of these commercial solutions was a feasible choice for this project (however, none of them was fitting to our needs). After testing several free implementations, the ESUP-Portail SSO group chose CAS (Central Authentication Service [6], developed by Yale University) for its Single Sign-On mechanism.

2 The reasons why we chose CAS

We chose for a number of reasons. CAS is made up of java servlets, can be run on any (JSP spec 1.2 compli-

ant) servlet engine, and offers a web-based authentication service. Its strong points include security, and flexibility, as well as its proxying features and numerous client libraries.

2.1 Security

Security is insured the following ways:

- Passwords are always sent through an encrypted tunnel, and pass only from the browser to the authentication server (Figure 2);
- Re-authentications are transparent to users, providing that they accept a single cookie, called the 'Ticket Granting Cookie' (TGC). This cookie is opaque (no personal information), protected (HTTPS) and private (only presented to the authentication server);
- Applications learn user identities from opaque one-time 'Service Tickets' (ST). Those tickets are emitted by the authentication server, transmitted to applications by the browsers, and finally validated by the authentication server (returning the corresponding identity). Thus, as it is the case for almost all serious SSO mechanisms, applications never see any password.

2.2 Flexibility

The package proposed by CAS developers offers a complete implementation of the authentication protocol, but the authentication itself (against a user database) is left to the administrator. We wrote a generic handler that provides several connectors (LDAP, X509 certificates, NIS domains, databases). These connectors can be used separately, or they can be used in combination to permit mixed authentication. This generic

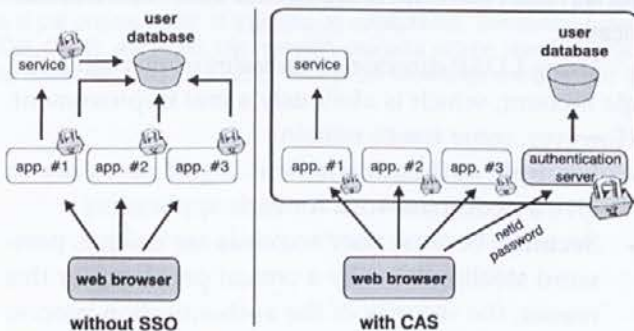


Figure 2: A software environment secured by CAS

handler can also be extended to allow system administrators access to other authentication methods, such as Kerberos or Active Directory.

2.3 Authentication proxying

Classic SSO mechanisms demand direct communication between the browser and the application, which prohibits the use of multi-tier installations in which an application must request a back-end service requiring authentication (for instance, a portal requesting a web service).

CAS v2.0 solves this problem by providing an elegant way to propagate authentication without propagating passwords; dedicated tickets (PGT: Proxy Granting Ticket and PT: Proxy Ticket) ensure the validity of user identities to third-party applications. This feature is obviously the strongest point of CAS, inherited from Kerberos concepts.

2.4 Client libraries

The code that handles the basic protocol (apart from proxying) is very simple to write on the client-side (applications). CAS provides **Client libraries** for Perl, Java, ASP and PL/SQL. We added a strong (proxy-able) PHP library. These libraries make it possible to CAS-ify existing applications by simply adding a few lines of code, offering an impressive degree of flexibility.

An **Apache** module (`mod_cas`) lets web servers authenticate users for static resources, since client libraries can not be used in such circumstances.

A PAM (**Pluggable Authentication Module** [7]) module (`pam_cas`) allows non web-based applications to be integrated at a very low level.

2.5 Moreover...

In addition to the reasons described above, CAS has another overwhelming argument in its favor: it is used by many American Universities, in conjunction with LDAP or Kerberos-based authentication, making us confident of its **durability** over time.

Even better, CAS can be plugged directly into **uPortal** [8], (used in the ESUP-Portail) which is on the way to becoming a standard for open source portals.

This article shows how Single Sign-On is achieved with CAS, and focuses on a precise technical issue: CAS-ifying a webmail (Horde IMP) and an IMAP server (Cyrus IMAP).

3 How CAS works

3.1 Architecture

3.1.1 The CAS server

Authentication is centralized on a single machine, called the CAS server. This machine is the only actor that knows user passwords. It has a double role:

- Authenticating users;
- Transmitting and certifying the identities of authenticated users to CAS clients.

3.1.2 Web browsers

Web browsers must meet the following requirements to take advantage of all CAS's easy features. They must:

- Own an **encryption engine** so that it can use HTTPS;
- Perform **HTTP redirections** (access a URL given by a Location header when receiving 30x responses) and understand **basic Javascript**;
- Store **cookies**, as defined in Š3Ć. In particular, for security purposes, private cookies should be transmitted only to the machines that emitted them.

These requirements are met by all classic web browsers, such as Microsoft Internet Explorer (since 5.0), Netscape Navigator (since 4.7) and Mozilla.

3.1.3 CAS clients

A web application equipped with a CAS client library, or a web server using `mod_cas`, is called a CAS client. It delivers resources only to clients previously authenticated by the CAS server.

CAS clients include:

- **Libraries**, compatible with the most widely used web-programming languages (Perl, Java, JSP, PHP, ASP);
- An **Apache module**, used in particular to protect static documents;
- A **PAM module**, used to perform system level authentication.

3.2 Basic operating procedure

3.2.1 User authentication

A previously non-authenticated user (or a user whose authentication has expired) accessing the CAS server

is presented with an authentication form, on which (s)he is invited to enter a netId and a password (Figure 3).

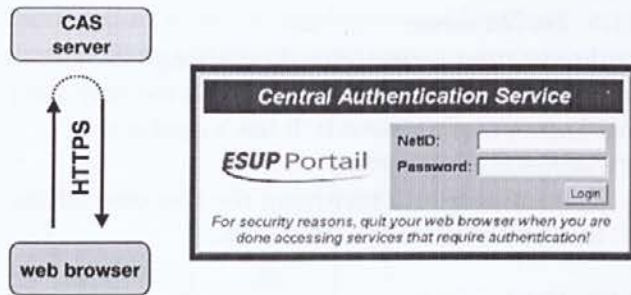


Figure 3: First access of a browser to the CAS server

If the netId and password are correct, the server sends a cookie called TGC (Ticket Granting Cookie) to the browser (Figure 4).

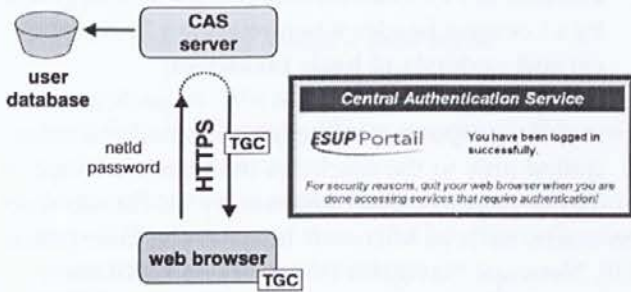


Figure 4: Authentication of a browser against the CAS server

This TGC is the user's passport to the CAS server. Its validity is limited, typically to a few hours. The TGC allows web browsers to get CAS client tickets from the CAS server, without needing to re-authenticate. A private cookie that is transmitted only to the CAS server, it is also protected ensuring that all requests to the CAS server are secured. Like all CAS tickets, it is opaque, containing no information about the user. It is simply a session identifier operating between the web browser and the CAS server.

3.2.2 Accessing protected web resources after authentication

When accessing a resource protected by a CAS client, the web browser is redirected to the CAS server. The browser, if previously authenticated, presents its TGC to the CAS server (Figure 5).

On presentation of the TGC, the CAS server delivers a Service Ticket (ST), an opaque ticket providing no user information, that is usable only by the service that asks for it. At the same time, the CAS server re-

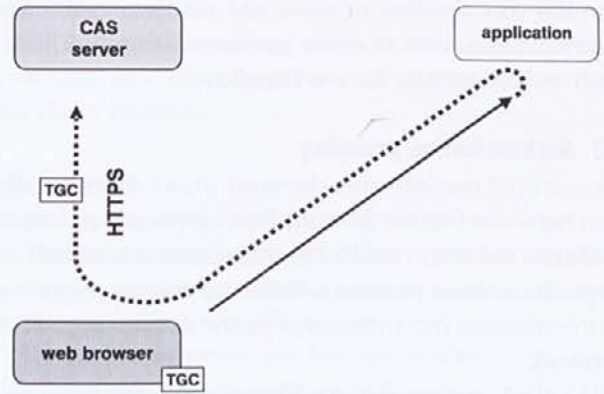


Figure 5: Redirection of an unknown browser to the CAS server

directs the browser to the calling service (the Service Ticket is a CGI parameter) (Figure 6).

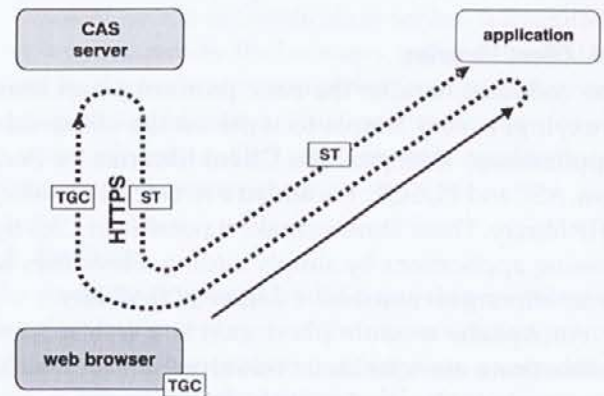


Figure 6: Redirection of the browser to the calling service after authentication

The ST is then validated by the CAS client with the CAS server (via an HTTP request) and the desired resource can be delivered to the browser (Figure 7).

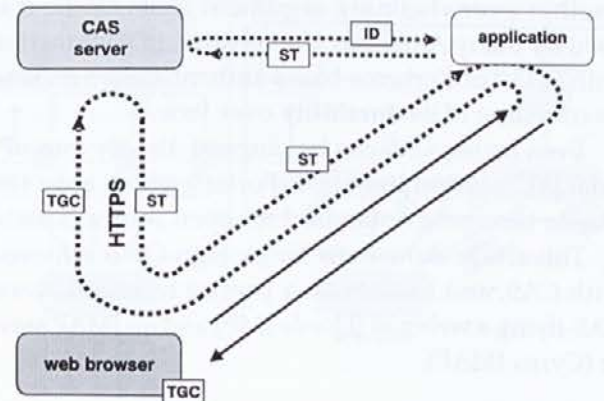


Figure 7: Validation of a Service Ticket

All the redirections mentioned above are transparent for the user: (s)he accesses the resource without being authenticated, without any interaction at all (Figure 8).

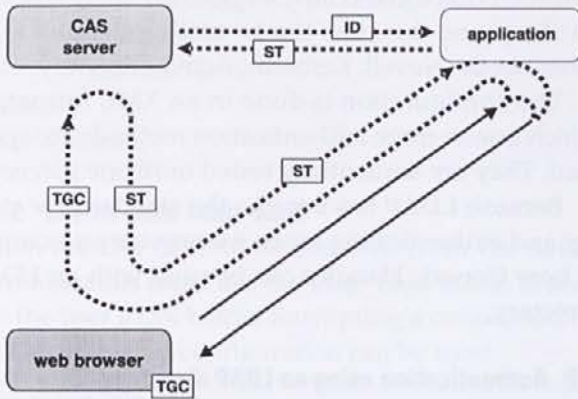


Figure 8: User view of the CAS redirections

The Service Ticket (ST) is the browser’s passport to a CAS client. This One-Time Ticket can not be presented twice to the CAS server, is valid only for the CAS client to whom it was delivered, and can be used only for a very short period of time (typically a few seconds).

3.2.3 Accessing protected web resources without authentication

The non-authenticated web browser attempting to access a resource protected by a CAS client is still redirected to the CAS server. As no TGC is given, the CAS server returns an authentication form.

Once the browser has submitted the form and is correctly authenticated, the CAS server:

- Sends the browser a TGC, that will exempt it from re-authenticating later;
- Provides a Service Ticket and redirects the browser to the CAS client calling service.

Thus, there is no need to be previously authenticated to access a protected resource: authentication is automatically performed the first time a user requests a protected resource.

3.3 Multi-tier installations

3.3.1 CAS proxies

The CAS multi-tier feature makes it possible for a CAS client to access a back-end service under the authenticated user’s identity. A CAS client that is able to

proxy credentials is called a CAS proxy. Most used CAS proxies are:

- Web portals, which need to access external applications (web services [9] for instance) under user identities;
- Webmail applications, which need to connect to an IMAP server to retrieve email under user identities.

In a multi-tier CAS installation, CAS clients no longer have access to the browser’s cookie cache, and so redirections can not be used.

3.3.2 2-tier installations

A CAS proxy, when validating a Service Ticket to authenticate a user, also asks for a PGT (Proxy Granting Ticket) from the CAS server (Figure 9).

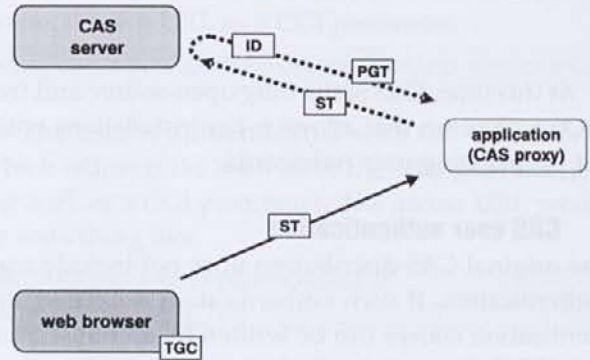


Figure 9: PGT retrieval by a CAS proxy

A PGT is the CAS proxy’s passport, representing a user, to the CAS server. It allows the CAS proxies to obtain tickets for CAS back-end services from the CAS server, without needing to validate an ST. The PGT is an opaque and re-playable ticket, delivered by the CAS server in response to a secured request, to insure TGC integrity and confidentiality. PGT validity is limited, like that of TGC, to a few hours.

PGT are to applications what TGC are to web browsers. A PGT allows applications (CAS proxies) to authenticate a user with the CAS server, and to obtain Proxy Tickets (PT are to CAS proxies what ST are to web browsers). Proxy Tickets, like Service Tickets, are validated by the CAS server before allowing access to protected resources (Figure 10).

3.3.3 N-tier installations

Obviously, the back-end service accessed by the CAS proxy in 2-tier installation can itself be a CAS proxy. CAS proxies can be chained, as shown in Figure 11.

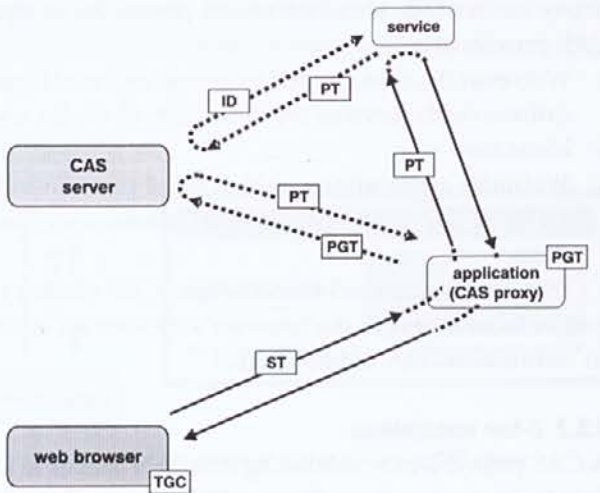


Figure 10: Validation of a Proxy Ticket by a back-end service

At this time, CAS is the only open-source and free SSO mechanism that allows n-tier installations without propagating user passwords.

4 CAS user authentication

The original CAS distribution does not include user authentication. If such authentication is needed, authentication classes can be written by administrators to fulfill their exact needs. Some example classes are provided by the cas-server distribution, for test purposes only.

4.1 The GenericHandler class

Developed by the ESUP-Portail project [10], the GenericHandler class [11] allows the implementation of many authentication methods: LDAP directories, databases, NIS (Unix Yellow Pages), NT domains, etc. Furthermore, this class can be easily extended to fit other needs (Novell, Kerberos, Active Directory, etc.).

The configuration is done in an XML format, in which one or more authentication methods are specified. They are sequentially tested until one succeeds.

Because LDAP has become the standard for storing and authenticating users, we provide an example of how Generic Handler can be used with an LDAP directory.

4.2 Authentication using an LDAP directory

Two different access modes are proposed, depending on the internal structure (DIT) of the LDAP directory.

4.2.1 Direct access mode (ldap_fastbind)

ldap_fastbind mode can be used with LDAP directories in which user DN (Distinguished Name) can be directly deduced from their netId. In practice, these are directories in which users are stored at the same hierarchical level, in the same OU for instance.

In this case, CAS tries to connect the directory using the DN and password provided by the user. Classically, the user is authenticated if the connection succeeds.



Figure 11: A chain of CAS proxies

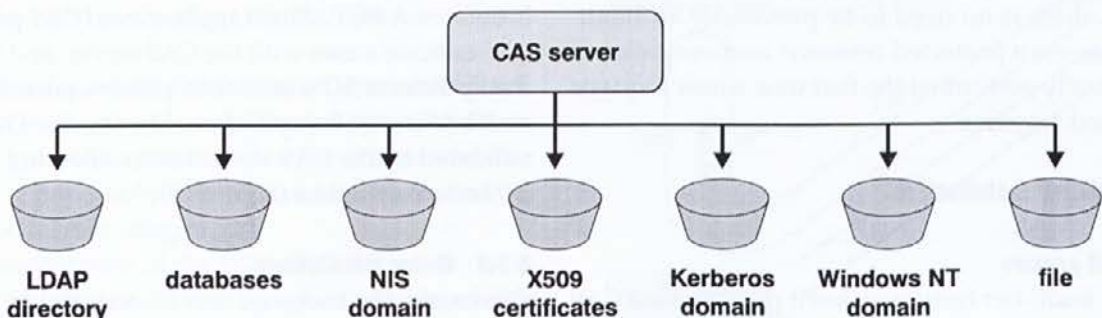


Figure 12: User authentication with ESUP-Portail GenericHandler

The following configuration can be used:

```
<authentication>
  <ldap version="3" timeout="5">
    <ldap_fastbind filter="uid=%u,dc=univ-rennes1,dc=fr" />
    <ldap_server host="ldap.ifsic.univ-rennes1.fr"
      port="389"
      secured="no" />
  </ldap>
</authentication>
```

4.2.2 Search mode (ldap_bind)

When the DN can not be deduced from the netId, administrators must use the ldap_bind mode to seek out the user's DN before attempting a connection.

The following configuration can be used:

```
<authentication>
  <ldap version="3" timeout="5">
    <ldap_bind search_base="dc=univ-rennes1,dc=fr"
      scope="sub" filter="uid=%u"
      bind_dn="admin" bind_pw="secret" />
    <ldap_server host="ldap.ifsic.univ-rennes1.fr"
      port="389" secured="no" />
  </ldap>
</authentication>
```

4.2.3 LDAP Redundancy

Generic Handler permits redundancy in order to be more fault-tolerant: it is possible to specify a list of LDAP servers, which are considered as replicas.

5 CAS-ifying a web application

CAS-ifying a web application is very easy, thanks to CAS client libraries.

Three kinds of CAS applications exist:

- CAS "simple" clients only need to authenticate users.
- CAS proxies need to both authenticate users, and use tier services. They need to be able to retrieve PGT from the CAS server, and later to transmit PT to back-end services in order to authenticate the users for whom they act.
- CAS back-end services need to validate the PT given by CAS proxies and to obtain user identities.

5.1 "simple" CAS clients

The principle is to use a function (or method) that will run the authentication mechanism and return the user's netId. This function must perform the following tasks:

- If the user is not already authenticated and no ST is provided, redirect the web browser to the CAS server (providing its own URL for redirection as shown in 3.2.2);
- If the user is not already authenticated and a ST is provided, validate the ST by using an HTTPS request to the CAS server. The CAS server should then return the corresponding user's netId.

To illustrate the simplicity of the CAS-ification of such a "simple" CAS client, we show below how a CAS client can be written in PHP. Of course, in a real application, a client library – in our case, phpCAS [12] – should be used instead.

5.1.1 Writing a PHP CAS client

If this script (`script.php`) is called without any parameter, it redirects the web browser to the CAS server, giving its own URL as a CGI parameter:

```
https://cas.univ.fr/login?service=http://test.univ.fr/script.php
```

The user is authenticated with the CAS server, which redirects the browser to the calling service, giving a ST as a CGI parameter. The access URL would be something like:

```
http://test.univ.fr/script.php?ticket=ST-2
uw2KEWinSFeZ9fotZlio
```

Our script will then try to validate the Service Ticket with the CAS server, by accessing the following URL:

```
http(s)://auth.univ.fr/serviceValidate?service=http://
test.univ.fr/script.php&ticket=ST-2-uw2KEWinSFeZ9fotZlio
```

The CAS server validates the ticket and returns the user's netId, in an XML response:

```
<cas:serviceResponse xmlns:cas='http://www.yale.edu/tp/
cas'>
  <cas:authenticationSuccess>
    <cas:user>paubry</cas:user>
  </cas:authenticationSuccess>
</cas:serviceResponse>
```

A possible implementation of this script is:

```
<?php /* PHP simple Cas client */
// localization of the CAS server
define('CAS_BASE','https://auth.univ.fr');
// own URL
```



```

$service='http://
'$_SERVER['SERVER_NAME'].$_SERVER['REQUEST_URI'];
/** Authenticate with the CAS server
 * @return the user's netId, or FALSE on failure
 */
function authenticate() {
    global $service;

    // retrieve the ticket
    if (!isset($_GET['ticket'])) {
        header("Location: '$CAS_BASE.'/
login?service='.$service);
        exit();
    }
    // try to validate the ST with the CAS server
    $fpage = fopen (CAS_BASE . '/serviceValidate?service='
        . preg_replace('/&/','%26',$service)
        . '&ticket=' . $ticket, 'r');
    if ($fpage) {
        while (!feof ($fpage)) { $page .= fgets ($fpage, 1024); }
    }
    // analyze the CAS server's response
    if (preg_match('<cas:authenticationSuccess>|mis', $page))
    {
        if(preg_match('<cas:user>(.*?)</cas:user>|',
$page,$match)){
            return($match[1]);
        }
    }
    // validation failed
    return FALSE;
}
if (($login = authenticate()) === FALSE ) {
    echo 'failure (<a href="'. $service. ">Retry</a>).';
    exit() ;
}
echo 'welcome user '$login!<br>'
echo '<a href="'. $CAS_BASE. '/logout"><b>logout</b></a>';
?>

```

5.1.2 Using the phpCAS client library

The phpCAS library [12] was developed by the ESUP-Portail project. Here is one way that it can be used:

```

<?php /* a simple CAS client using phpCAS */
include_once('CAS.php');
phpCAS::client(CAS_VERSION_2_0,'cas.univ.fr',443,");
phpCAS::authenticateIfNeeded();

```

```

?>
<html>
<body>
    <h1>Authentication succeeded!</h1>
    <p>User is <?php echo phpCAS::getUser(); ?></b></p>
</body>
</html>

```

5.2 CAS proxies

The procedure begins exactly as the one for “simple” CAS clients: retrieve a Service Ticket.

Next, when validating the ST, an additional parameter is given to the CAS server: a callback URL. In response, the CAS server returns:

- The user's netId (as for an ordinary CAS client); and
- A PGT, using the callback URL.

As seen in 3.3.2 (“2-tie”), the PGT will be used later to authenticate a user with the CAS server and to obtain the Proxy Tickets needed to access back-end services.

Java and PHP libraries mask the complexity of all this when developing a CAS proxy. The following is an example of the way a CAS proxy can be implemented using the phpCAS library:

```

<?php /* a CAS proxy using phpCAS */
include_once('CAS.php');
phpCAS::proxy(CAS_VERSION_2_0,'auth.univ.fr',443,");
phpCAS::authenticateIfNeeded();
?>
<html><body>
<p>User's netId: <?php echo phpCAS::getUser(); ?></p>
<?php
flush();
if (phpCAS::serviceWeb('http://test.univ.fr/ws.php',
$err_code, $output) {
    echo $output;
}
?>
</body></html>

```

5.3 CAS back-end services

Back-end services are as easy to CAS-ify as “simple” CAS clients because they do exactly the same job, i.e. validating a Proxy Ticket with the CAS server (instead of a Service Ticket).

The back-end service called by the CAS proxy shown before could be:


```
<?php /* a simple CAS back-end service */
include_once('CAS.php');
phpCAS::client(CAS_VERSION_2_0,'cas.univ.fr',443,");
phpCAS::authenticateIfNeeded();
echo '<p>User is ' . phpCAS::getUser() . '</p>';
?>
```

5.4 Precautions to take when CAS-ifying web applications

5.4.1 Sessioning

For obvious performance reasons, applications should maintain sessions so that the CAS mechanism is fired only once, rather than for each request.

This remark is true for CAS clients and proxies, which should maintain a session with the browser, as well as for back-end services, which should maintain a session with the CAS proxy.

5.4.2 Asynchronism

When using CAS client libraries, retrieving a PGT for a user in a CAS proxy is easy. However, developers should be aware of the possibility of desynchronization between the different sessions of a multi-tier CAS installation.

Imagine that a user connects to a web portal, which will act as a CAS proxy. The user is authenticated with the CAS server, the portal retrieves a PGT for the user, and a session is set between the portal and the browser. This session is set to last a few hours.

Let us now imagine that the PGT becomes invalid (expiration or user logout from another window of the browser). In this particular configuration, it is impossible for the portal to obtain the new PTs needed to access back-end services.

This situation should be handled by CAS proxies, by forcing the disconnection of the user, for instance.

5.5 CAS authentication for static web pages

The CAS mechanism can be used to protect static resources (typically HTML web pages), thanks to the `mod_cas` Apache module.

With simple Apache directives, access to a site (or part of it) can require an authentication from a CAS server. For instance, the following directives will redirect users to the CAS server located at `https://cas.univ.fr/cas` if no valid ST is given by browsers:

```
CASServerHostname cas.univ.fr
CASServerPort 8443
```

```
CASServerBaseUri /cas
CASServerCACertFile /etc/x509/cert.root.pem
<Location /protected>
AuthType CAS
Require valid-user
</Location>
```

6 CAS-ifying a non-web application

The main goal of an SSO mechanism is, of course, to provide a single authentication service for web applications that is both simple and efficient. CAS offers more by allowing the CAS-ification of non-web services, such as IMAP, FTP, etc.

In order to do this, these services must use PAM (Pluggable Authentication Module), as most Unix services now do.

6.1 The PAM `pam_cas` module

`Pam_cas` is included in CAS client distribution. Though it is powerful, it is also light (about 300 lines of C, half of them shared with `mod_cas`).

This module allows a service to authenticate a user by receiving an identifier (usually a `netId`) and a ticket (instead of a password). The ticket received by the service is then validated with the CAS server by `pam_cas`.

`Pam_cas` can not be used outside of a multi-tier installation: the CAS-ified service must be accessed by a CAS proxy. Indeed, it is inconceivable for a human being (a human user of an FTP service, for instance) to provide a CAS ticket.

Fortunately, the PAM modular concept allows `pam_cas` to be used in conjunction with other PAM modules. It is possible for a service to authenticate a user both using a traditional method (a `netId` and a password) and using a CAS method (`netId` and ticket).

The example below shows how this can be done.

6.2 Using `pam_cas` to CAS-ify an IMAP server

Our goal here is to CAS-ify an IMAP server in order to accept connections from a web portal using Proxy Tickets, while continuing to accept connections from traditional mail clients using passwords.

If the IMAP server is PAM-compliant (which is generally the case), the PAM configuration can be, for instance:

```
auth sufficient /lib/security/pam_ldap.so
auth sufficient /lib/security/pam_pwddb.so shadow nullok
```



```
auth required /lib/security/pam_cas.so \
-simap://mail.univ.fr \
-phhttps://ent.univ.fr/uPortal/CasProxyServlet
```

In this example, authentication will be attempted in three different ways: with an LDAP directory, with the local Unix user database, and eventually with `pam_cas`. The secret provided is validated with the CAS server (internally, only if it is ticket-shaped for obvious performance reasons).

6.3 CAS-ifying the Cyrus-IMAP server

The IMAP protocol is very specific, and is probably the most difficult to CAS-ify. IMAP clients and web-mails in general have the odd habit of generating large numbers of requests, by closing and opening connections repeatedly. This, of course, leads to numerous authentication requests for the CAS server.

When using a traditional webmail (where users authenticate with their `netId` and password), the only consequence is a heavier load for the web server running the webmail. However, within a CAS multi-tier installation, the load increase is supported both by the web server running the webmail, and by the CAS server.

Asking for and validating tickets for each request is clearly prohibitive from the point of view of performance. Consequently, a cache is needed on the IMAP server to allow the webmail to re-play the PT.

The implementation of such a cache comes straight with Cyrus. Indeed, Cyrus IMAP server uses Cyrus-SASL for authentication; now, Cyrus-SASL can also use other authentication mechanisms (PAM, LDAP, Kerberos, etc.) or call a Unix daemon, `saslauthd`.

This daemon, which communicates with Cyrus-SASL via a Unix socket, proposes a cache mechanism. This cache allows the mail client to play the same PT more than once, because `saslauthd` will not use PAM once the ticket is stored in its cache.

CAS-ifying Cyrus-IMAP this way reduced authentication requests by 95%. Only 5% were really played, i.e. with the tickets being validated by the CAS server.

6.4 CAS-ifying Horde IMP

Our primary goal was to add a webmail product into the ESUP-Portail software, if possible completely integrated into our SSO. We decided to try Horde IMP [13].

At first, IMP was adapted to become a CAS proxy. This was easily done by using the `phpCAS` library, as shown in 5.2 ("CAS proxies"). It was then possible to acquire a Proxy Ticket and make the IMAP server authenticate users, by validating PTs with the CAS server.

Next, the behavior of the webmail was modified to take into account the versatility of this new kind of password. Indeed, PTs are manipulated in the same

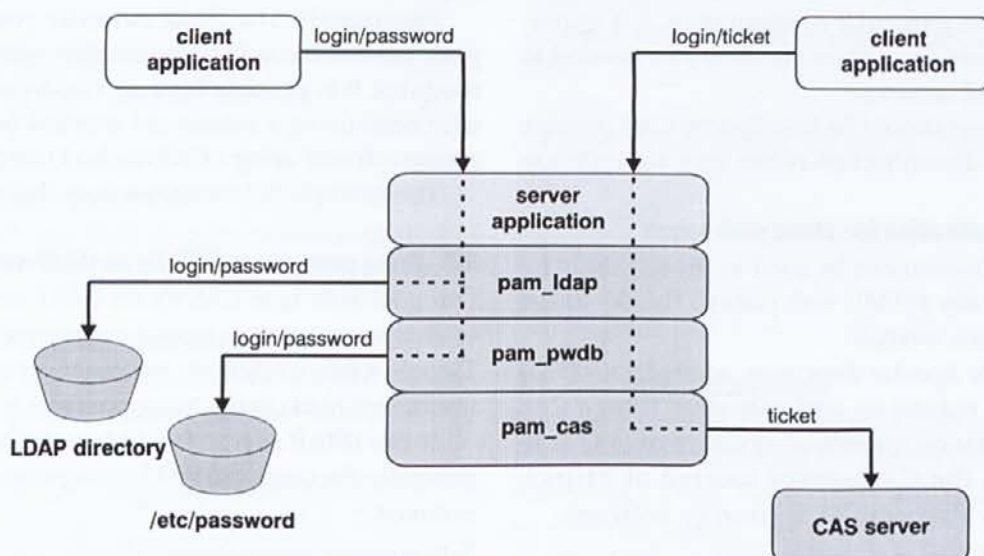


Figure 13: Using `pam_cas`

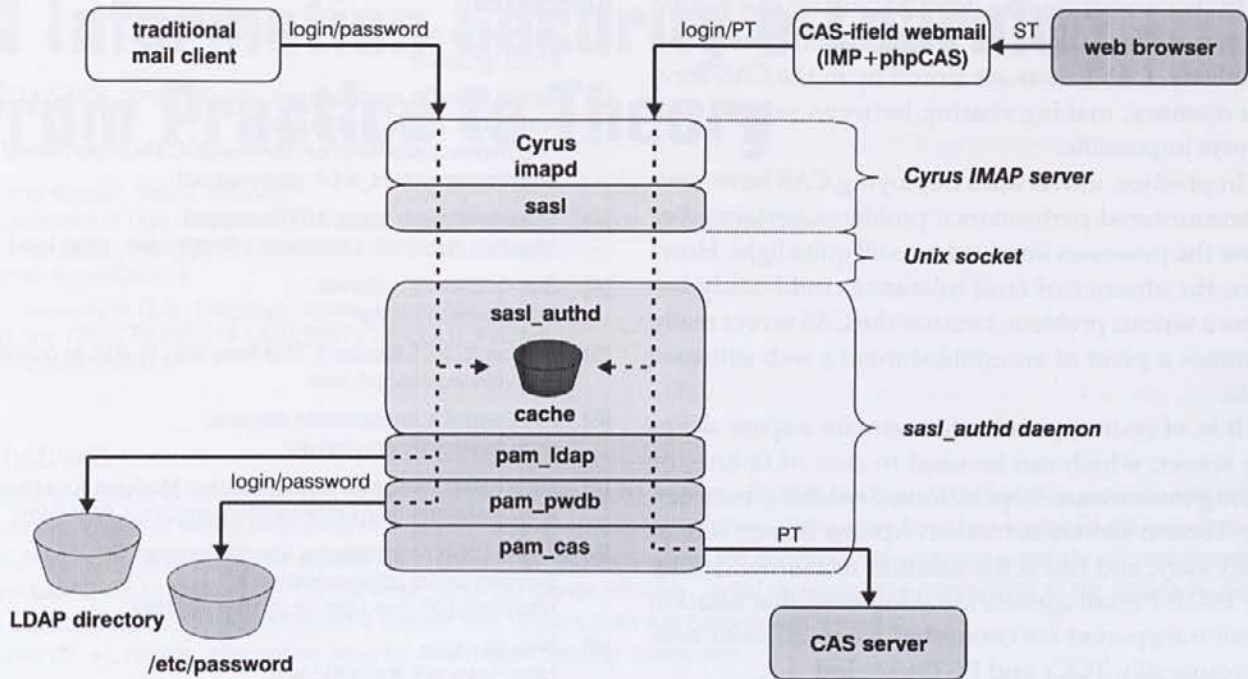


Figure 14: CAS-ification of Cyrus-IMAP

way that passwords are, although their validity is limited. In other words, the webmail can use a PT several times thanks to the IMAP cache, but a PT stored in the IMAP cache can be erased (because of the garbage collector of the IMAP cache), supplanted in the cache by another PT (if another webmail instance is running for the same user), or simply replaced by the user's password if the user concurrently accesses a traditional mail client. In all of these situations, the next connection with the PT would be refused by the IMAP server. To get around this problem, the webmail was modified to allow a new PT to be acquired from the CAS server, in order to make a second attempt at an IMAP connection.

Obviously, using CAS client libraries are not as simple as was implied in 5.1.1 ("Writing a PHP CAS client").

7 Restrictions and perspectives

In the first 6 sections of this article, we have described the strong points of CAS:

- CAS is an open-source, free product;
- CAS offers a more than satisfactory level of security;

- A CAS server is very easy to set up and configure;
 - Web applications are very easy to CAS-ify.
- In this section, we will examine its limitations, and offer some ideas for getting around these weak points.

7.1 CAS brings SSO, nothing else

CAS is a Single Sign-On mechanism that can also run at the system-level, thanks to `pam_cas`. On the other hand, it is strictly limited to user authentication: it does not (and probably never will) deal with authorizations or with the propagation of user attributes.

Moreover, the user databases are local, at the establishment-level. CAS does not address multi-establishments issues. Recent developments on Sympa [14] show an elegant way to allow users from several establishments to be authenticated, by relying on several CAS servers. However, the most promising way to permit different establishments to cooperate using CAS is certainly the Shibboleth internet2 project [15].

7.2 Performance and fault-tolerance

In a CAS installation, all the web applications depend on the CAS server. Its availability is critical.

In its current release, load balancing can not be implemented. Indeed, for reasons of efficiency and simplicity, CAS tickets are stored by in the CAS server's memory, making sharing between several CAS servers impossible.

In practice, universities deploying CAS have never encountered performance problems, certainly because the processes involved are still quite light. However, the absence of fault tolerance could easily become a serious problem, because the CAS server really becomes a pivot of an establishment's web software suite.

It is, of course, possible to maintain a spare sleeping server, which can be used in case of failure, or during maintenance operations. Switching between two Tomcat servers behind an Apache frontal is relatively easy, and this is the solution recommended by the ESUP-Portail consortium. However, this solution is not transparent for connected users: all valid tickets (especially TGCs and PGTs) are lost.

Another possibility is to store granting tickets (TGCs and PGTs) in a database. This is conceivable since in this case, switching from one CAS server to another would have a very limited effect on tickets (only STs and PTs would be lost), while preserving simplicity and thus performance.

8 What about CAS in the future?

The ESUP-Portail consortium has taken an active part in popularizing CAS, notably by distributing a CAS server quick-start, which allows any system administrator to setup and configure a CAS server in a few minutes.

We have every confidence in CAS. CAS has been adopted by the ESUP-Portail consortium as its SSO software, and will be deployed in the coming months in all those French Universities who choose the ESUP-Portail software. We strongly believe that it can become a standard.

References

- [1] Autorité de certification du CRU, in French, <http://igc.cru.fr>
- [2] Single Sign-On architectures, Jan de Clercq, RSA2003, November 2003, Amsterdam, http://www.rsaconference.com/rsa2003/europe/tracks/pdfs/implementers_w14_declercq.pdf
- [3] Persistent client state (HTTP cookies). http://wp.netscape.com/newsref/std/cookie_spec.html
- [4] Sun One Identity Server. <http://www.sun.com>
- [5] Microsoft .NET Passport: One easy way to sign in online. <http://www.passport.com>
- [6] ITS Central Authentication Service, <http://www.yale.edu/tp/cas/>
- [7] Linux-PAM: Pluggable Authentication Modules for Linux, www.us.kernel.org/pub/linux/libs/pam/Linux-PAM-html/
- [8] JASIG (Java Architectures Special Interest Group), Evolving portal implementations. <http://mis105.mis.udel.edu/ja-sig/uportal/>
- [9] Web Services, <http://www.w3.org/2002/ws/>
- [10] ESUP-Portail, <http://www.esup-portail.org>
- [11] CAS GenericHandler, <http://esup-casgeneric.sourceforge.net>
- [12] PhpCAS, <http://esup-phpcas.sourceforge.net>
- [13] The Horde Project, <http://www.horde.org>
- [14] Authentication and access control in Sympa mailing list server, Serge Aumont & Olivier Salaun, TERENA2004, June 2004, Rhodes, <http://www.sympa.org>
- [15] The Shibboleth Project, <http://shibboleth.internet2.edu/>

Acknowledgements

- Shawn Bayern and Drew Mazurek, for their great work on CAS.
- The ESUP-Portail SSO group for their feedback and contributions.

Pascal Aubry played with real-time systems at ECP until 1993. In the succeeding years, he worked at IRISA on the distribution of synchronous programs and received his Ph.D. in Computer Science in 1997. Currently at IFSIC, University of Rennes 1, he manages web-projects. He has been part of the ESUP-Portail project since its beginning in late 2002, working on web security (SSO, authorizations) and data storage.

Vincent Mathieu is in charge of network deployment and administration at University of Nancy 2. An LDAP expert, he also manages some internet services. He is the leader of the ESUP-Portail SSO group.

Julien Marchal is in charge of email services and other network-related web applications at University of Nancy 2. He is also part of the ESUP-Portail group, working on SSO and communication services, and is the leader of the ESUP-Portail uPortal group.

Information Security e-Learning - from Practice to Theory

Jorma Kajava*, Rauno Varonen**

* University of Oulu, Department of Information Processing Science, Linnanmaa, Oulu
P.O.Box 3000, FIN-90014 UNIVERSITY OF OULU, Finland
Jorma.Kajava@oulu.fi

** University of Oulu, Language Centre, Linnanmaa, Oulu
P.O.Box 7200, FIN-90014 UNIVERSITY OF OULU, Finland
rvaronen@cc.oulu.fi

Abstract

Focusing on security education, this paper describes an e-learning environment that has been constructed to increase information security awareness among employees of a Finnish telecommunications company. The system has already entered the testing phase, and the preliminary results are interesting – and somewhat unexpected: the designers' assumptions about the current knowledge level turned out to be too high. On the other hand, the design principle, based on making the components of the system as simple as possible, seems to have produced a system that delivers both functionality and stability.

Keywords: e-Learning, information security, information security awareness.

Povzetek

E-izobraževanje na področju varnosti podatkov – od prakse do teorije

Članek opisuje okolje za e-izobraževanje, ki smo ga razvili, da bi dosegli večjo ozaveščenost o problematiki varovanja podatkov med zaposlenimi neke finske telekomunikacijske firme. Sistem je že v fazi preizkušanja, prvi rezultati pa so zanimivi, vendar nekoliko nepričakovani: izkazalo se je namreč, da so razvijalci predpostavljali, da je raven znanja višja, kot je v resnici bila. Po drugi strani pa je osnovni pristop, ki temelji na kar se da enostavnih komponentah, pripeljal do sistema, ki je hkrati funkcionalen in stabilen.

Ključne besede: e-izobraževanje, varnost podatkov, ozaveščenost o varnosti podatkov.

1 Introduction

This paper discusses an e-learning based information security project carried out by a small organisation in the telecommunication service sector. Interestingly, the organisation has outsourced all software development work. All 500 employees on the payroll, mostly technically-oriented specialists, have the basic information processing skills required by their jobs, but only a few have a broader experience of the information sciences.

In a small organisation, it is relatively easy to construct a learning environment and test its functionality in practice. To be useful, the environment must be built in close collaboration with the intended users, i.e., the environment and its contents must be based on their opinions and experiences and meet their practical needs. In the present case, an extensive survey was conducted to find out the employees' wishes and needs regarding the environment [13].

Experiences gathered during this project form the basis for its adaptation to the university setting. The arrangement is advantageous, because development

work in a small organisation is fairly flexible, and experiences accumulate quickly. Universities, on the other hand, although constituting open learning environments, tend to suffer from a higher degree of bureaucracy, which may affect the design or implementation of the learning environment. Thus, a trial system tested in practice is an essential resource for planners and constructors [6].

The topic area, information security, was selected as it is rapidly becoming a key issue for businesses, institutes of education and society at large. Traditionally, the highlight has been on the technical aspects of information security, but during the past few years, human and organisational aspects have assumed an increasingly prominent role in discussions on security [1, 2, 3, 4, 19, 20, 21].

E-learning based information security education strives to raise the awareness level of all employees. The aim is to equip them with the necessary skills and knowledge to meet the challenges that they may have to face in their everyday work [12].

The ultimate purpose of this paper is exploring the use of an electronic learning environment to enhance information security awareness.

2 Methods and techniques used

This study was carried out in the target organisation in 2002. The immediate goal was to investigate the information security knowledge of the employees by means of questionnaires and interviews. The results were then used to create an educational programme to correct the perceived weaknesses. As teaching material, the programme utilised various organisational guidelines complemented by educational materials compiled at the University of Oulu.

In addition to the actual learning environment and its contents, the organisation has also implemented an automatic online assignment sheet for tracking and monitoring learning. This form was designed to automate the learning process such that anyone who possessed the necessary skills and knowledge, could take the test and – on successful completion – be automatically exempted from having to go through the learning material. The system automatically handles registration and also updates registry files, when students pass the test. In essence, the environment enables company employees to study at their own pace. What it still lacks is a rewarding system for employees who receive a good grade [14].

Research tends to progress from theory to practice. Having become familiar with information security from various perspectives including the user and end-user perspectives, we decided to go the other way – from practice to theory. Our starting point was that, within information security, relevant knowledge usually resides within the organisation in question. What an outsider, such as a consultant, can contribute is a model or a general framework for exploring, enhancing and utilising this knowledge. On this view, pertinent information that is possessed by company employees is collected and analysed by an outside consultant who introduces a theoretical framework for analysis and may also assist in the utilisation of the results of such analysis.

The practice to theory approach is also supported by the fact that all information security events comprise a variety of aspects, some of which are strongly in relief, while others can best be described as weak signals. Even these can be taken into account thanks to the increased computing power of modern com-

puters, which allows the unique features of each information security incident to be analysed in detail.

This study investigated the information security knowledge of different employee groups using a semi-structured theme interview. At the same time, we were able to establish which areas of information security knowledge needed improvement. This information formed the foundation for the design of the e-learning programme.

However, before delving into these issues, let us first discuss information security awareness in general.

3 Information security awareness

On the basis of a series of practical studies, we have come to the conclusion that there are several stages in how people respond to awareness. It seems that there are three stages of awareness that should be taken into account. We maintain that these stages constitute an implication relation, where – within practically every organisation – there are people at every stage, and the success of IT security awareness correlates with progress toward the next stage. The stages of IT security awareness are [7, 8, 9, 12, 22, 23]:

- (ii) drawing people's attention to security issues,
- (iii) winning user acceptance,
- (iv) getting users to learn and internalise the necessary information security activities.

The first stage includes drawing people's attention to information security-related issues and trying to catch their interest. The second stage involves user acceptance; having attracted the end-users' attention, it is important to get them to accept the organisational IT security policy. Finally, at the third stage, the end-users should have internalised the instructions they have received during their security-related education and should take corrective measures in accordance with the security policy. In this paper, the term 'awareness' includes all the aspects mentioned above.

IT security awareness should be comprehensive, well-organised and systematically executed from the start. In addition, the efficiency of all actions should be measured to ensure the on-going development of the organisational IT security awareness programme. As for our own IT security awareness programme in the university environment, we have come to the realisation that a wide range of tools and methods are necessary to implement security awareness for different people in different environments and at different stages of awareness. Security education is needed to

convince every user of the importance of following guidelines and to make them aware of the consequences of intentional violations of information security. Education is also needed to ensure that the achieved level of awareness (as defined above) will be maintained. Various awareness raising methods, such as campaigning and the so-called Hammer theory, are needed to provide incentives for end-users and to refresh the importance of these factors in the minds of people. And finally, awareness, comprising education and training, should ensure that people internalise security guidelines and abide by them in their daily work.

The awareness programme of any industrial organisation should follow the framework developed in Reference [7, 8]:

- (i) Identify programme scope, goals and objectives.
- (ii) Identify training staff.
- (iii) Identify target audiences.
- (iv) Motivate management and employees.
- (v) Administer the programme.
- (vi) Maintain the programme.
- (vii) Evaluate the programme.

The awareness programme should be targeted for at least four different groups. These are: top management, IS management, end-users and IT/IS specialists. Of course, there is no exact formal classification for these groups and, as a result, the end-user group, for example, remains relatively vague.

The IT security awareness programme should be implemented at all levels of the organisation, starting from the top management, who should be made aware of the need to establish and maintain an organisational security policy [7]. Then comes the creation of a security model including IT security policies, the allocation of responsibilities, etc. IT security awareness programmes are essential in keeping users in the "security team" and in ensuring the overall success of the organisation's security strategy. All evidence shows that to function satisfactorily, a security programme must find support in all parts of the organisation. Moreover, the top management has to accept the security policy wholeheartedly and allocate resources and appropriate financial support accordingly. As stated earlier, the effectiveness of the measures taken during an awareness programme should be evaluated as objectively as possible. The problem is that most organisations do not provide feedback or measure the success of their IT security awareness programme.

The security management should react to feedback and consider necessary improvements. Feedback should be based on organisational and end-user viewpoints and on the established results of particular security measures [8].

4 Security learning

Understanding information security issues from the technical point of view is an advantage that employees of the case company have [10, 15]. Nevertheless, since they do not have a wider perspective on other aspects of security, such as organisational or end-user related issues, they need information security training [11]. The problem is that, being small, the company does not have the resources to allow its personnel to take time off from work to participate in security training [16]. This paper seeks to answer the question of what technical tools the company could use in this situation.

One solution is to resort to e-learning and construct an online learning environment. Many e-learning environments are realized by long distance networks, via the Internet, but our solution was to build an intranet-based environment, within the company network. Most e-learning solutions consist of very sophisticated and complicated systems, filled with content that is more entertainment than work-oriented, but we proposed a solution that is both simple and practical.

In the long run, the project reported here aims to develop a five-level solution consisting of different guidelines custom-tailored for different groups. At the first stage of this research, the focus is on guidelines that apply to all user groups. First, a questionnaire on currently prevalent practices is sent to every group. Then, having analyzed the results, the most important guidelines are collected for organisational use using the e-learning environment. It is important that these guidelines are easy to understand and follow - and it would not hurt if they were presented in a humorous way [18].

In addition, it is vital that the information security education programme is automated and computer-supported as well as transportable to different environments, including those based on older systems. As a result, the e-learning environment can be transported to other organisations working in the same field. Moreover, as the basic guidelines pose no problems for employees with a deeper knowledge of the sub-

ject, they must have the option of going directly to the test part, without having to trudge through the entire programme.

5 Requirements for the learning environment

The starting point for this project was the fact the organisation under study is a profit-seeking commercial enterprise. Aiming at improving the security level of this organisation, the project also offers it a competitive edge through the provision of more secure telecommunication services. Technical solutions, although constituting the foundation of security, are insufficient and must be incorporated into a wider approach.

In a drive to promote information security across the organisation, the company initiated the e-learning project reported here. It seeks to find new, cost-effective, ways of offering security education to company employees. A guiding principle is this undertaking is that the education offered must be meaningful and immediately relevant to the employees.

In carrying out their everyday tasks, people tend to place a high value on usability, sometimes at the expense of security. Sadly enough, the significance of information security is often realized only after some mishap occurs.

Another balancing act is frequently observed in the context of e-learning [13]. Striving toward a more exciting and entertaining approach, educators sometimes lose sight of their original purpose, and become entertainers rather than educators. From the organisational viewpoint, the crucial question is which is more beneficial for the functionality of the organisation: a learning environment that is highly enjoyable or a learning environment that is highly usable and functional [5]?

The organisation under study here has posed a number of requirements for its new e-learning programme. For a start, it has to run on all computer platforms used within the company. In terms of structure, it has to be sufficiently simple to be reliable in all environments [16, 17].

6 Presentation and learning

Creating a multimedia e-learning environment requires not only technical and content-related expertise, but also a pedagogical advisor. Chief among the tasks of this advisor is to devise ways of presenting learning materials in a manner that enables learners

to assimilate new knowledge into their previous knowledge structures - and thereby understand what they have learned. Another function of the pedagogical advisor is to take account of different learning strategies and styles to maximise individual learning results. If no such advisor is available, and the design of learning materials is left to technical experts, for example, the various multimedia elements may have more entertainment than educational value [18].

All learners have their own learning strategies. Part of each individual's learning strategy is their learning style, which is an essential element of the learning process. We all have our own strengths, which we rely on when processing information. Some people are characterised as holistic, while others are best described as analytic learners. The difference lies in the way they tend approach a task; holistic learners immediately strive for the big picture, whereas analytically-oriented learners favour a piecemeal approach.

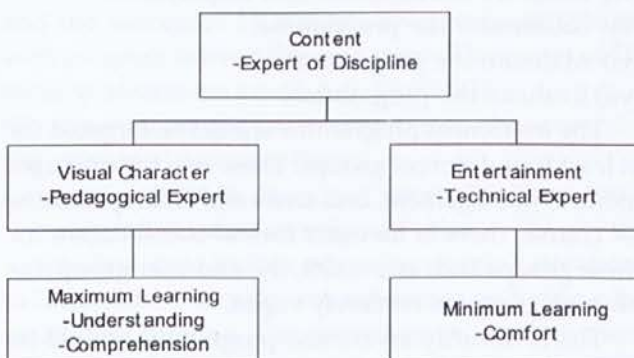


Figure 1. e-learning and multimedia.

Also our senses play an important part in the learning process. We receive and process information on the basis of our vision, hearing, tactile or kinetic sense. As a result, we have preferences as to how we want learning materials to be presented to us, how we want to see, hear, feel or experience the materials. Schools depend heavily on vision and hearing, at the expense of pupils who learn better by doing things, for instance. Some learners remember things as images, others as stories. A third group consists of learners who like to try things out through trial and error. Kinetically-oriented persons may memorise a telephone number or a melody, for example, as a sequence of finger movements, and learn certain things best while jogging or biking. Some people prefer to discuss things with other people, others teach themselves by

talking aloud. Most people have one or more preferred senses for receiving and processing information [16].

Learning styles based solely on one way of learning are very uncommon, as most people have their own learning strategies based on their strengths, habits and preferences. Consequently, a learning environment catering a large target group must be designed to accommodate a range of approaches and styles.

7 Significance of automated learning environment to the system

The purpose of the e-learning project was to construct a learning environment that does not require any additional hardware or software. In addition, the environment must be accessible from all workstations within the organisation [17].

For best possible functionality, the adopted solution seeks to minimise the effects of features that exist solely to increase the entertainment value of the environment. A major concern was the fact that an increased degree of automation inevitably leads to larger program code which, in turn, poses a threat to the stability and functionality of the entire system.

These two basic requirements, accessibility and full functionality from all workstations, relate to the levels of programming languages. It is a well-established truth that the lower the level of the used language, the faster the code and the smaller the memory requirement. Such code is also more secure. Nevertheless, implementing any system involves a compromise between automation and reliability, but the old adage "small is beautiful" is well worth bearing in mind.

8 Essential results

When we set out to design the e-learning environment, it was assumed that the basic information security guidelines of the organisation would be well-known by all employees. Therefore, it was a quite a revelation when the first tests in January 2004 indicated that some of the supposedly simple questions proved very hard to answer satisfactorily.

It was also revealed by the theme interviews that a great number of employees had no clear understanding of what information security is. All employee groups tended to describe it in terms of individual or isolated components. Moreover, about half of the interviewees could not explain in what ways information security issues would be relevant to their work.

This shows that security education should start by breaking down the definition of information security and analysing how it affects everyday work.

Feedback relating to the use of the e-learning system was mainly concerned with its technical implementation. Typical comments include "it is slow" and "it takes a long time to start". Generally, the learners either wanted the multimedia components to load quicker or they wanted more functionality, including muting or resolution changes on the fly.

The actual content matter of the programme was not commented on. What little feedback was received indicated that the intended practical approach was appreciated and that the learning topics were experienced as having a practical value. This relative lack of feedback may be explained by the fact that the learners did not have any expectations as to the content matter, since they were unfamiliar with the subject. Also, the design of content may have been better than the technical implementation. In general, the e-learning system was described as an interesting novelty and a number of learners indicated an interest to participate in similar training on other topics as well. With an average of 5 - 6 hours, many learners stated that they had used less time than expected on the tasks [6].

Measuring information security awareness is a difficult undertaking. One way of approaching it is to observe employees while they are working to establish the degree to which they follow the given guidelines. However, this study investigated the topic through an interview conducted among the learners. These interviews started by exploring how the learners understood the concept of information security which, after all, constitutes the foundation of information security awareness. The latter term refers to how well employees and members of society understand various information security threats and the related responsibilities. The results show that a high level of awareness has been achieved when all personnel understand the meaning of information security in its full extent and apply this knowledge in their work [6]. In addition, personnel must also be able to identify and manage a range of information security threats. Finally, they must also know what to do to avert these threats.

This brings us to the critical question: is it possible to achieve this goal using an online learning environment?

Answering that question involves separating two different kinds of responsibility. Technical issues and software concerns should be tackled by hardware and software manufacturers, while issues relating to users and organisations are the province of education and are best dealt with by universities. It then follows that universities have the responsibility to define the structure and content of information security studies. In a very broad sense, this involves taking responsibility for ensuring the wellbeing of future society, which is highly dependent on communication and computers.

9 Conclusions

This paper discussed an e-learning environment for information security education, designed and constructed by a small Finnish telecommunications company. The experiences gathered so far indicate that the implementation of an extensive learning system of this kind must be based on simple solutions that minimise system load.

It became clear during this study that, to be successful, e-learning requires that the designers and tutors are familiar with the learners' needs and learning styles. Diverse ways of presenting the learning materials makes it easier for individuals with different learning styles to take in the information. What renders the entire task more challenging is that the content matter of information security is often fairly abstract, highlighting the importance of careful design and presentation. The education offered must provide a range of possibilities for interaction, because the chosen medium, online teaching, markedly lessens personal contact among teachers and learners. Other important aspects include the provision of support to the learners and ways of creating an inspiring atmosphere conducive to learning. Feedback provided by the teacher is an integral part of learning, and its role is even more important in online teaching, where studies are usually conducted in (relative) isolation from other learners. Time must also be allocated to electronic communication between the participants. And finally, attention must also be given to developing the proficiency of the teachers and promoting their interaction.

An interesting – and quite unexpected – finding was that the planners' assumptions about the level of security knowledge among company employees was fairly low. In the information age, every citizen should have a basic understanding of information security,

otherwise they will be weak links in the chain, vulnerabilities that can be exploited. Increasing the level of security knowledge necessitates the implementation of various kinds of information security awareness programmes, an example of which is provided by the e-learning environment described here.

Acknowledgements

The authors wish to extend their thanks to Mrs Tiina Ramet, Mr Ilari Heikkinen and Mr Erkki Tuormaa, who have been responsible for the practical design and implementation of the e-learning system.

References

- [1] BS7799-1:fi. Standard. – Information Security Management. Part 1: Code of practice for information security management systems. Finnish standards association SFS, 15. February (1999).
- [2] BS7799-2:fi. Standard. – Information Security Management. Part 2: Specifications for information security management systems. Finnish standards association SFS, 15. February (1999).
- [3] A Code of Practice for Information Security Management, Department of Trade and Industry. DISC PD003. British Standard Institution, London, UK. (1993).
- [4] Computer Security Handbook, The Practitioners Bible Computer Security Institute. Mac/Donnel Printers, USA. (1984).
- [5] Epelboin, Y. : E-learning: putting documents On the web – Do and Don't. Workshop in the 8th Conference of European University Information Systems (EUNIS 2002). European University Information Systems (EUNIS) and University of Porto, Faculty of Engineering. June 19 - 22. Porto, Portugal. (2002).
- [6] Heikkinen, I., Ramet, T., "e-Learning as a part of information security education development from organisational point of view". Oulu University. Oulu. May (2004) (in Finnish).
- [7] ISO-IEC-27, Guidelines for the Management of IT Security (GMITS): Part 1 – Concepts and models for IT Security. (1994).
- [8] ISO/IEC JTC1/SC27, Guidelines for the Management of IT Security (GMITS). (1995).
- [9] Kajava, J. & Siponen, M., "Effectively Implemented Information Security Awareness - An Example from University Environment". In Jan HP Eloff and Rossouw von Solms, editors: *Information Security - from Small Systems to Management of Secure Infrastructures*. Proceedings of WG 11.2 and WG 11.1 of TC11 (IFIP TC-11 Sec'97, 13th International Information Security Conference). IFIP, 13 - 16th May, Copenhagen, Denmark. (1997).
- [10] Kajava, J., "IT Security Infrastructure – Opportunity or Threat for Future Society?" Politics & Internet. 2nd International Congress on Electronic Media & Citizenship in Information Society. The Finnish National Fund for Research and Development (SITRA) on the initiative of the Committee for the Future of the Finnish Parliament. 6-9 January, Espoo, Finland. (1999).

- [11] Kajava, J., Varonen, R., "Information Security Education: From the End-User Perspective to Public Administration Applications". In *Verwaltungs-informatik 2000: Verwaltungsinformatik in Theorie, Anwendung und Hochschulausbildung /3*. Internationale Fachtagungen "Verwaltungsinformatik" der Gesellschaft für Informatik (FTVI HBS 2000). Herausgegeben von Hans-Jürgen Lüttich und Claus Rautenstrauch. Otto-von-Guericke-Universität Magdeburg. 11 – 13th October. Halberstadt, Germany. - **mdv** Mitteldeutscher Verlag GmbH Halle (Saale), Germany. (2000).
- [12] Kajava, J., Varonen, R., "Incorporating Information Security into University Infrastructure". In Ligia Maria Ribeiro, Jose Marques dos Santos (eds): *The Changing Universities: The Challenge of New Technologies*. The 8th Conference of European University Information Systems (EUNIS 2002). Proceedings. European University Information Systems (EUNIS) and University of Porto, Faculty of Engineering. June 19 - 22. Porto, Portugal. (2002).
- [13] Kajava, J., Pyy, J., Tuormaa, E., Heikkinen, I., Mukari, J., Ramet, T., "Education material produced by TiKo project". Oulu Telecom Plc. May 16. Oulu, Finland. (in Finnish). (2003).
- [14] Kajava, J., Tuormaa, E., Ramet, T., Heikkinen, I., Kuusijarvi, T., Polvi, M., "Automated online assignment sheet for tracking and monitoring e-Learning. EFo program." Oulu Telecom Plc. November 20. Oulu, Finland. (in Finnish). (2003).
- [15] Kajava, J., "Information Security Challenges for Users, End-Users and Organizations in the Beginning of the new Millennium" (Abstract in English). *Administrative Studies Journal*. Volume 19, Number 2. Tampere, Finland. (2000).
- [16] Kajava, J., Varonen, R., Tuormaa, E., Nykanen, M., "Information Security Training through eLearning - Small Scale Perspective". In Eveline Riedling (ed.): *VIEWDET 2003*. Vienna International Conference on eLearning, eMedicine, eSupport. Vienna University of Technology. Nov. 26. - 28. Vienna, Austria. (2003).
- [17] Kajava, J., Varonen, R., "e-Learning as a Tool: Framework for Building an Information Security Awareness Programme for a Local Teleoperator". In Jeanne Schreurs and Rachel Moreau (eds.): *Euromedia'2004*. Tenth Annual Scientific Conference on Web Technology, New Media, Communications and Telematics Theory, Methods, Tools and Applications. Huize Corswarem. Hasselt, Belgium. April 19 - 21. A Publication of EUROISIS. Ghent, Belgium. (2004).
- [18] Neal, L., Perez, R., Miller, D., "eLearning and Fun". CHI'04 SIG. ACM. Vienna, Austria. April 26 - 29. (2004).
- [19] "The NIST handbook, An Introduction to Computer Security", NIST special publications. October. USA. (1995).
- [20] Parker, Donn B., "Computer Security Management". Prentice Hall, Reston, USA. (1981).
- [21] Royal Canadian Mounted Police, Security in the EDP Environment. Security Information Publication, Second Edition. Gendarmere Royale du Canada. Canada. (1981).
- [22] Thomson, M.E., von Solms, R., "An Effective Information Security Awareness Program for industry". In Jan HP Eloff and Rossouw von Solms, editors: *Information Security - from Small Systems to Management of Secure Infrastructures*. Proceedings of WG 11.2 and WG 11.1 of TC11 (IFIP TC-11 Sec'97, 13th International Information Security Conference). IFIP, 13 - 16th May, Copenhagen, Denmark. (1997).
- [23] Walsh, T., "Measuring the Effectiveness of Computer Security Training". 23th Annual Security Conference and Exhibition. CSI. November 11 - 13. Chicago, Il. (1996).

Jorma Kajava graduated from the University of Oulu in 1976 (M.Sc., Dept. of Technical Physics, Control and Systems Engineering) and completed his Licentiate Degree in 1978 (Faculty of Technology, Wireless Telecommunications, Adaptive Antennas). Since then he has been in the employ of the University of Oulu in various positions, including Lecturer, Head of Laboratory, Acting Associate Professor and Acting Professor. His teaching experience includes not only the University of Oulu, but also the University of Lapland, the Raahe and Oulu Polytechnics as well as other institutes of education. Aside from his involvement in numerous international projects (including the European Commission's Information Society Technologies (IST) Programme), he has found the time to write c. 350 articles, published in a variety of formats such as conference proceedings, journals, research reports, electronic publications, educational support materials and newspaper articles. Among his research interests are computer ethics, information security and e-learning.

Rauno Varonen has worked at the University of Oulu since his graduation in 1985 (M.A, Faculty of Humanities). He did his teacher training in 1985-1986 and completed his Licentiate Degree in 1996. Working as a Lecturer at the university's Language Center, he has a wide teaching experience. His publications include teaching materials and academic papers. One of his interests is e-learning, which he approaches from the practitioner's viewpoint. He is also a distinguished member of the Black Hole.

Portal Univerze v Mariboru

Milan Ojsteršek*, Dejan Dinevski**, Tomaž Klojčnik**

*Fakulteta za elektrotehniko, računalništvo in informatiko, Smetanova 17, 2000 Maribor

**Univerza v Mariboru, Slomškov trg 15, 2000 Maribor

ojstersek@uni-mb.si, dejan.dinevski@uni-mb.si, tomaz.klojcnik@uni-mb.si

Povzetek

Na Univerzi v Mariboru smo vzpostavili portal, ki omogoča zaposlenim in študentom dostop do različnih vsebin, njenega informacijskega sistema in podporo izobraževanju na daljavo. Do storitev portala lahko zaposleni in študenti dostopajo prek različnih elektronskih medijev (internet, mobilni telefoni, ki omogočajo WAP, infoterminali, dlančniki). Portal je integriran z informacijskim sistemom Univerze v Mariboru in omogoča upravljanje z dokumenti, učinkovito iskanje informacij, integracijo storitev in vsebin, predstavitev informacij v več jezikih, avtentikacijo in avtorizacijo dostopa do storitev in personalizacijo z vidika vsebine ter vizualnega izgleda. Portal smo dopolnili še s podpornim sistemom za izobraževanje na daljavo, ki omogoča študentom lažji dostop do študijskih gradiv, elektronsko oddajo nalog in elektronsko obveščanje o rezultatih opravljenih pedagoških obveznosti. Portal služi tudi za sistem komuniciranja med zaposlenimi in študenti. Podporni sistem za izobraževanje na daljavo vsebuje tudi elektronsko knjižnico učnih materialov, slovar izrazov, podporni sistem za učitelja, podporni sistem za administriranje portala in podporni sistem za študenta.

Abstract

The Portal of the University of Maribor

University of Maribor has established a portal, which enables its students and employees to access to its information system, contents and distance learning support. Basic features of the portal are document management, search facility, integration of contents and services, information categorization, independency of data from different user interfaces and electronic media, multi-lingual support, authorization of access to data and services, contents and user interface personalization. The portal serves as a tool for communication among employees and students. The portal also includes a distance learning support service, which is integrated with the University of Maribor information system. The distance learning service consists of lecturer support, student support, administration support, digital library, and electronic dictionary parts.

1 Uvod

Univerze po svetu se zmeraj bolj usmerjajo v informacijsko podporo izobraževanju in nudenju storitev izobraževanja na daljavo. Na ta način olajšamo študij »klasičnim« študentom, lahko pa omogočimo študij tudi precej večjemu številu ljudi. Na Univerzi v Mariboru smo se odločili za izdelavo lastnega portala, ki omogoča dostop do informacijskega sistema Univerze in vsebin, ki jih pripravljajo posamezne fakultete ter rektorat (novice, obvestila, objava rezultatov pedagoških obveznosti). Portal smo dopolnili še s podpornim sistemom za izobraževanje na daljavo, ki omogoča študentom lažji dostop do študijskih gradiv, elektronsko preverjanje in samo-preverjanje znanja, elektronsko oddajo nalog in elektronsko obveščanje o rezultatih opravljenih pedagoških obveznosti. Vzpostavitev podpornega sistema za e-izobraževanje je podprlo Ministrstvo za informacijsko družbo, razvoj programskega okolja pa je s ciljnim raziskovalnim projektom podprlo Ministrstvo za šolstvo, znanost in šport. Portal uporablja tudi storitve informacijskega sistema univerze (ISUM) [1], kar omogoča zaposlenim dostop do finančnih, kadrovskih in pedagoških podatkov. Študenti pa se lahko s pomočjo storitev

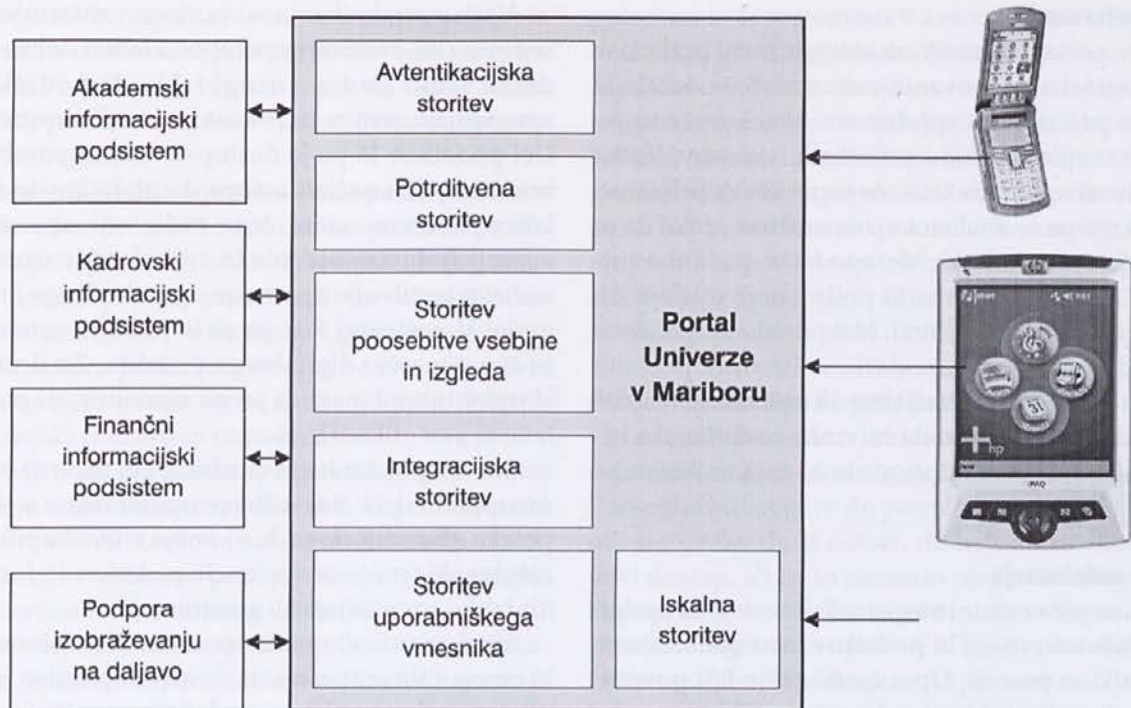
ISUM elektronsko prijavijo ali odjavijo od izpitov, pregledujejo podatke o svojih ocenah, dobivajo SMS sporočila o rezultatih opravljenih pedagoških obveznosti itd.

V drugem poglavju smo predstavili tehnološko realizacijo portala. Informacijski sistem Univerze v Mariboru je predstavljen v tretjem poglavju. V četrtem poglavju smo predstavili lastnosti okolja za podporo e-izobraževanja. Na koncu članka smo podali zaključne misli in napotke za nadaljnje delo.

2 Tehnološka realizacija portala

Portal predstavlja povezovalni člen med vsebinami in storitvami, ki jih nudijo univerza in fakultete (slika 1). Omogoča:

- upravljanje z dokumenti,
- iskanje po vsebinah,
- integracijo storitev ISUM, podpornega okolja za e-izobraževanje in vsebin,
- predstavitev vsebin na različnih elektronskih medijih (internet, mobilni telefoni, infoterminali, dlančniki, v prihodnosti tudi govorni uporabniški vmesnik),



Slika 1: Shematski prikaz informacijskega sistema Univerze v Mariboru

- predstavitev informacij v tujih jezikih,
- avtentikacijo in avtorizacijo dostopa do podatkov in storitev ter zagotavljanje varnosti,
- posebitev z vidika vsebine in vizualnega izgleda in
- pošiljanje sporočil po elektronski pošti ali s pomočjo SMS sporočil.

Vsebine predstavljajo različna besedila in večpredstavniki (slike, video posnetki, zvočni posnetki, animacije, predstavitve v MS Powerpointu, MS Excelove preglednice ...). S pomočjo storitev je možno kreiranje dinamične vsebine (npr. storitve omogočijo študentu prijavo na izpit) ali opravljanje določenih nalog (npr. obveščanje študentov o opravljenih vajah prek SMS). Portal shranjuje vsebine in storitve v hierarhično strukturo področij, ki jih lahko administrator tudi navzkrižno povezuje.

Upravljanje z dokumenti

Upravljanje z dokumenti omogoča vstavljanje, spreminjanje in brisanje besedilnih dokumentov, slik, zvoka, animacij ali videa. Vsaka sprememba dokumenta se beleži kot nova verzija dokumenta. S tem omogočimo uporabniku vpogled v zgodovino spreminjanja dokumentov. Po potrebi si lahko uporabnik (lastnik

dokumenta) starejšo verzijo dokumenta tudi prenese iz strežnika. Aplikacija, ki informacijsko podpira vstavljanje dokumentov, beleži tudi, kdo in kdaj je spreminjal določen dokument. Portal omogoča tvorjenje in spreminjanje hierarhične strukture področij, na katera lahko vstavljamo vsebine in storitve. Omogoča tudi avtomatsko brisanje vsebin, ki niso več aktualne, in urejanje vsebin na enak način, kot je to možno v MS Wordu. Možno pa je tudi urejanje v obliki zapisa v jeziku HTML (HyperText Markup Language). V besedilna okna lahko dodajamo tudi slike, video, zvok, animacije, java programe, flash datoteke in hiperpovezave. Vsebinsko lahko v tekstovna okna prenašamo tudi iz drugih programov (npr. MS Word, MS Excel) ali iz drugih spletnih strani na principu »povleci – spusti« (Cut and Paste).

Sistem za upravljanje z dokumenti omogoča:

- sočasno spreminjanje dokumenta enemu samemu uporabniku (check in/out dokumenta),
- spremljanje verzij dokumenta,
- zagotavljanje varnosti oz. omogočanje dostopa do dokumentov samo določenim uporabnikom oz. skupinam uporabnikov, ki imajo ustrezno definirane pravice dostopa,
- spremljanje, kdo in kdaj je spreminjal določen dokument.

Iskanje informacij

Iskanje v portalih temelji na kategorizaciji podatkov na podlagi tako imenovanih metapodatkov. Iskalniki običajno preiskujejo spletne strani rekurzivno po principu iz spletne strani izvlečenih naslovov. Vsaka raziskana stran se indeksira, če je potrebno, primerno uteži, hkrati pa se analizira njena vsebina – tako da se ustvarjajo metapodatki. Metapodatke pa lahko pridobimo tudi iz šifrantov, ki podrobneje opišejo dinamično ustvarjeno vsebino. Naš portal omogoča oba načina določanja metapodatkov. Iskalnik je tesno povezan s storitvijo poosebitve in s storitvijo za dodelitev dostopa do portala in vrača podatke, do katerih lahko uporabnik dostopa in so zanj najbolj relevantni.

Storitve sodelovanja

Portal omogoča nabor integriranih storitev, ki podpirajo pretok informacij in podatkov med aplikacijami, uporabniki in procesi. Uporabniki želijo biti povezani med seboj. Pridobljene informacije želijo izmenjevati s sodelavci, na njihovi podlagi sprožiti akcijo itd. Portal omogoča skupinsko delo ter tako podpira sodelovanje med uporabniki v realnem času, da le-ti enostavno poiščejo sošolce ali profesorja ter z njim komunicirajo ne glede na to, kje se nahajajo. Te storitve omogočajo uporabnikom obveščanje o dosegljivosti sodelujočih v procesu učenja, izmenjavo dokumentov in aplikacij med njimi, pogovor idr. Zelo koristna je tudi zbirka nosilcev strokovnih znanj, v kateri lahko uporabniki poižvedujejo po osebah, znanjih, sposobnostih, izkušnjah, projektih, izobrazbi, delovnih mestih in drugih atributih. Tako zlahka pridejo do oseb, ki poznajo odgovor na njihovo vprašanje. Portal omogoča sodelovanje in upravljanje z dokumenti (Collaborative authoring tools), klepetalnico, elektronski rokovnik (web organizer) in forum.

Avtorizacija, avtentikacija in zaščita dostopov do podatkov in storitev

Informacije v portalu morajo biti zavarovane pred nepooblaščenim dostopom, hkrati pa morajo omogočati enostaven in hiter dostop. Portal je integriran z avtentikacijskim in avtorizacijskim sistemom informacijskih storitev, ki jih omogoča informacijski sistem univerze. S strani uporabnika je omogočena le ena prijava v portal, pri tem pa storitev upravljanja dostopa do informacij zagotavlja dostop do vseh relevantnih informacij, do katerih ima uporabnik pooblastilo.

Vsak uporabnik ima svojo vlogo v informacijskem sistemu (IS). Nekateri uporabniki lahko določene podatke samo gledajo, drugi lahko te podatke tudi spreminjajo, tretjim pa je dostop do njih onemogočen. Del podatkov IS pa je dostopen vsem uporabnikom interneta. Tak način dostopa do podatkov in funkcij lahko dosežemo samo, če se mora vsak uporabnik IS avtentificirati. Za študente in zaposlene je uporabljen način avtentikacije s pomočjo uporabniškega imena in gesla. V naslednji fazi pa se bo možno avtentificirati tudi s pomočjo digitalnega potrdila. Za dostop do storitev in vsebin, ki so javno dostopne, ni potrebno izvesti avtentikacije.

Ko se uporabnik avtentificira, ga IS razvrsti v skupino uporabnikov. Razvrščanje uporabnikov v skupine poteka glede na vlogo, ki jo imajo v IS. Skupini uporabnikov IS omogoča dostop do podatkov in funkcij, ki jih lahko uporabljajo ali spreminjajo.

Implementirali smo lastno avtentikacijsko storitev, ki omogočala centralizacijo osnovnih profilov uporabnikov z avtentikacijskimi podatki in zagotavlja poenoten sistem avtentikacije tako, da ga je možno enostavno integrirati v katerikoli informacijski sistem. Osnovne lastnosti avtentikacijskega sistema so:

- vsak uporabnik ima osnovni profil s svojimi osebniimi podatki,
- uporabnik definira pravice dostopa spletnih aplikacij do podatkov osebnega profila,
- sistem ob prijavi posreduje globalni identifikator uporabnika,
- poenotena avtentikacija,
- enkratna prijava v vse informacijske sisteme,
- enkratna odjava iz informacijskih sistemov,
- globalna terminacija uporabniškega profila,
- varna izmenjava podatkov med avtentikacijskim sistemom in informacijskimi sistemi,
- enostavna integracija avtentikacijske storitve v informacijske sisteme.

Osnovni profil hrani avtentikacijske podatke, kot so uporabniško ime in geslo, skrito vprašanje in skrit odgovor, ki lahko služi za primere, ko uporabnik pozabi geslo, in v primerih, kjer je potrebna dodatna avtentikacija in je treba poleg gesla vnesti še skrit odgovor. Pomemben podatek je identifikacijska številka uporabnika, ki služi za ustvarjanje posebnih profilov za potrebe posameznih spletnih storitev, kot so poosebitev izgleda oz. funkcionalnosti portala Univerze. Osebni podatki v profilu so ime, priimek, datum rojstva, spol, naslov prebivališča s pošto in državo, EMŠO,

jezik, telefonska številka v stacionarnem omrežju in GSM, naslov elektronske pošte in naslov domače spletne strani. Podatki niso obvezni za vpis, razen avtentikacijskih podatkov (to sta v našem primeru uporabniško ime in geslo).

Uporabnik definira pravice dostopa do podatkov v profilu tako, da določi, ali se lahko podatek profila pošlje dalje. Uporabnik lahko dovoli pošiljanje podatkov iz svojega profila, prepove ali dovoli pošiljanje podatkov iz svojega profila samo v primeru, da informacijski sistem podatke potrebuje (avtentikacijski sistem ima informacijo o tem, katere podatke določen informacijski podsistem potrebuje).

Globalni identifikator se ob avtentikaciji vedno posreduje informacijskemu sistemu, ki ga ta uporabi za lokalno identifikacijo in za kreiranje lastnih, specifičnih profilov v namen poosebitve vizualnega izgleda ali storitev informacijskega sistema.

Poenotena avtentikacija omogoča, da se vse prijave v vse sisteme izvajajo prek avtentikacijske storitve. Spletne aplikacije v tem primeru ne potrebujejo lastnih avtentikacijskih obrazcev.

Enkratna prijava/odjava omogoča, da se uporabnik avtentificira v avtentikacijsko storitev enkrat, nakar se pri obisku drugih sistemov namesto uporabnika avtentificira avtentikacijska storitev, razen v primerih, če uporabnik poenostavljene prijave ne želi uporabljati oz. če informacijski podsistem zahteva uporabnikovo avtentikacijo (samo v primeru, ko gre za varnostno kritične vire). Odjava se sproži pri avtentikacijski storitvi, ki poskrbi za lokalno odjavo pri informacijskih sistemih.

Globalna terminacija uporabniškega profila se proži pri avtentikacijski storitvi, ki avtomatsko obvesti vse informacijske sisteme, da uporabnik zahteva izbris profilov, vezanih nanj.

Varna izmenjava podatkov med avtentikacijsko storitvijo in informacijskimi sistemi je zagotovljena z uporabo šifrirnih algoritmov, ki so zasnovani na sistemu javnih in privatnih ključev, med uporabnikom in avtentikacijskim sistemom pa z uporabo varnih komunikacijskih kanalov, kot je SSL.

Enostavna integracija je implementirala v obliki modula, ki ga informacijski sistem uporabi v lastni implementaciji. Konkretno so pri razvoju ali nadgradnji obstoječih informacijskih sistemov razvijalci dobili paket, ki vsebuje programske module in navodila za integracijo in uporabo modula, ki komunicira z avtentikacijsko storitvijo. Posamezen informacijski sistem je

potrebno le še registrirati v avtentikacijskem sistemu in izdelati javni ter privatni ključ, ki služi za šifriranje podatkov. S tem skrajšamo čas in stroške razvoja posameznega informacijskega sistema, hkrati pa povečamo varnost zasebnih podatkov, saj so občutljivi podatki skrb enega sistema ali manjšega števila sistemov. Tak avtentikacijski sistem lahko kasneje izpopolnimo, kar poveča varnost in vrednost celotne mreže storitve in informacijskih sistemov, čeprav ostalih informacijskih sistemov ne bi direktno modificirali.

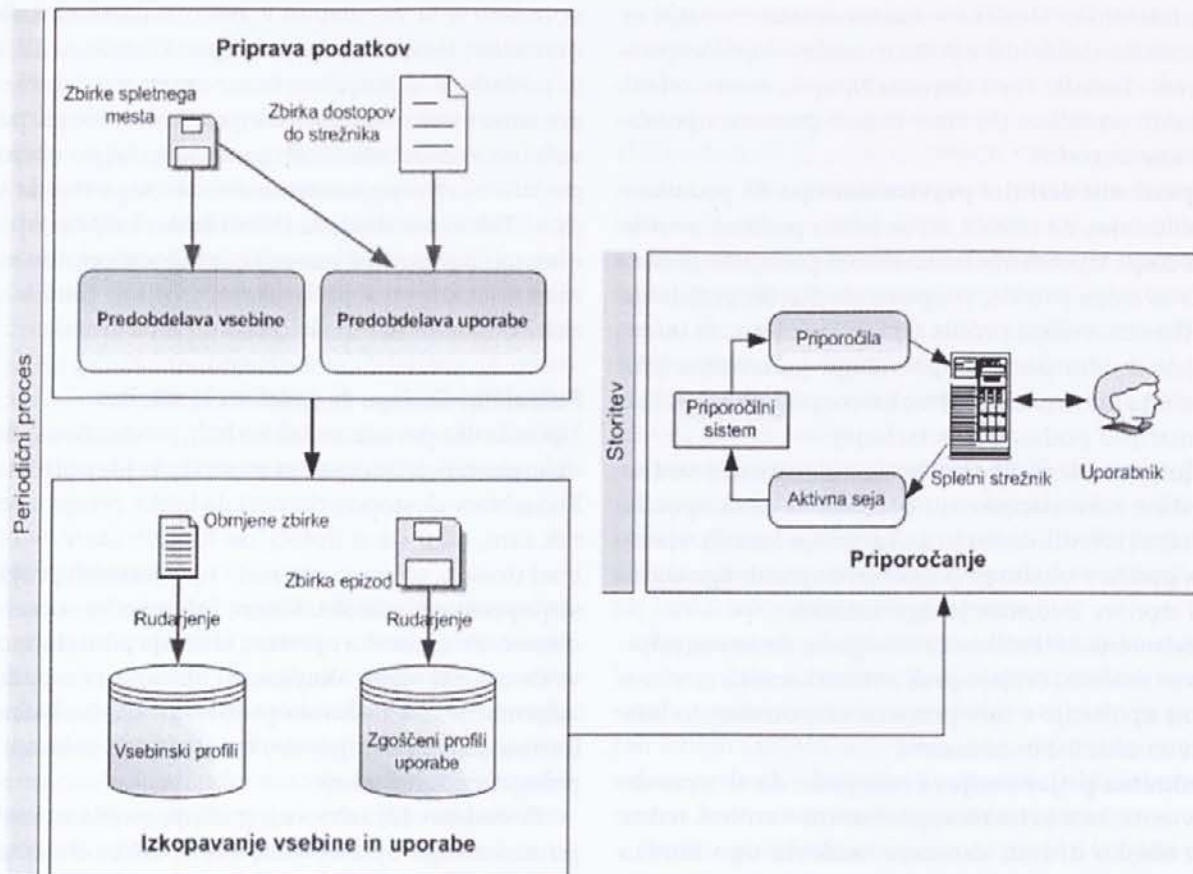
Poosebitev dostopa do podatkov in storitev

Uporabniki portala so lahko bolj produktivni, če jim daje samo tiste vsebine in storitve, ki jih potrebujejo. Poosebitev dostopov do portala lahko rešuje uporabnik sam, tako da si določi, do katerih storitev bi rad imel dostop, ali pa to namesto njega naredi programska oprema, ki določa, katere informacije so zanj zanimive. Programska oprema, ki izvaja poosebitev, razvršča uporabnike v skupine, ki jih zanimajo podobne informacije. Za vsako skupino nato določa, katere informacije so za te uporabnike zanimive in zanje tudi prilagaja pogled nanje.

Poosebitev [2] zahteva izgradnjo profila za vsakega posameznega uporabnika. Profil lahko določimo z uporabnikovim aktivnim sodelovanjem, lahko pa ga zgradimo ali dopolnimo in posodabljammo s pasivnim spremljanjem njegove aktivnosti na spletnem strežniku. Naš portal uporablja oba pristopa, saj imata oba svoje dobre (pa tudi slabe) strani. Prvi je sicer subjektiven, vendar omogoča, da uporabnik eksplicitno pove, kaj ga zanima, drugi pa se časovno spreminja.

V splošnem so v procesu poosebitve (slika 2) potrebni trije koraki: priprava in pretvorba podatkov, odkrivanje vzorcev in priporočanje. Priprava podatkov je prva in bistvena faza, ki omogoča kasnejše odkrivanje znanja in analizo korakov. Zanesljivost zanimivih vzorcev, ki jih pridobimo prek algoritma za odkrivanje znanja, je močno odvisna od kakovosti podatkov.

Učinkovitost priporočilnega sistema zagotovimo z izvajanjem v realnem času. V našem primeru se priporočanje izvaja v realnem času, medtem ko se fazi priprave podatkov in odkrivanja vzorcev izvajata s pomočjo opravil v ozadju. Za priporočanje potrebujemo sistem za delo z vsebino dokumentov, ki omogoča razvrstitev dokumentov po tematskih kategorijah in mehanizem za sledenje posameznemu uporabniku, ki izhaja iz naše komponente za beleženje



Slika 2: Proces posebitve

uporabnikovih akcij. Naloga priporočilnega sistema je, da izračuna množico priporočil, ki najbolj ustrezajo profilu trenutnega uporabnika. Poznamo dolgoročni in kratkoročni način priporočanja. V primeru, ko nimamo na voljo razpoznavnega mehanizma, gre za kratkoročno spremljanje. Takšnega uporabnika sledimo s pomočjo fiksnega pomičnega okna. Velikost okna je ponavadi zelo majhna, ker uporabnik svoje navade pogosto hitro spreminja. Sledimo torej samo kratkoročni zgodovini. Gre za učinkovit in časovno kot tudi sicer učinkovit način, ko drugi pristopi zatajijo.

Proces klasične posebitve uporabnikov smo v integracijski storitvi nadgradili še s posebitvijo uporabniškega vmesnika [3]. V ta namen sledimo uporabnikovim akcijam na samem uporabniškem vmesniku. Za vsak posamezni uporabniški vmesnik beležimo obnašanje uporabnika. Ob ponovnem prikazu istega vmesnika se ta uporabniški vmesnik lahko prikaže

prilagojen za posameznega uporabnika, kar pomeni, da za določenega uporabnika ne prikažemo določenih gradnikov oziroma se spremeni vrstni red prikaza gradnikov.

Na našem portalu smo združili oba klasična pristopa priporočanja, tj. s pomočjo preostalih uporabnikov oz. s širjenjem "od ust do ust" (collaborative filtering; CF) in s pomočjo vsebine medijev, ki ustrezajo uporabniškemu profilu (content-based approach) [5].

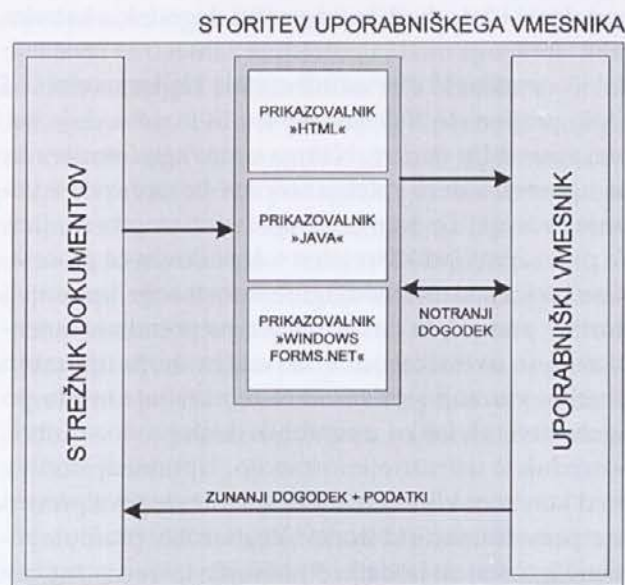
Pošiljanje sporočil po elektronski pošti in prek SMS

Določene podatke uporabniki po predhodni naročitvi prejema po elektronski pošti (statistike, potrditev prijave na izpit, rezultati pisnih izpitov, kolokvijev ...) ali na mobilni telefon (pošiljanje SMS sporočil). Del podatkov portal avtomatsko pošilja tistim, ki so upravičeni do teh podatkov (npr. podatki o rezultatu pisnega dela izpita in datum ustnega zagovora se bodo po vnosu takoj poslali študentu). Nekatera obvestila

so uporabnikom poslana samodejno, na osnovi njihovih vlog v okviru sistema, druga obvestila so uporabnikom poslana na njihovo željo. Seznam izbirnih obvestil je zabeležen v profilu vsakega uporabnika, kjer je zabeleženo tudi, kako želi prejemati obvestila. Običajni medij za pošiljanje obvestil je elektronska pošta, alternativni medij je SMS. Uporabnik lahko za sprejem izbirnih obvestil izbere en medij ali oba.

Predstavitev vsebin v različnih elektronskih medijih

Neodvisnost vsebine od njene predstavitve zagotavlja storitev uporabniškega vmesnika. Uporabniški vmesnik lahko izrišemo za različne platforme in za različne medije. Groba arhitektura storitve uporabniškega vmesnika je razvidna iz slike 3.



Slika 3: Arhitektura storitve uporabniškega vmesnika

Izdelali smo lasten jezik za opis uporabniškega vmesnika [3]. Zgledovali smo se po jeziki User Interface Markup Language (UIML) [6] in Microsoft Extensible Application Markup Language (XAML) [7]. Oba jezika imata skupno pomanjkljivost in sicer ne upoštevata lastnosti, da je potrebno z enim opisom uporabniškega vmesnika omogočiti prikaz uporabniškega vmesnika na katerikoli platformi in kateremkoli mediju. XAML je vezan na Microsoftovo platformo .NET in predvsem na en medij. UIML pa ima težavo, ker je preveč vezan na programski jezik Java, saj se pri definiciji gradnikov uporabniškega vmesnika sklicuje na imena predefiniranih java razredov. Strežnik dokumentov

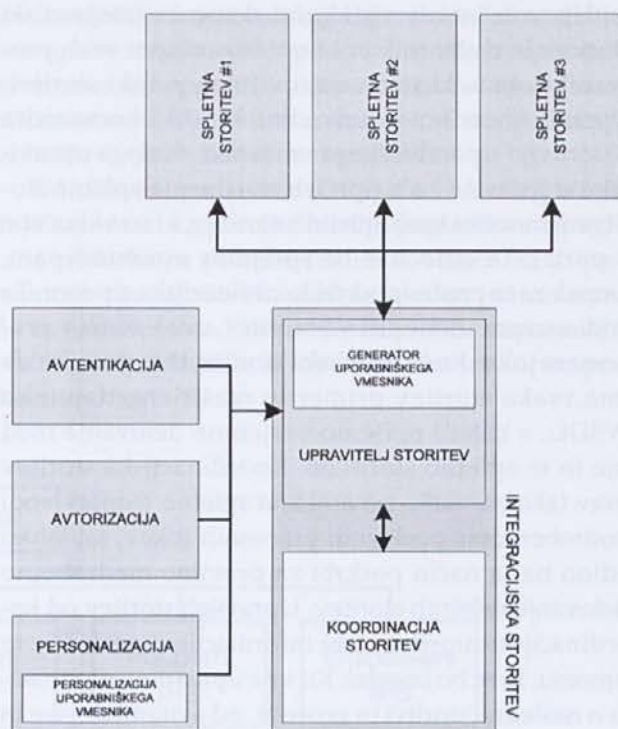
sprejema XML dokumente, ki so opisani v našem jeziku za definiranje uporabniškega vmesnika in jih preslika v ustrezen uporabniški vmesnik na ustrezni platformi. Prikazovalnik prejme od strežnika dokumentov dokument XML. Najprej v opisu preveri, za katero platformo se mora tvoriti uporabniški vmesnik. Če obstaja ustreza vrsta prikazovalnika, strežnik dokumentov preusmeri dokument XML k ustreznemu prikazovalniku.

Razširljivost storitve uporabniškega vmesnika se kaže ravno v tem, da je posameznih prikazovalnikov lahko neomejeno število. Slaba lastnost takšnega pristopa pa je ta, da je treba za vsako platformo in vsak medij posebej programirati ustrezen prikazovalnik. Ko izdelamo novi prikazovalnik, ga registriramo v centralni prikazovalnik.

Integracija storitev in vsebin

Posamezne storitve portala ne morejo delovati samostojno, zato potrebujemo storitev, ki omogoča integracijo storitev v portal.

Iz slike 4, na kateri je prikazana arhitektura integracijske storitve, lahko vidimo, da so namenskim spletnim storitvam (storitve iz informacijskega sistema



Slika 4: Arhitektura integracijske storitve

Univerze in storitve portala za izobraževanje na daljavo) poleg integracijske storitve dodane še avtentikacijska, avtorizacijska in personalizacijska storitev. Dostop do posameznih spletnih storitev je lahko omejen z uporabniškim imenom in geslom ali digitalnim potrdilom, kar zahteva avtentikacijo in avtorizacijo. Posamezne storitve lahko nudijo tudi informacije, vezane na posameznega uporabnika. V ta namen smo dodali že obstoječo rešitev personalizacijske storitve in jo razširili s personalizacijo uporabniškega vmesnika. Naloga upravitelja storitev je, da na eni strani preveri uporabnika in njegove pravice ter ugotovi njegov interes za uporabo posamezne storitve. Na drugi strani pa mora najti pravo storitev oziroma zagotoviti ustrezen tok dogodkov v sami storitvi. Naloga koordinacijske storitve je povezava več storitev v procesni tok. Za opis procesnih tokov smo uporabili jezik Business Process Execution Language (BPEL) [8]. Tudi ta jezik temelji na tehnologiji XML. S pomočjo jezika BPEL opišemo koordinacijo spletnih storitev in procesnih tokov. WSDL vmesnik definira specifično uporabo dovoljenih operacij spletne storitve, medtem ko BPEL definira njihovo zaporedje uporabe. WSDL opisuje javne vhodne in izhodne točke za vsak BPEL proces ter vse informacije o podatkovnih tipih, ki jih potrebujemo pri zahtevah med procesi.

Upravitelj storitev je ključni element v integracijski storitvi. Je nadzornik in hkrati organizator vseh procesnih tokov, ki se izvajajo v integracijski storitvi. Upravitelj storitev je tudi edini, ki lahko komunicira s storitvijo uporabniškega vmesnika. Naloga upravitelja storitev se začne pri izbiri ustrezne spletne storitve v množici vseh spletnih storitev, ki so vključene v portal. Te odločitve ne sprejema upravitelj sam, ampak za ta proces poskrbi koordinacijska storitev. Ta vodi seznam vseh spletnih storitev, udeleženih v procesnem toku. Koordinacijska storitev tudi poskrbi, da ima vsaka storitev primerno razširjeno datoteko WSDL, s katero opišemo vzajemno delovanje med njo in to spletno storitvijo. Koordinacijska storitev prav tako za vsako posamezno spletno storitev vodi podroben opis poslovnih procesnih tokov, saj lahko edino na ta način poskrbi za pravilno medsebojno delovanje spletnih storitev. Upravitelj storitev od koordinacijske storitve dobi informacijo o naslednjem procesu, ki se bo izvedel. Ko ima upravitelj informacijo o naslednji storitvi in procesu, od avtentikacijske in avtorizacijske storitve dobi podatke, ki jih posreduje storitvi. V odvisnosti od avtorizacijskih podatkov splet-

na storitev vrne ustrezen odgovor. Dodatno lahko upravitelj storitev doda storitvi še podatke o posebitvi za določenega uporabnika, v primeru da storitev podpira posebitev. Na podlagi vseh teh vhodnih podatkov se lahko izvede klic procesa. Ko storitev vrne odgovor, upravitelj storitev pokliče storitev uporabniškega vmesnika. Njegova naloga je, da za določen proces na podlagi vhodnih parametrov opiše uporabniški vmesnik v primerni obliki. Storitve uporabniškega vmesnika in integracijska storitev sta povezani tako, da je upravitelj storitev iz integracijske storitve, povezan s strežnikom dokumentov iz storitve uporabniškega vmesnika. Ti dve storitvi si izmenjujeta zahteve prek spleta po naši lastni XML shemi (podrobnejši opis storitve uporabniškega vmesnika je podan v prejšnjem poglavju).

Kako poteka komunikacija med uporabnikom in portalom? Uporabnik mora prožiti dogodek, s katerim prek izbranega medija posreduje zahtevo za izvajanje določene storitve. To zahtevo, ki je zapisana v obliki XML, prejme *strežnik dokumentov* in jo posreduje naprej *upravitelju storitev*. Naloga upravitelja storitev je, da ugotovi, katera spletna storitev bo prevzela izvajanje procesa. To nalogo izvaja *koordinacijska storitev*. Ta pravzaprav poišče celoten tok poslovnega procesa, ki se bo izvajal. Na podlagi te informacije upravitelj storitev preveri, če naslednji proces potrebuje avtentikacijo in avtorizacijo. V ta namen mora uporabiti posebni zunanji storitvi, ki to ugotavljata in mu po ugotovitvi, ali lahko uporabnik dostopa do storitve, posredujeta ustrezno informacijo. Upravitelj storitev pred končnim klicem metode spletne storitve preveri še v personalizacijski storitvi, če morda v profilu uporabnika obstajajo podatki o posebitvi, ki so vezani na trenutno izvajajočo se storitev. S temi podatki in podatki o procesu storitve lahko sedaj izvedemo klic metode spletne storitve. Upravitelj storitev vodi za vsak proces posebej tudi podatke o uporabniškem vmesniku. V primeru, da nimamo vnaprej predvidenega uporabniškega vmesnika za določen proces, ga na podlagi vrnjenih rezultatov in seznama vhodnih podatkov avtomatično tvori upravitelj storitev (tvori dokument XML, opisan po naši shemi). Upravitelj storitev posreduje dokument XML prek spleta storitvi uporabniškega vmesnika. Tam ta dokument prejme *strežnik dokumentov*, ki dokument posreduje *centralnemu prikazovalniku*. Centralni prikazovalnik iz opisnega jezika razbere, na kakšni platformi se mora prikazati in v svojem seznamu posebnih prikazovalnikov poišče ustreznega. Če je platforma neznana oziroma

centralni prikazovalnik ne najde primerne prikazovalnika, potem javi napako. V nasprotnem primeru opisni dokument XML posreduje posebnemu prikazovalniku. Prikazovalnik izdelava uporabniški vmesnik, ki je posredovan uporabniku. Notranji dogodki se izvajajo interno na uporabniškem vmesniku. Zunanji dogodki pa se posredujejo strežniku dokumentov. Dogodki se posredujejo zopet v posebno oblikovanem dokumentu XML. V ta dokument se zabeležijo tudi vse spremembe vrednosti gradnikov. Strežnik dokumentov ta dokument XML posreduje nazaj integracijski storitvi. Tam dokument prevzame zopet upravitelj storitve. Iz dokumenta razbere, kateri dogodek se je prožil. Zopet od koordinacijske storitve prejme podatke o tem, za katero preslikavo dogodka v poslovni proces gre. Tukaj se proces delovanja upravitelja zopet ponovi in s tem se zaključi krog delovanja.

3 Informacijski sistem Univerze v Mariboru

Univerza v Mariboru je pristopila k prenovi svojega informacijskega sistema (IS), ki nudi informacijsko podporo pri izvajanju pedagoškega, raziskovalnega in upravno-administrativnega procesa [1]. V okviru prenove, ki je potekala v letih 1999–2002, so:

- prenesli obstoječe aplikacije informacijskega sistema na novo tehnologijo,
- razvili informacijsko podporo znanstveno-raziskovalnemu procesu, elektronskemu poslovanju s partnerji v slovenskem prostoru in mednarodnemu sodelovanju,
- izvedli izobraževanje delavcev univerze.

Informacijski sistem je zasnovan modularno, kar pomeni, da so gradniki informacijskih podsistemov ločeni moduli, ki so medsebojno ustrezno povezani in uporabljajo skupno bazo podatkov, v kateri so odpravljena vsa podvajanja podatkov. Na ta način smo dosegli odprtost in enostavno prilagodljivost informacijskega sistema.

Logično lahko arhitekturo IS predstavimo z informacijskimi podsistemi. IS tvori pet podsistemov, ki jih prikazuje slika 5.

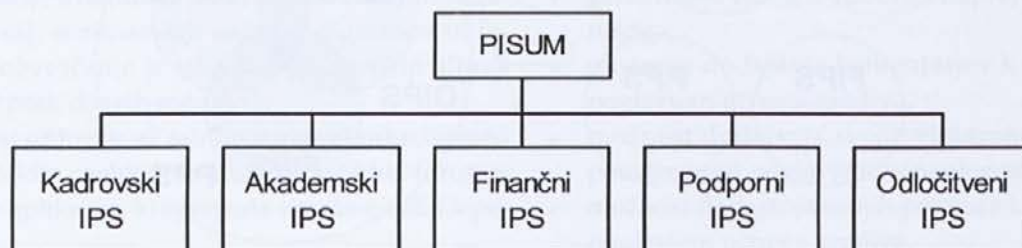
Kadrovski informacijski podsistem (KIPS) vodi evidenco o zaposlenih na Univerzi, ki jo sestavljajo osnovni podatki o zaposlenih, njihovih funkcijah na univerzi in na fakultetah, njihovih koeficientih za plačo, habilitacijah, nagradah in delovnih razmerjih. Podatke iz tega informacijskega podsistema uporablja portal za določanje avtorizacije dostopa do podatkov in storitev IS. Kadrovske podatke uporabljajo vsi informacijski podsistemi (IPS).

Akademski informacijski podsistem (AIPS) vodi evidenco o študentih in izvajanju študijskega procesa. Sestavljajo ga vpisna evidenca, izpitna evidenca, predmetniki, najava študijskega procesa, evidenca pedagoških obveznosti, evidenca o postopkih, ki se izvedejo ob diplomah, magisterijih in doktorskih nalogah, izdelava različnih statistik, študentska anketa, evidenca izrednega študija in skupne funkcije.

Finančni informacijski podsistem (FIPS) sestavljajo glavna knjiga, saldakonti, plače in drugi osebni prejemki, osnovna sredstva, stroškovno računovodstvo, obračun DDV, fakturiranje in finančni tokovi.

Podporni podsistem (PPS) vključuje evidenco vodenja projektov in investicij, administracijo šifrantov, evidenco raziskovalnega dela, sistem za vodenje postopkov in pisarniško poslovanje, zbirko znanja s sistemom samopomoči, sistem elektronskega komuniciranja med študenti in zaposlenimi, podsistem za izobraževanje na daljavo ter portal.

Odločitveni podsistem (OPS) je še v fazi izgradnje in je zasnovan za potrebe hitrih vpogledov v agregirane podatke za potrebe različnih analiz, ki so pomembne pri odločitvah vodstva fakultet in univerze. Za ta namen se na Univerzi v Mariboru vzpostavlja podatkovno skladišče, ki črpa podatke iz skupne podatkovne baze IS.



Slika 5: Informacijski podsistemi

Posamezni IPS niso ločene enote. Med njimi zasledimo preseke – tako v podatkovnem smislu, kakor funkcijskem smislu (slika 6).

Ločimo naslednje uporabnike IS:

- študenti,
- profesorji in asistenti,
- delavci v administraciji,
- predstojniki inštitutov, kateder ali laboratorijev članic UM,
- vodstveni delavci,
- zunanji uporabniki IS.
- delavci, ki skrbijo za sistem interaktivne pomoči IS,
- administratorji IS,
- vzdrževalci IS,
- obiskovalci, ki pridejo do podatkov iz IS prek svetovnega spleta.

Vsak uporabnik ima določeno vlogo v IS, ki mu omogoča dostop do podatkov in funkcij IS.

Pedagoški proces, ki na univerzi vključuje največje število subjektov (študenti in pedagoški kader), uporablja funkcije IS prek portala. Študentom je tako omogočeno, da opravijo formalnosti in najdejo potrebne informacije na enem mestu. Opravijo lahko elektronski vpis, izvedejo elektronsko prijavo (ali odjavo) izpita, preverijo svoje znanje, posredujejo svoje seminarske naloge profesorju po elektronski poti, elektronsko plačujejo svoje finančne obveznosti, zbirajo informacije o študijskem procesu (urniki, izpitni roki, govorilne ure, lokacije izvajanja študijskega procesa ...), predelujejo digitalno učno gradivo, ocenjujejo pedagoško delo z izpolnitvijo elektronskih anket in vprašalnikov ali si izpisujejo potrdila z elektronskim posredovanjem izdajateljem.

Pedagoškemu kadru je na voljo enostavno spletno orodje za:

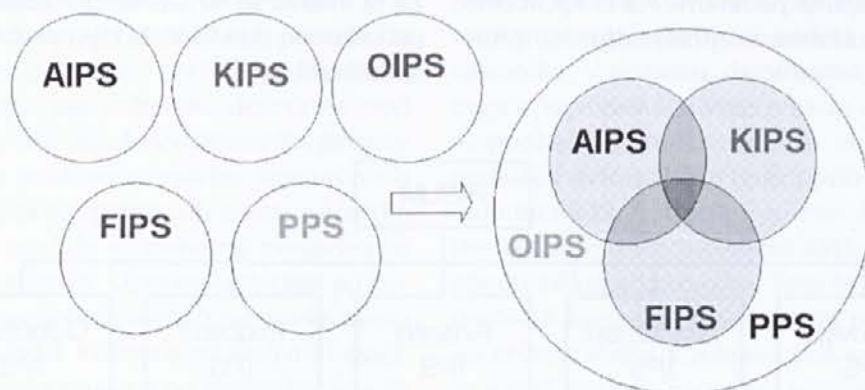
- pregled informacij o stroških na stroškovnem mestu,
- izvedbo izpita – pregled prijav, vnos ocen ...,
- izdelavo opisov predmeta, študijskih obveznosti, povezav na elektronsko učno gradivo

Funkcije IS, ki zahtevajo veliko posegov s tipkovnico, so za administrativne službe (študijski referati, kadrovske in računovodske službe) dostopne prek programske opreme, izdelane v tehnologiji uporabnik-strežnik.

4 Izobraževanje na daljavo

Univerza v Mariboru razvija lastno programsko okolje za podporo e-izobraževanju. Lasten razvoj je potreben zaradi možnosti integracije v obstoječi informacijski sistem univerze, zaradi uporabe slovenskega jezika in večjezičnosti ter zaradi možnosti razvoja lastnih modulov in servisov za e-izobraževanje. Programsko okolje je primerljivo z modernimi komercialnimi produkti (WebCT, IBM Learning Space, Blackboard, Microsoft Class Server).

Učna gradiva v portalu so predstavljena v obliki učnih enot (learning units), iz katerih je mogoče tvoriti poljubna hierarhično ali kako drugače organizirana učna gradiva (npr. v obliki grafa), ki so lahko vezana na skupino študentov ali pa so lahko prilagojena predhodnemu znanju posameznega študenta. Iz učnih enot je možno tvoriti poljubne učne vsebine, jih med seboj navzkrižno povezovati in določati zaporedje, v katerem jih mora študent predelati. Okolje tudi služi za sistem komuniciranja med zaposlenimi in študenti



Slika 6: Preseki med IPS

Univerze. Okolje vsebuje elektronsko knjižnico učnih materialov, slovar izrazov, podporni sistem za učitelja, podporni sistem za administriranje portala in podporni sistem za študenta.

V letu 2005 bomo okolje nadgradili z možnostjo uvoza in izvoza podatkov v format SCORM, tako da bo možen enostaven prenos učnih vsebin v druga učna okolja.

Programsko okolje daje študentu:

- informacije o njegovem urniku predavanj in vaj ter pedagoškemu osebju,
- informacije o predmetih, ki jih poslušata ter pripadajočih izpitnih rokov,
- elektronsko prijavo/odjavo na izpite in pregled ocen izpitov,
- elektronski vpis,
- elektronsko izbiranje predmetov,
- servisiranje zahtev po potrdilu o izpiti in frekvencajskem potrdilu,
- dostop do obrazcev fakultet, ki so vezani na študente,
- dostop do elektronskih študijskih gradiv za predmete, ki jih poslušata,
- možnost dodajanja komentarjev, zaznamkov in vprašanj profesorju ali asistentom k posameznim poglavjem učnega gradiva,
- možnost dodajanja svojih elektronskih gradiv k posameznim poglavjem učnega gradiva,
- možnost predlaganja novih pojmov za slovar strokovnih izrazov za učno gradivo,
- možnost dodajanja svojih povezav k posameznim poglavjem učnega gradiva,
- možnost samopreverjanja usvojenega znanja,
- možnost elektronskega oddajanja vaj, seminarskih nalog, programov,
- informacije o ocenah vaj, seminarskih nalog, kolokvijev, pisnih izpitov (obveščanje je možno prek spleta, WAP aplikacije, govorne aplikacije ali na zahtevo študenta prek SMS sporočila ali na podlagi pošiljanja SMS sporočila študentu),
- informacije o datumu, uri in prostoru ustnih zagovorov vaj, seminarskih nalog, programov in izpitov (obveščanje je tako kot pri prejšnji alineji možno prek drugih medijev),
- možnost sinhrono ali asinhrono komunikacije med pedagoškim osebjem in študenti v obliki foruma, spletne aplikacije, ki sprejema vprašanja ali klepetalnice,
- možnost izpolnjevanja različnih anket in vprašalnikov o kakovosti pedagoškega procesa,

- možnost iskanja določene informacije v slovarju ali v vsebini (razen za kviz),
- dostop do podatkov oglasne deske:
 - pregledovanje obvestil o odpadlih predavanjih in vajah,
 - napoved vabljenih predavanj, dogodkov, zagovorov diplom, magisterijev itd.,
 - zapisniki sestankov organov fakultet, univerze in drugih izobraževalnih organizacij ...

Programsko okolje ponuja pedagoškemu osebju:

- primerna sistematično strukturirana navodila za pripravo spletnih učnih gradiv in uporabe tehnologij izobraževanja na daljavo,
- informacije o urniku predavanj in vaj,
- dostop do obrazcev fakultet, ki so vezani na delavce fakultet,
- informacije o študentih, ki jim predavajo ali vodijo vaje,
- možnost dopolnjevanja osnovnih podatkov o predmetih in vajah, ki jih predavajo,
- objavlanje elektronskih študijskih gradiv za predmete ali vaje pri predmetih, ki jih predavajo ali vodijo vaje (gradiva so lahko kateregakoli večpredstavnega tipa),
- možnost uporabe vsebine poglavij učnih gradiv, vsebin vaj, domačih nalog, seminarskih nalog iz prejšnje generacije ali iz druge skupine v isti generaciji ali celo iz drugega predmeta, ki ga predava profesor,
- možnost dodajanja literature k posameznim poglavjem in k celotni vsebini predmeta,
- možnost pregledovanja slovarjev izrazov,
- možnost tvorjenja slovarja izrazov ter povezovanja izrazov v slovarju,
- možnost razvrščanja študentov v skupine,
- definiranje vsebine vaj, seminarskih nalog, programov ali domačih nalog (vsebine so lahko vezane na točno določenega študenta, skupino študentov ali so enake za vse študente),
- generiranje nalog iz nabora vnaprej pripravljenih nalog,
- možnost dodajanja komentarjev k posameznim poglavjem učnega gradiva,
- možnost dodajanja svojih elektronskih gradiv k posameznim poglavjem učnega gradiva,
- možnost dodajanja svojih povezav k posameznim poglavjem učnega gradiva,
- možnost dodajanja svojih povezav k celotni vsebini predmeta,

- možnost definiranja elektronskih kvizov za študente,
- pregledovanje in ocenjevanje elektronsko oddanih vaj, seminarских nalog, programov prek spletne aplikacije,
- možnost definiranja pravil ocenjevanja vaj, domačih nalog, skupne ocene ...,
- določanje rokov oddaje posameznih vaj, zagovorov vaj, seminarских nalog, programov, zagovorov ustnih izpitov,
- vstavljanje informacije o ocenah vaj, seminarских nalog, kolokvijev, pisnih izpitov,
- vstavljanje informacije o datumu, uri in prostoru ustnih zagovorov vaj, seminarских nalog, programov in izpitov,
- možnost sinhrono ali asinhrono komunikacije med študenti v obliki foruma, spletne aplikacije, ki sprejema vprašanja ali klepetalnice,
- možnost urejanja pogosto zastavljenih vprašanj iz celotnega nabora vprašanj študentov,
- nalaganje informacije o študentih ali skupini študentov na dlančnik,
- vpisovanje zabeležk o študentu prek dlančnika ali prek spletne aplikacije,
- pošiljanje obvestil posameznim študentom, skupinam študentom ali vsem študentom, ki poslušajo predmet po elektronski pošti ali s pošiljanjem SMS,
- možnost vodenja evidence o prisotnosti na vajah in predavanjih z uporabo spletne aplikacije ali dlančnika,
- ocenjevanje vaj in ocenjevanje študentov na predavanjih z dlančnikom,
- vodenje zabeležk o študentih prek dlančnika ali spletne aplikacije,
- možnost dodajanja nalog z rešitvami,
- možnost spremljanja statistike za posameznega študenta, skupine študentov ali vseh študentov (prisotnosti na vajah in predavanjih, število oddanih vaj, ocene kolokvijev, vaj, seminarских nalog, domačih nalog, napredovanje študentov pri usvajanju učne snovi, ocene pri samopreverjanju)
- pregledovanje prijav študentov v sistem.

Programsko okolje omogoča administratorju:

- dodeljevanje uporabniških imen,
- generiranje novih predmetov,
- določanje parametrov predmeta (način oddaje dokumentov, ocenjevanje ...),
- povezovanje podatkov z novim predmetom (predmet, pedagoški delavci, študenti),

- dodeljevanje dostopov študentov in pedagoškemu osebju do posameznih funkcij programskega okolja,
- pregledovanje sistemskih sporočil,
- pregledovanje prijav študentov v sistem.

Portal omogoča tvorjenje svoje strukture portala za vsak predmet za vsako generacijo študentov oziroma za vsako skupino študentov znotraj generacije posebej. Beležiti moramo namreč zgodovino, saj drugače ne vemo, če je neki študent pred več leti opravil vaje ali predavanja.

5 Sklep

V članku je predstavljen tehnični in vsebinski vidik izgradnje portala Univerze v Mariboru. Portal je integriran z informacijskim sistemom Univerze v Mariboru in programskim okoljem za podporo e-izobraževanju. Bralcu članka se najverjetneje poraja vprašanje zakaj smo se odločili za lasten razvoj portala, če pa obstaja že množica komercialno dostopnih produktov, s katerimi bi lahko dosegli enake učinke. Razlog je predvsem v lažji in boljši integraciji portala z drugimi podsistemi informacijskega sistema Univerze v Mariboru. Zaradi omejenih finančnih sredstev smo v izhodišču projekt prenove informacijskega sistema Univerze v Mariboru zasnovali tako, da so ga v največji meri izvajali strokovnjaki z univerze. Če bi hoteli doseči podobne funkcionalnosti s komercialno dosegljivimi produkti, bi to zahtevalo večja finančna sredstva, potrebna dodatna znanja sodelavcev na projektu, rezultat pa bi bil najverjetneje dosežen zunaj časovnih rokov, ki smo si jih zastavili na Univerzi v Mariboru. Portal je možno enostavno nadgraditi z novimi storitvami, ki bodo omogočale izmenjavo podatkov z drugimi informacijskimi sistemi (npr. zapise o študentih in zaposlenih, študijska gradiva), kar bo omogočilo lažjo izmenjavo študentov in profesorjev z drugimi univerzami in izvedbo kreditnega študija znotraj univerze in med univerzami. Portal in programsko okolje za podporo e-izobraževanju gradimo tako, da ga bo z manjšimi modifikacijami možno implementirati tudi v drugih (predvsem večjih) izobraževalnih institucijah.

Pri uvajanju portala in ostalih informacijskih podsistemov ter okolja za podporo izobraževanju na daljavo smo naleteli na precej organizacijskih problemov. Eden izmed najbolj perečih problemov je bila vzpostavitev ažurne kadrovske evidence. Brez ažurne kadrovske evidence namreč nismo mogli vzpostaviti enotnega avtentikacijskega sistema za zaposlene. Precejšen problem je

bila tudi delna računalniška nepismenost zaposlenih na univerzi. Nekateri zaposleni še namreč niso znali v zadostni meri uporabljati programskega paketa MS Office, elektronske pošte in spletnega brskalnika. Slabo računalniško pismenost odpravljamo z izvajanjem internih izobraževalnih tečajev. Prav tako izvajamo tudi tečaje na področju uvajanja e-izobraževanja (priprava študijskih gradiv, uporaba okolja za podporo e-izobraževanju). Najhitreje so novosti in možnosti sprejeli izredni in redni študenti, saj jim sedanji sistem nudi dostop do vsebin in storitev informacijskega sistema univerze prek interneta, mobilnih telefonov in dlančnikov, kar jim prihrani čas in denar (npr. čas in stroški prevoza), ki bi ga potrošili, če bi želeli te storitve opraviti na fakulteti.

Zahvala

Projekt je bil realiziran s sredstvi, ki so jih prispevali na Univerzi v Mariboru, Ministrstvu za šolstvo znanost in šport in Ministrstvu za informacijsko družbo. Vsem se iskreno zahvaljujemo za finančno in organizacijsko podporo.

Milan Ojsteršek je izredni profesor na Fakulteti za elektrotehniko, računalništvo in informatiko v Mariboru in vodja Centra za heterogeno procesiranje in računalniškega centra na tej fakulteti. Skupaj s sodelavci se ukvarja z razvojem porazdeljenih in spletnih aplikacij. Namen njegovega dela in dela Centra za heterogeno procesiranje je prenos najnovejših znanstvenih dosežkov in tehnologij v podjetja in na študente računalništva in informatike Fakultete za elektrotehniko, računalništvo in informatiko v Mariboru.

Dejan Dinevski je docent in vodja Oddelka za informacijski in tehnološki razvoj na Univerzi v Mariboru. Zadolžen je za univerzitetne informacijske sisteme, prvenstveno pa za uvajanje informacijske podpore izobraževalnemu procesu na univerzi.

Mag. Tomaž Klojčnik je predstojnik Računalniškega centra Univerze v Mariboru (RCUM). Ukvarja se predvsem s koordinacijo različnih projektov posodobitve informacijske in komunikacijske infrastrukture Univerze v Mariboru.

Viri

- [1] KLOJČNIK, Tomaž, OJSTERŠEK, Milan, MASIĆ, Štefan, HORVAT, Branko, SCHAFFRANIETZ, Ervin, BEDEKOVIČ, Violeta: Projekt PISUM : [razvojni projekt Prenove informacijskega sistema Univerze v Mariboru], Maribor: Univerza, Računalniški center, 1999–2001.
- [2] HORVAT, Branko. Spletno rudarjenje in personalizacija : magistrsko delo. Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor: 2003.
- [3] GORENJAK, Borut. Univerzalni model za gradnjo uporabniških vmesnikov in integracijo spletnih storitev : magistrsko delo. Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, 2004
- [4] BREZOVNIK, Janez. *Izdelava sistema za upravljanje z digitalnimi identitetami : diplomsko delo univerzitetnega študijskega programa*, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, 2004.
- [5] YU, Philip S.: Data Mining and Personalization Technologies, 6th International Conference on Database Systems for Advanced Applications (DASFAA '99), Hsinchu, Tajvan, 1999, str. 6–14.
- [6] UIML.org. Home of the User Interface Markup Language. Dostopno na naslovu: <http://www.uiml.org> . Obiskano: 8. decembra 2004.
- [7] Microsoft. Longhorn Markup Language (code name »XAML«) Overview, april 2003. Dostopno na naslovu: <http://longhorn.msdn.microsoft.com/?//longhorn.msdn.microsoft.com/lh/sdk/core/overviews/about%20xaml.aspx> Obiskano: 8. decembra 2004.
- [8] T. Andrews, F. Curbera, H. Dholakia, Y. Golland, J. Klein, F. Leymann, K. Liu, D. Roller, D. Smith, S. Thatte, I. Trickovic, S. Weerawarana. Business Process Execution Language for Web Services Version 1. 1. november 2003. Dostopno na naslovu: <http://www-106.ibm.com/developerworks/library/ws-bpel/>. Obiskano: 2. julija 2004.

Vpliv spletnih portalov na kakovost visokošolske izobraževalne institucije

Jaka Lindič, Aleš Popovič, Jurij Jaklič, Mojca Indihar Štemberger

Faculty of Economics, University of Ljubljana, Kardeljeva ploščad 17, 1000 Ljubljana, Slovenia

jaka.lindic@ef.uni-lj.si, ales.popovic@ef.uni-lj.si, jurij.jaklic@ef.uni-lj.si, mojca.stemberger@ef.uni-lj.si

Povzetek

Tako kot druge organizacije se morajo tudi izobraževalne institucije prilagajati vse hitreje spreminjajočemu se okolju, ki prinaša vedno nove izzive. Pri uvajanju e-izobraževanja se na fakultetah srečujemo z mnogimi ovirami, ki jih lahko razdelimo v tehnološke in kulturne. Mnoge prednosti e-izobraževanja, kot npr. stroškovna učinkovitost, večja odzivnost na spremembe, ažurne vsebine, fleksibilnost pri dostopu in prinašanje vrednosti za uporabnika ne temeljijo zgolj na uporabi napredne tehnologije. Ne moremo pričakovati, da je tehnologija sama po sebi dovolj za spremembe v načinu dela ljudi. Tehnološke rešitve v obliki portalov, ki omogočajo integracijo razpršenih virov, so znane že nekaj let. V zvezi s temi rešitvami postaja področje njihove učinkovitosti in uporabnosti vse aktualnejše, saj integracija več ne zadostuje. Uporabniki želijo do informacij prihajati na različne načine – personalizirano.

Članek obravnava področje uporabe interneta kot medija za doseganje enega izmed osnovnih ciljev izobraževalnih institucij, to je izboljšanje kakovosti. Prikazali bomo, kako visokošolsko izobraževalno okolje vpliva na doseganje tega cilja. Model smo praktično preizkusili na primeru Ekonomske fakultete Univerze v Ljubljani. Primer prikazuje tudi, da se spremembe nikoli ne zgodijo »čez noč«.

Abstract

The impact of web portals on quality of institutions of higher education

Institutions of higher education are just like other organizations forced to adapt to the ever faster changing environment that brings many new challenges. There are several obstacles in the way of introducing e-learning in these institutions. We can divide them into technology-based and culturally based. Many benefits of e-learning such as cost-effectiveness, enhanced responsiveness to change, timely content, flexible accessibility, and providing value to the customer are not based only on use of high technology. We cannot expect that the use of advanced technology is enough to change the way we work as a human being. Technologic solutions in form of portals have been known for quite some years. A recurring problem has become efficiency and usefulness of these solutions. Integration of dispersed sources is not sufficient. Individual users should obtain information in a variety of ways - in a personalized way.

The paper will address the topic of using Internet as a medium to achieve one of the primary goals of institutions in higher education; that is quality improvement. We will show the influence of e-learning environment to achieving this goal. The model has also been tested in practice, at the Faculty of Economics at the University of Ljubljana. This case has also proved that cultural changes never take place over night.

1 Izzivi in priložnosti

Izobraževalne institucije se vse bolj srečujejo z zahtevami po povečevanju operativne učinkovitosti in uspešnosti, ki vodi v vidnejše uspehe tako študentov kot tudi pedagogov. Visokošolske izobraževalne institucije se morajo tudi vse hitreje prilagajati spremembam v okolju (večanje števila študentov, večja odgovornost do skupnosti, kulturne spremembe, vseživljenjsko učenje, večja dostopnost tehnologije, novi in močnejši obstoječi konkurenti). Pomemben je tudi vpliv heterogenosti študentov, njihove informacijsko-komunikacijske opremljenosti, etnološkega ozadja, znanja in sloga učenja. Z združevanjem Evrope postajata ideja o popolni mobilnosti profesorjev in študentov ter konkurenca na področju izobraževanja močna in pomembna faktorja v deželah, za katere to v

preteklosti ni bilo značilno. V kolikor želijo visokošolske institucije ostati konkurenčne morajo nenehno skrbeti za uvajanje in uporabo novih, inovativnih metod učenja in temeljito spreminjati načine izobraževanja.

Prisiljene so v uporabo znanja in prenos izkušenj iz poslovnega sveta. Vse bolj se zavedajo, da niso pomembne zgolj in samo osrednje funkcije, temveč tudi podporne. Novi pristopi k učenju predstavljajo izzive in priložnosti tako za profesorje kot tudi za strokovno oz. administrativno osebje. E-izobraževanje dandanes običajno povezujemo z uporabi zahtevne tehnologije (širokopasovnih povezav, zmogljive strojne in programske opreme). Eden izmed večjih problemov je

pomanjkanje virov, povezanih z informacijsko tehnologijo, tako tehničnega osebja kot tudi tehnologije.

V drugem poglavju obravnavamo posebnost izobraževalnega okolja v luči e-izobraževanja. V tretjem poglavju predstavljamo model vpliva spletnih portalov na kakovost visokošolske izobraževalne ustanove. Sledi poglavje s predstavitvijo študije primera institucije, kjer je tovrstna rešitev uspešno vpeljana. V sklepu povzemamo naše izkušnje in poudarimo ključne dejavnike za uspešen razvoj, uvajanje in uporabo rešitve.

2 Visokošolsko izobraževalno okolje in e-izobraževanje

E-izobraževanje je nadgradnja učenja in izobraževanja. Pogosto ga razumemo in opredelimo v smislu materialov, ki se prenašajo prek elektronskih medijev vključno z internetom, intraneti, ektraneti, satelitskimi širokopasovnimi povezavami, avdio/video trakovi, interaktivno televizijo in CD-ROM-i [2]. E-izobraževanje lahko razumemo tudi kot uporabo interneta in z njim povezanih tehnologij za prenos materialov, podporo in izboljšavo poučevanja, učenja in ocenjevanja. Pri uvajanju e-izobraževanja v visokošolske ustanove se srečujemo s številnimi ovirami, ki so tehnološkega in kulturnega značaja. Tehnološki ponudniki se usmerjajo predvsem na tehnološki vidik e-izobraževanja. Ponujajo visokotehnološke rešitve, ki omogočajo interaktivne izobraževalne materiale, običajno v večpredstavnostni obliki in v obliki predavanj, ki se prenašajo preko videokonferenčnih povezav. Ključna razlika med tradicionalnim učilniškim izobraževanjem in e-izobraževanjem je v mediju preko katerega izobraževanje poteka [3]. V tradicionalnem okolju ima pedagog popoln nadzor nad učnim okoljem. Pri e-izobraževanju pa je s pedagoškega vidika značilna predvsem zmanjšana možnost prilagajanja.

Med tehnološke ovire štejemo predvsem dostop do omrežnih povezav. V zadnjem obdobju je izpostavljen predvsem vidik dostopnosti širokopasovnih povezav. Kljub temu, da se dostopnost širokopasovnih povezav v večini držav povečuje, ta še vedno ni dovolj visoka, da bi lahko e-izobraževanje temeljili na njej. V ZDA še vedno več kot 67 % uporabnikov uporablja povezave s hitrostjo do 56 Kbps [9]. V Sloveniji pa je delež uporabnikov širokopasovnih povezav 21% [8]. Druge tehnološke ovire so povezane s potrebami po specializirani strojni in programski opremi. Za izdelavo večpredstavnostnih učnih materialov namreč tipično potrebujemo posebno programsko opre-

mo. Poleg tega je treba upoštevati tudi potrebe po posebnem znanju in izkušnjah. Pedagog mora imeti namreč posebna znanja, kako pripraviti večpredstavnostna gradiva. E-izobraževanje je namreč bistveno več, kot enostaven prenos knjig in drugih tiskanih gradiv v elektronsko obliko. Kljub temu, da na prvi pogled izgleda, da je ključni problem tehnologija, so po našem mnenju to v prvi vrsti ljudje.

Tradicionalni način, na katerega smo pridobivali znanje in formalno izobrazbo, se zelo dolgo ni spremenil. Neracionalno je pričakovati, da je za spremembo načina razmišljanja in delovanja dovolj zgolj uporaba napredne tehnologije. Kulturne spremembe se nikoli ne zgodijo čez noč.

Vseeno pa e-izobraževanje prinaša številne koristi vključno s stroškovno učinkovitostjo, večjo odzivnostjo na spremembe, konsistentnostjo odziva na spremembe, ažurnimi vsebinami, fleksibilnostjo pri dostopu in dodajanjem vrednosti za uporabnika [7]. Kot lahko vidimo ključne koristi niso povezane z uporabo napredne tehnologije. Glavna prednost e-izobraževanja je podpora učinkoviti komunikaciji med študenti, profesorji in strokovnimi službami. To pa vodi v učinkovitejše in uspešnejše poučevanje, učenje in ocenjevanje.

Rešitev, ki jo predlagamo, je vpeljava spletnih portalov. Poslovna sfera se že leta srečuje s preobiljem informacij. S tehnološkim napredkom pa se to preobilje še stopnjuje. Rešitev, ki je znana že vrsto let, so spletni portali. Prepogosto so ti razumljeni predvsem v smislu tehnologije, ki pa predstavlja zgolj en vidik. Veliko pomembnejša je vsebina samega portala in procesi, ki so povezani z njim. S tehnologijo je po nekaterih ocenah povezanega približno tretjina dela za postavitev dobrega portala [5]. Kar portale razlikuje od statičnih spletnih strani je njihova značilnost, da vključujejo podatke iz različnih virov, v različnih oblikah in jih predstavljajo na poenoten konsistenten način, prek enotne točke dostopa. Z vidika uporabnika to pomeni, da predstavlja portal enotno izhodiščno točko za pokrivanje njegovih informacijskih potreb na nekem delovnem področju. V primeru izobraževalne institucije to npr. pomeni, da študentski portal podpira vse aktivnosti študenta v okviru izobraževalnega procesa in ne uporablja ločenih sistemov za različna opravila (dostop do gradiv, prijavljanja na izpite in drugo). Med tipične funkcionalnosti, ki jih mora vsebovati portal, štejemo varnost, omrežje, administrativna orodja, iskalna orodja, orodja za upravljanje z vse-

binami, posebitve, razširljivost in enostavnost uporabe [6].

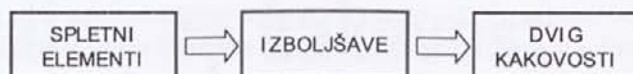
Najpomembnejši namen izgradnje spletnih portalov je izboljšanje učinkovitosti in uspešnosti zaposlenih. Učinkovitost izboljšujemo z večjo uporabnostjo (hitrejše in natančnejše pridobivanje informacij, hitrejše in pravilnejše opravljanje nalog) in zmanjšanjem odvečnih in ponavljajočih se postopkov (podatki naj se v sistem vnašajo samo enkrat, kjer nastanejo in ko nastanejo). Boljša učinkovitost zaposlenih ima številne pozitivne učinke. Namesto, da se zaposleni ukvarjajo z rutinskimi opravili, lahko koristneje uporabijo svoj čas (za pridobivanje novih znanj, optimizacijo drugih procesov itd.). Kot ugotavljajo raziskave na področju organizacijske teorije, zmanjšanje rutinskih opravil vodi v boljše zadovoljstvo pri delu. Vse to pa lahko zopet vodi v večjo uspešnost posameznikov in institucije v celoti. Pri tem je treba poudariti, da portali vse to zgolj omogočajo. Da bi portali dejansko vodili k dvigu kakovosti, je treba poskrbeti tudi za druge vidike, predvsem vsebine pa tudi uporabnost.

3 Model vpliva spletne rešitve na kakovost visokošolske izobraževalne ustanove

Pri uvajanju različnih informacijskih rešitev je pogosto težko pokazati neposredno povezanost med vlaganji oz. njihovimi rezultati (rešitvami) ter uspešnostjo (merljivimi rezultati). V nadaljevanju predstavljamo model povezanosti elementov portala in kakovostjo visokošolske izobraževalne institucije. Samo kakovost institucije lahko opredelimo na različne načine. Sestavljena je iz mozaika različnih področij:

- kakovosti študija in pridobljenega znanja (zaposljivost študentov);
- kakovosti pedagoškega procesa (prehodnost študentov med letniki, zadovoljstvo študentov);
- kakovosti raziskovalnega dela pedagogov in študentov (npr. število objavljenih člankov v znanstvenoraziskovalnih revijah);
- vpetosti institucije v poslovno in širše družbeno okolje (sodelovanje gostov iz prakse, prenos znanja oz. raziskovalnih izsledkov v prakso, povečanje pretoka znanja in idej);
- vpetosti v mednarodno okolje (mednarodna izmenjava študentov in profesorjev);
- poslovne uspešnosti (raziskave za trg, svetovalna in korporativna izobraževalna dejavnost).

Spletne rešitve s posameznimi elementi, ki jo sestavljajo, omogočajo izboljšave na različnih področjih.



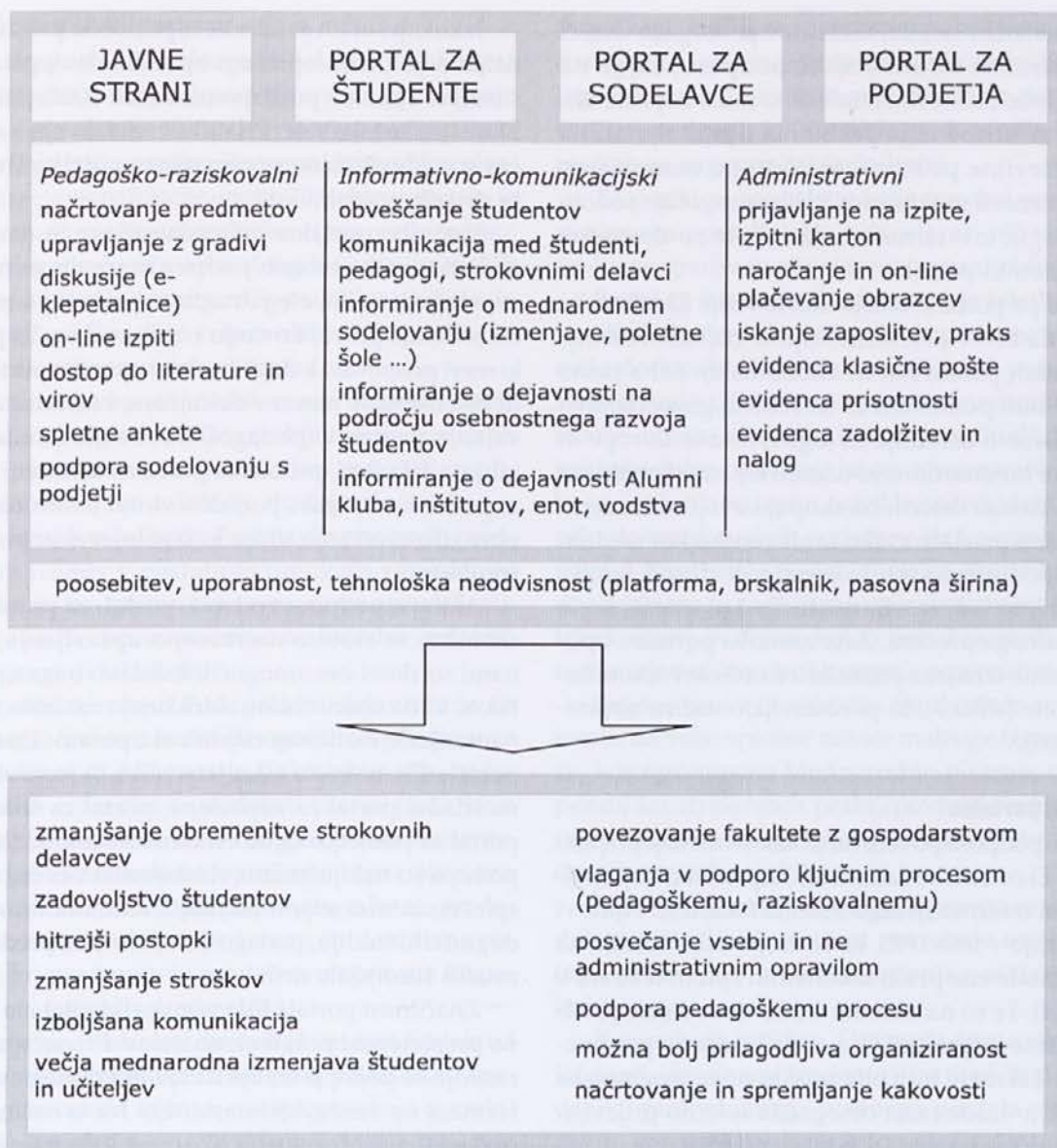
Slika 1: Povezanost med spletnimi elementi in dvigom kakovosti institucije

Te izboljšave pa posredno ali neposredno omogočajo dvig kakovosti institucije (slika 1). Portali lahko vsebujejo različne vsebinske elemente, ki jih lahko razdelimo v tri skupine: pedagoško-raziskovalni, informacijsko-komunikacijski in administrativni (slika 2). V prvi skupini so elementi, ki so povezani z načrtovanjem predmeta, njegovim upravljanjem in sodelovanjem pri predmetu. Pedagogi lahko objavljajo obvestila, pripravijo terminski načrt in uporabljajo dokumentni sistem za objavljanje gradiv. Pedagogi, študenti in zunanji sodelavci (npr. gostje iz prakse) lahko skupaj sodelujejo v on-line diskusijah. Študenti lahko dostopajo do on-line študijske literature in virov (on-line baze člankov) in prek spleta opravljajo izpite. S pomočjo spletnih anket je možno pridobiti ocene študentov in posledično izboljšati pedagoški proces.

Med informacijsko-komunikacijske elemente lahko štejemo različne informacije, povezane s študijem (npr. obvestila predmetov, terminski načrt, termin ogledov kolokvijev, aktualne spremembe govornih ur, ogled izpitov ipd.) in informacije na ravni posameznih enot (npr. enotami za mednarodno sodelovanje, poslovno sodelovanje, razvoj študentov ipd.) ali ravni fakultete (npr. informacije o dejavnosti alumni kluba, dejavnostih vodstva ipd.).

V zadnji, a nič manj pomemben sklop elementov, lahko vključimo administrativne postopke. Študenti se lahko prijavijo na izpite, si ogledajo izpitni karton, izpolnjujejo in naročajo različne obrazce, poiščejo lahko zaposlitev ali študijsko prakso. Zaposlenih lahko npr. upravljajo s prejeto in poslano pošto, evidentirajo prisotnost ter spremljajo zadolžitve in naloge.

Posamezni elementi na različne načine in v različnem obsegu vplivajo na kakovost institucije. Tako administrativni elementi vplivajo na hitrejše in učinkovitejše izvajanje administrativnih postopkov. Na ta način so zaposleni manj obremenjeni, poveča se zadovoljstvo študentov, stroški pa se znižajo. Tudi nekateri informativni elementi vplivajo na boljšo obveščeno in s tem pozitivno vplivajo na izvajanje procesov. S pomočjo spletnih anket lahko institucija učinkoviteje spremlja kakovost dela profesorjev in strokovnih



Slika 2: Model vpliva spletnih portalov na kakovost visokošolske izobraževalne institucije

delavcev. S tako pridobljenimi informacijami lahko nenehno izboljšuje svoje poslovanje. Z odpravo oziroma poenostavitvijo rutinskih opravil se lahko pedagogi namesto administrativnim opravilom bolj posvetijo vsebini, strokovni delavci pa podpora pedagoškega procesa. S posebnim portalom za podjetja lahko partnerji institucije dostopajo do življenjepisov študentov, izbirajo ustrezne študente in stopijo v kontakt z njimi, ponudijo strokovne prakse ipd.

4 Portali Ekonomske fakultete – spletna triada

Ekonomska fakulteta v Ljubljani je med največjimi fakultetami v Sloveniji, saj v izobraževalnem procesu sodeluje več 10.000 dodiplomskih in skoraj 2000 podiplomskih študentov, zaposlenih pa je le nekaj več kot 220 delavcev, od tega 150 pedagoških, kar pomeni skoraj 80 študentov na pedagoškega delavca. V takšnih pogojih je zelo težko imeti s študenti osebni odnos in zagotoviti ustrezno kakovost študija. Povečanje kako-

vosti pri takšnem številu študentov je mogoče doseči samo z ustrežno informacijsko podporo, saj je na fakulteti glede na število študentov malo zaposlenih.

Za premostitev tega problema uporabljamo na fakulteti številne pristope, še zlasti pa namenjamo veliko pozornosti možnostim, ki jih omogočajo sodobne tehnologije z internetom na čelu in so dostopne prek fakultetnih portalov.

Potreba po portalih izhaja tudi iz vizije Ekonomske fakultete, da bo »viden član skupine najkakovostnejših evropskih poslovnih in ekonomskih šol z jasno prepoznavnim pomenom za slovensko gospodarstvo in gospodarstvo osrednje in jugovzhodne Evrope in z najvišjimi mednarodnimi učnimi ter raziskovalnimi standardi, zaradi katerih bo skupaj s svojimi študenti in diplomanti privlačna tako za slovensko kot mednarodno gospodarstvo« [1]. To je mogoče doseči samo s povečevanjem kakovosti institucije kot celote, ne le izobraževalnega procesa. Zato zasnova portalov upošteva oziroma izhaja iz strateških usmeritev Ekonomske fakultete (slika 3), ki predstavljajo načine uresničevanja vizije.

4.1 Razvoj portalov

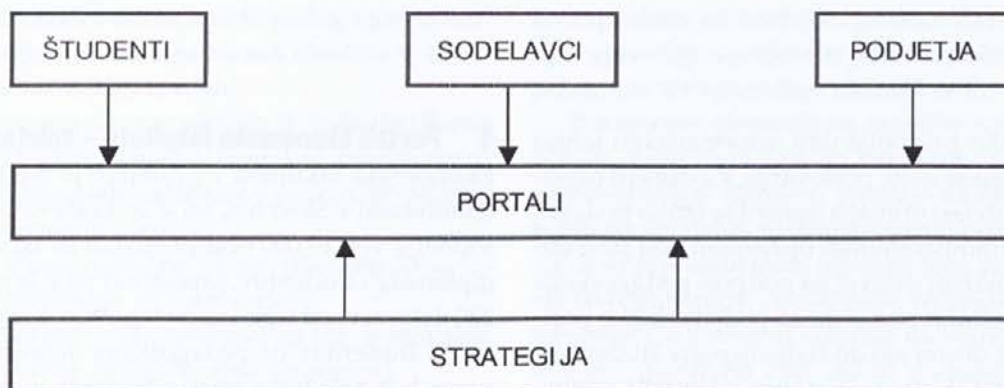
Učinkovito podporo povečevanja kakovosti dajejo spletni portali Ekonomske fakultete, ki predstavljajo najvidnejši del informacijskega sistema fakultete. Njihovi začetki segajo v leto 1995, ko so za potrebe študija na daljavo nastale ene prvih fakultetnih spletnih strani v Sloveniji [4]. Te so na začetku študentom dajale predvsem koristne informacije v zvezi s študijem, predmeti in izpiti. Takrat je tudi bila razvita prva programska oprema, ki je študentom omogočila spletno prijavljanje na izpite ter vpogled v izpitne rezultate.

Izkušnje, ki jih je fakulteta pridobila pri študiju na daljavo, je kmalu s pridom uporabila tudi pri rednem študiju. Tako so postopoma začeli nastajati portali Ekonomske fakultete. Pri njihovem razvoju so bila ves čas kot izhodišča na prvem mestu potrebe študentov in drugih uporabnikov.

Uporabo portalov pri pedagoškem in drugih administrativnih procesih podpira in spodbuja tudi vodstvo fakultete, ki poleg drugih uporabnikov tudi tvorno sodeluje pri oblikovanju novih vsebin. To je v veliki meri prispevalo k dejstvu, da je uporaba portalov pridobila ustrezno mesto v običajnem, vsakodnevnem izvajanju procesov (pedagoških, raziskovalnih in drugih) na fakulteti, pri čemer vodstvo fakultete in njeni zaposleni v uporabi portalov vidijo tehnološko podporo uresničevanju vizije in izpolnjevanju poslanstva fakultete.

Aktivna podpora vodstva, posluš za potrebe uporabnikov in skrbno načrtovano upravljanje z vsebinami so skozi čas omogočili izdelavo bogatejših portalov, ki na enem mestu združujejo različne vsebine, namenjene različnim ciljnim skupinam. Danes tako portale Ekonomske fakultete (slika 3) sestavlja spletna triada: portal za zaposlene, portal za študente in portal za podjetja. Kljub vsebinsko zaokroženim sklopom pa so naključnemu obiskovalcu ves čas na voljo splošne informacije o fakulteti, aktualnih novicah in dogodkih, študiju, pedagoških delavcih, predmetih in ostalih študijskih virih.

Značilnost portala Ekonomske fakultete je tudi tesna prepletenost posameznih delov. Pri načrtovanju in razvoju se pojavlja težnja k cilju enkratnega vnosa informacij na mestu, kjer nastanejo. Na ta način je zagotovljena pravočasnost in ažurnost informacij.



Slika 3: Portali za različne javnosti podpirajo strategijo fakultete

4.2 Funkcionalnosti portalov

Portal za zaposlene in druge sodelavce je fakulteta uvedla leta 2001. Zaposlenim omogoča opravljanje različnih opravil, tako vsakodnevnih kot tudi tistih, ki jih opravljajo bolj poredko. Za portal je značilna visoka stopnja personalizacije vsebin. Vsebina portala je prilagojena skupinam uporabnikov (npr. pedagogom, zaposlenim v referatu ali v drugih organizacijskih enotah ipd.), vsak posameznik pa lahko vsebino še nadalje prilagodi svojim željam ali potrebam, ne glede na to, ali so te trenutne ali trajne.

Upravljanje z obvestili in dokumenti. Zaposlenim je omogočeno pregledovanje in objavlanje različnih obvestil in dokumentov (pravilnikov, navodil, zapisnikov, poročil ...) ter določanje dostopa do le-teh.

Medsebojna komunikacija. Predstavlja eno prvih predstavljenih rešitev v portalu in omogoča zaposlenim komunikacijo na medosebni ravni, ravni opredeljenih skupin (npr. vodstvo, posamezne katedre, referati, ipd.) ali na ravni fakultete.

Pedagoška dejavnost. Pedagoškim delavcem je na preprost in učinkovit način omogočeno upravljanje z lastno pedagoško dejavnostjo. Tako lahko npr. pedagoški kader upravlja s predmeti (ureja podatke o predmetih, objavlja obvestila in gradiva za študente, ureja pogosto zastavljena vprašanja, upravlja s terminskim načrtom dela pri predmetu tako za predavanja kot vaje, upravlja s prijavi študentov na obveznosti pri predmetu in moderira razprave v klepetalnicah oz. forumih predmetov).

Upravljanje z vsebinami spletnih strani enot. Sodelavci enot in centrov fakultete lahko prek portala za zaposlene na enostaven in hiter način upravljaajo z vsebinami in osnovno strukturo predstavitvenih strani svojih enot oz. centrov. Tako lahko npr. urejajo obvestila, objavljajo različna gradiva in poročila, sprejemajo prijave na dogodke ipd.

Dostop do raziskovalnih podatkov. Za zaposlene, še posebej pa pedagoge, so na enem mestu zbrani dostopi do različnih raziskovalnih podatkov: baze zaključnih računov slovenskih podjetij, Centralne ekonomske knjižnice, Evropskega dokumentacijskega centra, inštitutov in drugih centrov fakultete za potrebe raziskovalnega dela.

Finančni pregledi in evidence. Vključujejo finančne preglede lastnih raziskovalnih sredstev in drugih projektov, na katerih je sodelavec za tak pregled pooblaščen. Spletne evidence podpirajo zlasti fakultetne administrativne procese: evidence prisotnosti nepe-

dagoškega kadra, evidence prispele, tranzitne in oddane pošte, evidence lastnih in tujih zadolžitvev ter nalog, evidence dejavnosti zaposlenih na domači instituciji in na drugih inštitucijah, evidence članstva v komisijah za zagovor zaključnih del.

Za portal za zaposlene velja posebej poudariti, da je njegova uporaba zelo zaživelja, saj nič več ne uporabljamo gradiv in obveščanja v pisni obliki.

Portal za študente (Študent-net) je fakulteta uvedla leta 2003. Študentom omogoča, da prek enotne vstopne točke dostopajo do vsebin in orodij, ki jih potrebujejo pri svojem študiju. Tako imajo na enem mestu zbrane informacije, novice, obvestila, dokumente, študijska gradiva in klepetalnice, opravljajo lahko administrativne postopke (prijavljanje na izpite, ogled lastnih evidenc ...), dostopajo do raziskovalnih podatkov ter komunicirajo s pedagogi. Študentski portal omogoča študentom dostop do vsebin glede na skupine uporabnikov (način študija, letnik, smer), podobno kot za portal za zaposlene, pa je tudi zanj značilna visoka stopnja personalizacije.

Splošne novice in obvestila. Študentom so na enem mestu na voljo splošne novice in druge tematske novice, kot npr. novice Mednarodne pisarne, razpisi štipendij ter strokovnih praks, povpraševanja podjetij, novice Centra za svetovanje in razvoj študentov ter drugih organizacijskih enot.

Dokumentni sistem. Omogoča povezave na različne dokumente, ki jih študent potrebuje med študijem (npr. pravilnike, navodila, zapisnike, različne obrazce ipd.).

Komunikacijski sistem. Omogoča komunikacijo med študenti, skupinami študentov ter med študenti in pedagogi prek osebnega poštnega predala v portalu. V okviru komunikacijskega sistema je študentom na voljo tudi splošna klepetalnica, v kateri študenti med seboj izmenjujejo mnenja (študentski glas tako doseže tudi pedagoge ob sočasnem zagotavljanju anonimnosti študentov) in posredovanje pobud in predlogov odgovornim (študenti posredujejo svoje pobude in predloge odgovornim na fakulteti ter spremljajo odgovore nanje).

Pregled lastnih predmetov. Omogoča študentom popoln nadzor nad dogajanjem pri lastnih predmetih (obveznih ali izbirnih). Pregled lastnih predmetov, ki ga lahko študent kadarkoli prilagodi svojim potrebam ali željam, predstavlja enega najpomembnejših modulov študentskega portala in med drugim omogoča povezave na spletne strani izbranih predmetov, dostop do obvestil predmetov, gradiv izbranih predmetov (npr. prosojnic predavanj, nalog za vaje, rešitev ipd.), klepetalnic pri

predmetih, preverjanje znanja pri predmetih v obliki elektronskih izpitov in samopreverjanja znanja pri predmetih ter komunikacijo med študenti izbranega predmeta.

Virtualni referat. Prek virtualnega referata so študenti neposredno povezani z delovnimi procesi v referatu za študijske zadeve. Ta jim omogoča spremljanje obvestil referata, komunikacijo z referatom, spremljanje objavljanje izpitnih rokov, prijavljanje na izpite, pregled statusa izpitnih prijav, odjavljanje od izpitov, pregled časov ter prostorov razpisanih izpitnih rokov, ogled lastnih izpitnih evidenc, vpis v višji letnih študija ipd.

Študijski viri. Na tem mestu so zbrani študijski viri, ki jih študent pri svojem študiju potrebuje. Tako je npr. navedena študijska literatura z neposredno povezavo na založništvo fakultete, viri Centralne ekonomske knjižnice, Podatkovno analitičnega središča Ekonomske fakultete (PASEF), Evropskega dokumentacijskega centra, povezave na podatkovne vire v Sloveniji ter zbirke podatkov v tujini, poleg tega pa so predstavljene tudi študentske organizacije ter dejavnosti s področja delovanja fakultete. Študentom je na voljo tudi podatkovno skladišče PASEF-a, v katerem so shranjeni zaključni računi podjetij, temelji pa na tehnologiji sprotne analitične obdelave podatkov (OLAP). Do nje lahko uporabniki transparentno dostopajo kar skozi Student-net.

Pregled aktualnega dogajanja na fakulteti. Študenti lahko sprotno pregledujejo, kaj se trenutno dogaja na fakulteti: urnik pedagoškega procesa, zagovori diplomskih del, izvajanje obštudijskih seminarjev in delavnic in druga dogajanja.

Portal za podjetja je najmlajši sklop spletne triade in je deloma še v razvoju. Osnovni namen tega portala je podpora sodelovanju med Ekonomsko fakulteto in podjetji na različnih nivojih in preko različnih fakultetnih centrov.

Komunikacija z različnimi centri fakultete. Fakulteta sodeluje z gospodarstvom prek različnih centrov (Raziskovalnega centra, Pedagoškega centra, Centra za izpopolnjevanje in svetovanje, Centra za svetovanje in razvoj študentov). Podjetjem je prek portala omogočeno neposredno komuniciranje s posameznim centrom, izmenjava sporočil ter dokumentov s centrom in samostojno posodabljanje lastnih kontaktnih podatkov.

Spremljanje poslovnega, raziskovalnega in akademskega sodelovanja. Podjetjem je na enostaven in pregleden način omogočen vpogled v stanje sodelovanja s po-

samezno fakultetno enoto (npr. diplomska dela, projekti, seminarske naloge in poslovni primeri, gostovanja na predavanjih in vajah, obiski študentov v podjetjih, seminarji in delavnice za podjetja, ki jih organizira fakulteta ipd.).

Izbira in spremljanje potencialnih kadrov. Prek dostopa do življenjepisov dodiplomskih in podiplomskih študentov lahko podjetje izbira kadre, spremlja njihovo delo in dosežke v času študija na fakulteti in neposredno komunicira z njimi. Hkrati lahko podjetja objavljajo prosta delovna mesta ali potrebe za občasna dela.

S portalom za podjetja je fakulteta stopila korak bliže podjetjem in prek enotnega vstopnega mesta omogočila celovito spremljanje medsebojnega sodelovanja, komuniciranje na različnih ravneh ter izbiro potencialnih kadrov.

4.3 Izvedba

Izvedba portalov je komponentno usmerjena s poudarjeno vlogo koristi za uporabnika. Vse kar je v portale vključeno, mora imeti za določenega uporabnika vrednost, to vrednost pa je potrebno s čim bolj smiselno uporabo tehnologije maksimizirati.

Najmanjša enota portalov Ekonomske fakultete je portlet. Pri tem portlet razumemo kot najmanjšo vsebinsko zaključeno enoto, ki jo je mogoče uporabiti na portalu. Lahko gre za:

- navadno spletno povezavo (lahko gre za statično spletno povezavo ali dinamično, ki je rezultat izvedbe programske logike – npr. povezava, ki v URL doda še posamezen posebej parameter);
- enostaven spletni obrazec (iskanje po imeniku zaposlenih, iskalnik po dokumentnem sistemu, iskalnik zadnjih novic ipd.);
- drugo spletno vsebino (npr. izpis finančnih podatkov, vendar le za področja, ki so za uporabnika aktualna in je avtoriziran za dostop do njih. V tem primeru do preverjanja pravice za dostop pride znotraj samega portleta.).

Sami portleti so nato združeni v posamezne kategorije. Kategorija je vsebinsko zaključena celota (npr. pedagoška dejavnost, raziskovalna dejavnost, osebna izkaznica, kontakti, administracija, dokumenti, novice in obvestila itd.). Posamezna skupina uporabnikov ima lahko dostop do kategorije omogočen ali ne. Vsebinske kategorije (portleti) je izbrana tako, da v danem trenutku najboljše omogočajo dosego ciljev (tako sistema, organizacije, kot tudi uporabnikov samih). Vsebinske, ki

bo uporabniku (neposredno ali prek skupine, katere član je) na voljo, določa skrbnik portala. Pri tem naj poudarimo, da gre za vsebinskega skrbnika urednika. Skrbnik izbere kategorije, ki bodo uporabniku dostopne, in tiste, ki bodo prikazane. Dostopne pomeni, da ima uporabnik pravico dostopa do kategorije in njene vsebine. Sam pa se odloči, ali bo v danem trenutku kategorijo uporabljal (prikazoval) ali ne. Skrbnik tudi smiselno (za skupine in uporabnike) razvrsti vsebino (z uporabo portletov) po posameznih kategorijah.

Ena izmed ključnih značilnosti portalov je posebitev. Portali Ekonomske fakultete so že posebljeni oziroma prilagojeni posameznim vlogam uporabnikov, poleg tega pa omogočajo tudi *avtonomno individualno posebitev*. Uporabnik lahko v skladu z zadnjima nivojema zaščite (dovoljenje za dostop do kategorij in dovoljenje za dostop do portletov) poljubno razvršča, prikazuje in izklaplja posamezne portlete in kategorije portala ter prilagaja nekatere vsebine (npr. izbira prikazanih predmetov).

Pri portalih se uporablja večnivojska zaščita dostopa in centralni nadzor. Prvi nivo zaščite je dostop do portala. Pri prijavi sistem ugotovi, katerim skupinam uporabnik ustreza. Seveda mora imeti uporabnik tudi pravico za dostop in uporabo posameznega portleta. Portleti, za katere uporabnik nima dovoljenja, se v

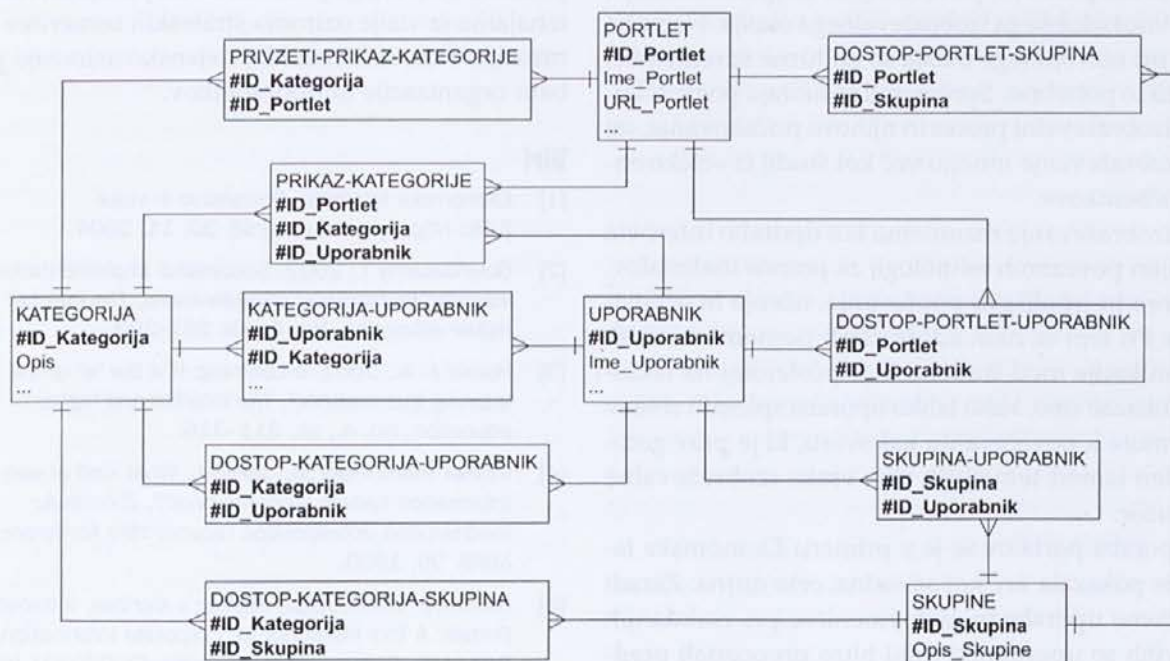
posamezni kategoriji ne pojavijo, tudi če je uporabniku dostop do nje omogočen. Na ta način lahko administrator vsebine za posamezno kategorijo predvidi več portletov, kot jih posamezni uporabnik dejansko vidi.

Pri razvoju je vseskozi upoštevana specifičnost okolja, zlasti heterogenost uporabnikov, ki do portala dostopajo z različnih lokacij, ki so različno tehnološko opremljene. Tako je uporaba portala neodvisna od uporabljenih brskalnikov in prilagojena prenosu tudi na manj zmogljivih povezavah, ki so še vedno prevladujoča oblika dostopa do interneta v Sloveniji. Tako je rešitve je moč zadovoljivo uporabljati tudi pri klicnem dostopu s hitrostjo 28.800 Kb/s.

4.4 Uporaba

Uporaba portala je zelo zaživela, saj nič več ne uporabljamo gradiv in obveščanja v pisni obliki. K temu je veliko pripomogla tudi podpora vodstva, ki aktivno sodeluje pri definiranju potreb za portal (aplikacije in vsebine) in pri stimulaciji uporabe portala med zaposlenimi. Trenutno portal za zaposlene redno uporablja 88 % zaposlenih in 83 % pedagogov.

Statistika uporabe delov portala Ekonomske fakultete, ki so že izdelani, kažejo na visoko informacijsko osveščenost uporabnikov in visoko zanimanje tako



Slika 4: Posebitvam prilagojen podatkovni model portala Ekonomske fakultete

študentov kot tudi zaposlenih za uporabo informacijske tehnologije pri vsakdanjem delu. Tako danes že več kot 98 % študentov uporablja Študent-net. V letu 2003 je vsak od njih v povprečju obiskal študentski portal 92-krat. Čeprav je portal že privzeto prilagojen posameznemu študentu, je 19 % študentov uporabilo možnost dodatne poosebitve glede na lastne potrebe in želje.

Konec leta 2003 je bila opravljena študija uporabnosti študentskega portala, v okviru katere je bila opravljena tudi primerjalna analiza s še dvema slovenskima fakultetama. Rezultati so pokazali nekaj manjših uporabnostnih problemov študentskega portala Ekonomske fakultete, ki vplivajo na učljivost. S pomočjo tipičnih scenarijev smo testirali učinkovitost izkušenih uporabnikov. Študenti Ekonomske fakultete so bili pri uporabi svojega portala bistveno učinkovitejši kot študenti drugih dveh fakultet pri uporabi njihovega. Za uspešno opravilo nalog, ki so bile določene v scenarijih, so porabili od 69 do 78 % manj časa.

5 Sklep

Vse prevečkrat smo zaradi dobaviteljev informacijsko-komunikacijske tehnologije, ki povečujejo njene zmožnosti in pomen, pod vtisom, da je le-ta samo po sebi dovolj za učinkovito in uspešno e-izobraževanje. Seveda pa informacijsko-komunikacijska tehnologija, npr. širokopasovne povezave, ni najpomembnejši vidik visokošolskega izobraževalnega okolja. Največja ovira pri razvoju tega okolja so kulturne spremembe, ki so za to potrebne. Spremeniti se morajo pogledi ljudi na izobraževalni proces in njihova pričakovanja, saj je e-izobraževanje mnogo več kot študij iz »elektronskih učbenikov«.

E-izobraževanje razumemo kot uporabo interneta in z njim povezanih tehnologij za prenos materialov, podporo in izboljšavo poučevanja, učenja in ocenjevanja. Pri tem se nam zdi še zlasti pomemben vidik komunikacije med študenti in zaposlenimi na fakulteti. Pokazali smo, kako lahko uporaba spletnih rešitev pripomore k povečevanju kakovosti, ki je prav gotovo eden izmed temeljnih cilje vsake izobraževalne institucije.

Uporaba portalov se je v primeru Ekonomske fakultete pokazala več kot smiselna, celo nujna. Zaradi enostavne uporabe in razbremenitve pri vsakdanjih opravilih so uporabniki sami hitro prepoznali prednosti, ki jih prinaša uporaba novih rešitev. Ob izdat-

ni podpora vodstva in samoiniciativnosti uporabnikov smo lahko relativno hitro zagotovili tako uporabnike, kot tudi vsebine, ki jih uporabniki sami prispevajo v portal.

Ključni dejavniki za uspešen razvoj, uvajanje in uporabo rešitve so bili:

- podpora vodstva;
- dejstvo, da smo v veliki meri uspeli zagotoviti to, da so informacije v sistem vnesene takrat, ko nastanejo, tam, kjer nastanejo – s tem smo uspeli zagotoviti ažurnost samih informacij;
- enostavnost uporabe in intuitiven uporabniški vmesnik;
- velik vpliv na produktivnost, kar je močno vplivalo na samoiniciativnost uporabnikov in njihovo aktivno sodelovanje pri razvoju sistema (v obliki predlogov, idej, kritik, komentarjev).

V prihodnosti želimo portalu dodajati nove vsebine, ki bodo vse od informacijske podpore vsakdanjim opravilom pa do integracij zunanjih podatkovnih virov. Želimo pa tudi slediti trendom v razvoju in predvsem dostopnosti tehnologije. Zlasti imamo v mislih dostop do obstoječih vsebin prek vse bolj popularnih in v ta namen uporabnih prenosnih naprav (GSM aparatov, dlančnikov).

Seveda znova ugotavljamo, da je tudi pri povečevanju kakovosti izobraževalne institucije s pomočjo portalov bistvenega pomena to, da pri njihovi zasnovi izhajamo iz vizije oziroma strateških usmeritev organizacije. Tako rešitve lahko dejansko ustrezajo potrebam organizacije in uporabnikov.

Viri

- [1] Ekonomska fakulteta: Poslanstvo in vizija [URL: <http://www.ef.uni-lj.si>], 30. 11. 2004.
- [2] Govindasamy T., 2002. Successful implementation of e-learning: Pedagogical considerations, *The internet and higher education*, No. 4., str. 287–299.
- [3] Hamid A. A., 2001. e-Learning: Is it the "e" or the learning that matters?, *The internet and higher education*, No. 4., str. 311–316.
- [4] Indihar Štemberger M., Jaklič J.: What kind of web based information system students need?, Zbornik 4. mednarodne izobraževalne računalniške konference MIRK '99, 1999.
- [5] Nielsen J. 2003. *Jakob Nielsen's Alertbox*, Intranet Portals: A Tool Metaphor for Corporate Information, [URL: <http://www.useit.com/alertbox/20030331.html>], 31. 3. 2003.

- [6] Raol M. J. et al., 2002. An identification and classification of enterprise portal functions and features, *Industrial Management & Data Systems*, Volume 102, Number 7, str. 390–399.
- [7] Sanderson P. E., 2002. E-learning: strategies for delivering knowledge in the digital age, *The internet and higher education*, No. 5., str. 185–188.
- [8] Slovenian Ministry of information society: Slovenia has reduced the lag in Internet usage compared to EU (in Slovene), [URL: <http://mid.gov.si/mid/mid.nsf/0/6327C8407CE4276BC1256E240044D504>], 23. 1. 2004.
- [9] Web Site Optimization, 2002. Web Connection Speed Trends - United States Users, [URL: <http://www.websiteoptimization.com/bw0301.html>], 31. 3. 2003.

Aleš Popovič je asistent na Ekonomski fakulteti Univerze v Ljubljani. Na dodiplomskem študiju vodi vaje na poslovno-informacijski smeri. Raziskovalno se ukvarja z modeliranjem in simulacijami poslovnih procesov ter informacijsko tehnologijo v izobraževanju. Kot sodelavec Inštituta za poslovno informatiko sodeluje na projektih prenove in informatizacije poslovanja.

Mag. Jaka Lindič je zaposlen na Ekonomski fakulteti Univerze v Ljubljani kot asistent za področje poslovne informatike. Raziskovalno se ukvarja s področjem elektronskega poslovanja, predvsem s poslovnimi modeli za elektronsko poslovanje. Sodeloval je pri številnih projektih s področja razvoja spletnih rešitev.

Doc. dr. Jurij Jaklič je predavatelj predmetov s področja poslovne informatike na Ekonomski fakulteti Univerze v Ljubljani. Je tudi sodelavec Inštituta za poslovno informatiko, v okviru katerega je sodeloval pri več aplikativnih projektih s področja strateškega načrtovanja informatike in prenove poslovnih procesov. Njegovo področje raziskovanja so zlasti ravnanje s podatki in njihova uporaba za poslovno odločanje, modeliranje in simulacija poslovnih procesov. Več let je vodil projekt razvoja portalov Ekonomske fakultete, v letu 2004 pa projekt razvoja spletne rešitve terminološkega slovarja informatike Islovar.

Doc. dr. Mojca Indihar Štemberger je docentka za poslovno informatiko na Ekonomski fakulteti v Ljubljani, kjer je leta 2000 doktorirala. Na dodiplomskem in podiplomskem študiju sodeluje kot predavateljica pri več predmetih, raziskovalno pa se ukvarja s prenovo poslovnih procesov, sistemi za podporo odločanju, e-poslovanjem ter uporabo informacijske tehnologije pri izobraževanju. Sodeluje tudi pri več aplikativnih projektih s področja prenove poslovnih procesov in strateškega načrtovanja informatike, ki jih izvaja Inštitut za poslovno informatiko na Ekonomski fakulteti. Že nekaj aktivno sodeluje pri organizaciji posvetovanja Dnevi slovenske informatike, saj je tri leta vodila organizacijski odbor, letos pa vodi programskega.

▣ Spletni portal kot informacijski vir za spodbujanje razvoja in raziskovanja e-izobraževanja v Sloveniji

Lea Bregar, Ekonomska fakulteta, Univerza v Ljubljani, lea.bregar@ef.uni-lj.si
 Margetita Zagmajster, Andragoški center RS, margetita.zagmajster@acs.si
 Marko Papič, Fakulteta za elektrotehniko, Univerza v Ljubljani, marko.papic@lfe.org

Povzetek

Spletni portali se uveljavljajo kot pomembna informacijska podpora v razvoju izobraževanja, njihov pomen pa se še povečuje. Prispevek obravnava uporabnost spletnega portala kot orodja za spodbujanje razvoja in raziskovanja e-izobraževanja v Sloveniji. Razvoj portala o e-izobraževanju je eden od podprojektov, ki so potekali v obdobju 2002–2004 v okviru projekta Poučevanje in učenje na daljavo, ki ga je financiralo Ministrstvo za šolstvo, znanost in šport. Prispevek preučuje konceptualna vprašanja spletnih izobraževalnih portalov in v tem okviru analizira vlogo spletnega portala o izobraževanju v Sloveniji. Predstavljena so tudi razvojna strategija, vsebina in tehnična podpora portala o e-izobraževanju v Sloveniji.

Abstract

Web Portal as Information Resource for Encouraging E-learning Development and Research in Slovenia

Web portals have become important information source in supporting development of education and their importance is still growing. The paper will focus on the implementation of portal as a means of encouraging development and research of e-learning in Slovenia. The development of e-learning portal is one of the sub-projects within the project Distance Learning in Slovenia run in the period 2002–2004 and financed by the Ministry of Education, Science and Sport. In the paper, the concept and the role of e-learning portal in general and in Slovenia will be discussed. The development strategy, content and technical support of e-learning portal, developed in Slovenia will also be presented.

UVOD

Nezadržen razvoj tehnologije v zadnjih letih in intenzivne družbene spremembe vse bolj uveljavljajo sodobne fleksibilne in dinamične oblike študija kot alternativo tradicionalnemu izobraževalnemu modelu. Proces uvajanja informacijske in telekomunikacijske tehnologije (ITKT) v izobraževanje se je pričel najprej na področju študija na daljavo (ŠND), v zadnjih letih pa uporaba sodobne tehnologije vse bolj prodira tudi v tradicionalno izobraževanje. Priča smo tako imenovani spremembi izobraževalne paradigme, kjer učitelj ni več izključni monopolist nad znanjem, ampak je znanje prisotno v globalnem omrežju. Razmerja udeležencev izobraževalnega procesa se v teh razmerah dramatično spreminjajo: učitelj postaja vse bolj moderator izobraževalnega procesa in svetuje učencem, kako priti do pravih informacij in kako iz teh informacij kreirati novo znanje, učenec pa dobiva v izobraževalnem procesu aktivno vlogo.

Stapljanje ŠND in tradicionalnega izobraževanja se v novih tehnoloških in družbenih razmerah izraža z

različnimi izvedbenimi inačicami novih oblik izobraževanja, ki so navadno poimenovane po svojih osnovnih značilnostih kot odprto izobraževanje (angl. open learning), fleksibilno izobraževanje (angl. flexible learning) ali porazdeljeno izobraževanje (angl. distributed learning). V zadnjem času pa se je uveljavil izraz e-izobraževanje (angl. e-learning), ki je v najsplošnejši rabi sinonim za izobraževanje, podprto z uporabo sodobne ITKT.¹ E-izobraževanje se v razvitem svetu vse bolj uveljavlja in nezadržno širi. Razvoj e-izobraževanja je povezan z inoviranjem pedagoških pristopov, študijskih pripomočkov, gradiv in orodij za izvajanje pedagoškega procesa; med slednjimi zavzemajo pomembno mesto portali.

Četudi so se portali najprej razvili v poslovnem svetu, postajajo vse bolj zanimivo sredstvo inoviranja izobraževalnih procesov.²

Namen tega prispevka je predstaviti pojem, vlogo in izhodišča za razvoj spletnih portalov v sodobnem

¹ Podrobneje o opredelitvi pojma e-izobraževanje v prispevku Bregar, L., Zagmajster M.: *Portals as a Means of Innovating Education (7th Adult Education Colloquium, 2004)*.

² Olsen navaja, da se v letnem pregledu Educause 2002 institucionalni izobraževalni portali uvrščajo kratkoročno med deset najpomembnejših tehnoloških izzivov.

izobraževanju in na tej podlagi prikazati in ovrednotiti razvojno delo pri snovanju in izvedbi spletnega portala o e-izobraževanju v Sloveniji.

Pojem in vloga izobraževalnih portalov

Kaj so pravzaprav portali? Slovar slovenskega knjižnega jezika razlaga portal kot 'arhitektonsko poudarjen vhod v stavbo: portal stare katedrale; portal palače' ..., skratka, portal pomeni širok, impresiven, jasno prepoznaven vhod v prostor. Metaforična uporaba besede *portal* v obravnavanem kontekstu torej aludira na portal kot vstopno točko, ki nam na široko odpre vrata v brezmejni prostor medmrežja. Vendar ima beseda portal v informacijskem smislu dodatno pomensko dimenzijo, ki je sam uveljavljeni pomen besede portal ne vsebuje, in sicer, da portal odpira vrata organiziranih potovanj po internetu, prirejenih potrebam posameznega obiskovalca.

To specifično pomensko lastnost e-portalov poudarjajo tudi nekateri tuji pojmovniki s področja informatike.³ Learning Circuits Glossary definira portal kot spletno mesto, ki deluje kot 'dveri' za celoten internet ali določen vsebinski segment interneta. CISCO Glossary označuje portal kot specifičen izsek spletnega mesta, ki usklajuje zahteve posameznika z razpoložljivo informacijsko ponudbo.

V splošnem povezujejo portali raznolike informacijske vire v enotno centralizirano ('one-stop') spletno stran. Na tak način predstavljajo orodje za obvladovanje nepregledne množice spletnih informacij in hkrati orientacijsko točko, ki preprečuje brezciljno tavanje in izgubljanje uporabnikov na spletu. Raznovrstnost uporabnikovih interesov seveda zahteva, da portali prilagajajo izbor informacijskih virov potrebam posameznikov in jim tako dajejo na ogled le zanje zanimive informacije (tako imenovane osebne nastavitve). Vse več portalov omogoča personalizacijo, ki pomeni posredovanje različnih osebnih podatkov. Povzamemo torej lahko, da je portal tisti spletni segment, ki združuje in povezuje informacije ter jih prikazuje na način, ki ustreza potrebam posameznikov ali skupin glede na njihove preference oz. funkcije (Dolphin, 2002).

Portali so se razvili iz spletnih strani znanih iskalnikov (na primer Yahoo, Excite; Lycos) in ponudnikov internetnih storitev (na primer AOL, Earthlink, Prodi-

gy). Vendar uporabnost portalov bistveno presega vsebino URL strani. Portal je osnovna točka, sidro ali mesto, kamor se vrača uporabnik interneta. To je hkrati mesto, kjer shranjuje informacije in od koder komunicira z drugimi (Looney in Lyman, 2000).

V praksi pogosto naletimo, da običajne spletne strani razglasaajo za portale. Za razumevanje vsebine pojma portalov je bistveno razlikovanje med pojmi internetna spletna stran, intranet in portal.⁴

Dolphin (Dolphin, 2002) opozarja, da je mejo med intranetom in portalom težko postaviti in ugotovljati, da lahko intranet označimo za portal, kadar združuje in povezuje informacije iz različnih (notranjih in zunanjih) virov.

Portale lahko v grobem obravnavamo kot splošne portale in kot specializirane portale. Splošni portali, kot na primer Yahoo, Excite, CNET, AOL, BBC portal ali Najdi.si v Sloveniji, ponujajo vrsto storitev, od različnih novic s področja, športa, vremena, prometa, do storitev elektronske pošte, klepetalnic in, najpomembnejše, iskanja informacij. Baza podatkov, ki jo je mogoče pregledovati z različnimi iskalniki, je navadno rezultat sodelovanja z upravljalcem večjih iskalnikov ali direktorjev, kot na primer FAST, Google ali Looksmart.

Specializirani portali so namenjeni posebnim interesnim skupinam (angl. community portals) ali določenim skupinam potrošnikov (angl. consumer portals). Portali potrošnikov so v glavnem komercialno usmerjeni k ciljem, da z ustrezno organizacijsko in vsebinsko prilagojeno informacijsko ponudbo pridobijo čim večji del internetnega prometa potencialnih kupcev. Portali, ki se nanašajo na specialne interesne skupine, pa običajno vsebujejo ustrezno organizirane informacije o vsebinsko zanimivih relevantnih temah (Looney in Lyman, 2000).

Izobraževalne portale najdemo tako med portali skupnosti in tudi med portali potrošnikov. Learning Circuit Glossary definira izobraževalne portale kot spletne strani, ki omogočajo učečim oz. organizacijam dostop do izobraževalnih vsebin iz različnih virov. Ponudniki izobraževalnih portalov se imenujejo tudi povezovalci vsebin (angl. aggregators), distributerji (angl. distributors) ali gostitelji (angl. hosts).

Butcher (Butcher, 2002) razvršča izobraževalne portale v tri skupine:

- povezovalni portali (angl. networking portals),

³ Slovar informatike (Islovar), ki ga je izdalo Slovensko društvo INFORMATIKA kot poskusni snopič leta 2004, ne vsebuje izraza 'portal'.

⁴ Podrobno razlago razlik med navedenimi pojmi je mogoče najti v članku: Christopher G. Connolly: *From Static Web Site to Portals*.

- organizacijski ali institucionalni portali,
- portali, usmerjeni k virom za izobraževanje (angl. resource-based portals).

Povezovalni portali zagotavljajo različnim skupinam udeležencev izobraževalnega procesa (izobraževalci, učenci, menedžerji, administrativno osebje) osrednje mesto, s katerega dostopajo do raznovrstnih izobraževalnih orodij in pripomočkov. Kot primere zanimivih izobraževalnih povezovalnih portalov naj omenimo e-learning europa, E-content village, Eduscapes.

Tipična sestavina povezovalnih portalov je rubrika viri (angl. resources), ki je pogosto razčlenjena v več kategorij, na primer povezave, novice, informacije o razvojnih dosežkih ali aktivnostih organizacije, diskusijske skupine, tehnična in strokovna pomoč. Portali navadno vsebujejo tudi originalno pripravljene slovarje, direktorije, tezavre, pojmovnike, sheme in druga informacijska orodja. Praviloma je večina informacij na portalu prosto dostopna, specializirane in bolj poglobljene informacije pa so praviloma dostopne naročnikom proti plačilu.

Med izobraževalnimi portali so najbolj pogosto zastopani organizacijski (institucionalni) portali visokošolskih izobraževalnih ustanov. Osnovna funkcija takih portalov je promoviranje in posredovanje izobraževalnih programov. Kot primere institucionalnih visokošolskih izobraževalnih portalov lahko navedemo portale: University of Washington, Open University (International Center for Distance Learning), Ekonomska fakulteta Univerze v Ljubljani.

Ti portali vsebujejo praviloma osnovne informacije o organizaciji, poslanstvu in strategiji ustanove, ustanoviteljih, kadrih; pomemben del vsebine predstavlja informacije o proizvodih in storitvah ustanove, s katerimi želijo pritegniti pozornost potencialnih kupcev oziroma uporabnikov storitev.

Portali, usmerjeni k izobraževalnim virom, omogočajo dostop do različnih spletnih informacijskih virov; praviloma vsebujejo ustrezna iskalna orodja, povezave k drugim informacijsko zanimivim spletnim stranem in možnost naročanja na izbrane storitve. Posebnost portalov, usmerjenih k virom, je ravno omejitve dostopa na podlagi plačila storitev oz. sklenitve posebnega naročniškega razmerja. V mnogih primerih

je k virom usmerjeni (plačljivi) portal sestavina povezovalnega portala (kot npr. Eduscapes).

Strategija razvoja izobraževalnih portalov

Prvi korak v razvoju portala je neodvorna opredelitev ciljev in strategije razvoja portala. Preden se torej odločimo, katere funkcionalnosti in katere storitve bo portal zagotavljal, je potrebna strateška odločitev, čemu naj pravzaprav portal služi: ali lažjemu in bolj organiziranemu dostopu do vsebin, razpoložljivih kjerkoli na spletu, ali pa organiziranemu ponujanju lastne vsebine, katere posrednik ali tvorec je skrbnik portala.

Prva možnost zadeva v glavnem povezovalne, druga pa organizacijske portale. Seveda pa v praksi odločitev za eno ali drugo možnost ni izključujoča in v procesu razvoja portalov prihaja praviloma do prepletanja oz. nadgrajevanja obeh možnosti.

Poleg te splošne dileme, ki je v končnem pogojena z osnovnim poslanstvom skrbnika portala,⁵ je treba pri načrtovanju portala opredeliti osnovne skupine uporabnikov portala. Četudi je videti na prvi pogled zamisel o oblikovanju univerzalnega portala, ki bi bil zanimiv za vsakogar (ki se ukvarja z izobraževanjem), privlačna, je vendarle treba opredeliti ciljne skupine, ki jim bo prilagojena vsebina in struktura portala. Od tu dalje je potreben še razmislek o vrsti zunanjih okoliščin, ki bodo v končnem oblikovale podobo portala (npr. stopnja razširjenosti ITKT in dostopnosti do interneta nasploh in med ciljnim skupinami, raven informacijske in internetne pismenosti, splošno poznavanje problematike, ki jo pokriva portal, razpoložljivost finančnih virov in njihova dolgoročna stabilnost, skrbništvo in polnjenje portala, pogoji dostopa do portala, avtorske pravice itd.).

Ovrednotenje vseh navedenih elementov bo v končnem determiniralo nabor funkcij oz. storitev, ki jih bo portal omogočal. V preglednici 1 prikazujemo najznačilnejše storitve izobraževalnih portalov,⁶ razvrščene glede na temeljne funkcije v tri osnovne skupine: vsebinske storitve, storitve, namenjene upravljanju z vsebino, in komunikacijske storitve.

Seveda so pogoj za učinkovito delovanje izbranih storitev portala ustrezne tehnološke rešitve in podpora.

⁵ S tega vidika je pomembno predvsem, ali so skrbniki portala javne, neprofitno usmerjene ustanove in združenja ali pa zasebna, tržno usmerjena podjetja.

⁶ Podrobneje so storitve izobraževalnih portalov opisane v Butcher (2002) in v Butters (2003).

Preglednica 1: Najznačilnejše storitve izobraževalnih portalov

Vsebina	Upravljanje z vsebino	Komunikacije
Spletni katalogi	Baze informacij	Elektronska pošta
Spletni izobraževalni programi	Taksonomija	Klepetalnice
Elektronska knjižnica	Dodajanje vsebin	Videokonference
Novice	Urejanje vsebin	Diskusijski forum
Osebnostne nastavitve	Pogoji dostopa	
Personalizacija		
Iskanje		

Razvoj portala je torej večstopenjski proces, ki se začne z opredelitvijo glavnih strateških elementov, to je opredelitvijo ciljev in poslanstva portala, ki upošteva tudi analizo okolja. Strategija se nato uresničuje z izbiro ustreznih storitev in funkcionalnosti, ki v končnem postavlja okvire potrebni tehnološki podpori.

Na podoben način smo se lotili razvoja spletnega portala e-izobraževanja v Sloveniji. V nadaljevanju najprej opisujemo strateška izhodišča, zunanje okoliščine ter vsebinsko zasnovo in izvedbo portala, prispevek pa zaključujemo s predstavitev tehničnih značilnosti portala in možnosti njegovega nadaljnjega razvoja.

Razvoj portala o e-izobraževanju v Sloveniji

Razvoj portala o e-izobraževanju⁷ je bil eden od ciljev projekta Poučevanje in učenje na daljavo (Celovita uvedba e-izobraževanja na nacionalni ravni).⁸ Poglavitni cilj projekta je bil pripraviti nacionalno strategijo za sistematično uvajanje e-izobraževanja v Sloveniji. Strategija naj bi vsebovala priporočila za nadaljnji razvoj e-izobraževanja v Sloveniji z različnih vidikov, kot so organizacijski, didaktični, ekonomski, tehnični, pravni ipd. Vsi naštetih vidiki naj bi služili kot podlaga za bolj sistematično uvajanje e-izobraževanja na različnih ravneh izobraževanja (osnovnošolsko, srednješolsko, visokošolsko in izobraževanje odraslih).

Projekt je finančno podprlo Ministrstvo za šolstvo, znanost in šport, koordiniral pa ga je Laboratorij za telekomunikacije na Fakulteti za elektrotehniko

Univerze v Ljubljani. V projekt je bilo vključenih 15 partnerjev iz Slovenije, večinoma izobraževalne ustanove ali institucije nacionalnega pomena na področju izobraževanja.⁹ Med izobraževalnimi institucijami je bilo največ visokošolskih ustanov, kar je vplivalo tudi na zasnovo in strukturo portala.

Pričakovani pozitivni učinki projekta so večje možnosti za izobraževanje s pomočjo sodobne informacijske in telekomunikacijske tehnologije, uvajanje novih fleksibilnih oblik dostopa do strokovnega znanja tako za strokovnjake kot tudi za širšo javnost in vzpostavljanje novega učinkovitega kanala za promocijo e-izobraževanja in spodbujanje inovacij v izobraževanju.

Projektne aktivnosti so potekale sočasno v trinajstih sklopih, ki so še nadalje vsebinsko razdeljeni v podsklope. Eden od podsklopov se nanaša na razvoj spletnega portala o e-izobraževanju. Portal o e-izobraževanju¹⁰ naj bi služil kot vir informacij in znanja več ciljnim skupinam, ki zajemajo tako institucije kot posameznike, ki jih zanima e-izobraževanje (udeleženci v izobraževanju; učitelji; nosilci razvoja e-izobraževanja, ki želijo osvežiti svoje znanje ali pridobiti novega za nadaljnji razvoj e-izobraževanja; vodstveni delavci, odgovorni za razvoj izobraževanja v Sloveniji itd.).

Za razvoj spletnega portala o e-izobraževanju so bile v projektu odgovorne naslednje institucije: Andragoški center Republike Slovenije in Ekonomska fakulteta Univerze v Ljubljani za vsebinski del portala ter Laboratorij za telekomunikacije, Fakulteta za elektrotehniko Univerze v Ljubljani za tehnično postavitev portala.

⁷ Informacije v portalu se nanašajo tako na ŠND kot na e-izobraževanje. Oba pojma se danes v mnogih primerih prepletata.

⁸ Projekt se je zaključil oktobra 2004.

⁹ Več informacij o projektu je na voljo na domači strani projekta <http://www.lfe.org/crp>.

¹⁰ Domača stran portala je <http://www.e-studij.net>.

Portal o e-izobraževanju bi lahko označili kot povezovalni portal z močno komponento značilnosti portala, usmerjenega k virom za izobraževanje. Glede na to usmeritev portala je bila prioriteta v razvoju vsebinskih storitev. Za razvoj portala so bili odgovorni trije partnerji projekta, vsi drugi partnerji v projektu pa so imeli možnost, da prispevajo k vsebinski graditvi portala (članki o ŠND in e-izobraževanju, naslovi zanimivih spletnih strani ipd.).

Cilj portala je omogočiti dostop do sistematično urejenega izhodišča za pridobivanje različnih informacij s področja e-izobraževanja. Nekatere so bile posebej pripravljene za potrebe portala (npr. seznam tiskanih publikacij in revij), medtem ko so bile druge zbrane z različnih mednarodnih in nacionalnih virov o e-izobraževanju. Portal je obogatila tudi zbirka informacij na spletni strani Nacionalne projektne enote za študij na daljavo - NCP (<http://www.uni-lj.si/ncp/>).¹¹ Ker je bila večina informacij NCP zbrana konec devetdesetih let, je bilo treba vse te informacije preveriti oz. posodobiti, saj mnoge povezave niso bile več aktivne ali ustrezne z vidika portala. Večina informacij v portalu je bila zbrana iz mednarodnih virov. Nekatera mednarodna združenja, v katera je včlanjen Andragoški center, redno posredujejo zanimive informacije s področja e-izobraževanja. Poleg tega so se informacije za portal zbirale tudi iz profesionalnih elektronskih mrež, v okviru katerih njihovi člani redno izmenjujejo zanimive informacije in primere dobre prakse.

Vsebina portala je bila pripravljena oz. selekcionirana z namenom, da bi zadovoljili potrebe različnih ciljnih skupin bodisi institucij bodisi posameznikov, ki jih zanimajo različni vidiki e-izobraževanja (študenti, potencialni študentje, učitelji, tutorji, strokovnjaki, ki želijo pridobiti nova znanja ali osvežiti svoje znanje, vodilno osebje v izobraževalnih institucijah, tisti, ki so odgovorni za razvoj izobraževanja v Sloveniji ipd.). Portal tudi omogoča pridobivanje informacij kot podlago za bolj poglobljeno raziskovanje in razvojno delo na različnih področjih e-izobraževanja.

Pri vsebinski zasnovi portala smo izhajali iz splošne opredelitve pojma e-izobraževanje kot izobraževanja, katerega bistvena sestavina je uporaba sodobne informacijske in telekomunikacijske tehnologije. V

portalu smo skušali zajeti pomembne vire s področja e-izobraževanja, čeprav to morda ni razvidno iz konkretnega poimenovanja posameznega spletnega naslova. Tako smo v portal uvrstili npr. institucije, ki so se razvile iz študija na daljavo, a imena niso spremenile (npr. Open University, EADTU).

Informacije v portalu so razvrščene v več vsebinskih sklopih in podsklopih:

- ponudniki: izobraževalne ustanove, ki ponujajo e-izobraževanje v Sloveniji in tujini;
- združenja: mednarodna in nacionalna združenja na področju e-izobraževanja;
- publikacije (povezave na elektronske publikacije in seznam tiskanih publikacij);
- elektronske revije specializirane za ŠND in s prispevki o ŠND;
- tiskane revije specializirane za ŠND in s prispevki o ŠND;
- članki (slovenski in tuji);
- konference, povezane z e-izobraževanjem, v letih 2003, 2004 in 2005;
- viri informacij: o e-izobraževanju na internetu (portali, diskusijske skupine);
- razvoj: informacije potrebne za razvoj e-izobraževanja (zanimivi projekti, demo spletni učbeniki, informacije o programih usposabljanja in izobraževanja s področja e-izobraževanja, slovarji in pojmovniki);
- specializirane teme: povezave na specializirana vsebinska področja, kot so razvoj programov, študijska podpora, menedžment, kakovost in ostalo;
- domače iniciative na področju e-izobraževanja (gre večinoma za različne projekte s področja e-izobraževanja; v okviru nekaterih razvijajo tudi spletne učbenike oz. krajše izobraževalne programe).

Poleg vsebinskih storitev omogoča portal minimum komunikacijskih storitev. Obiskovalci portala lahko kontaktirajočasne skrbnike portala, pridobivajo ustrezne novice, se vključujejo v diskusijski forum in iščejo določene informacije.

Razvoj portala o e-izobraževanju je potekal v letih 2003 in 2004. V letu 2003 je bila izvedena prva faza projekta, v kateri so bile opravljene naslednje aktivnosti: opredelitev strategije razvoja portala z analizo okolja, določitev funkcionalnosti portala, specifikacija

¹¹ Nacionalna projektna enota za študij na daljavo (NCP) je bila nacionalni koordinator na področju študija na daljavo v okviru programa Phar (Phare Multi-country Programme for Distance Education) v drugi polovici devetdesetih let. NCP je imel sedež na Ekonomski fakulteti Univerze v Ljubljani do konca leta 2001. V letih 2002 in 2003 so se nekatere aktivnosti NCP izvajale na Andragoškem centru Republike Slovenije.

vsebinskih zahtev portala, specifikacija tehničnih zahtev portala, proučitev možnosti izvedbe znotraj obstoječega portala www.ltfe.org in predlog tehnične rešitve portala, tehnična postavitev portala, uskladitev vsebinske zasnove s tehnično postavitvijo portala ter poskusno polnjenje portala.

V letu 2004 so odgovorni partnerji v projektu preverjali funkcionalnost portala in predlagali tehnične in vsebinske spremembe in dopolnitve. Na podlagi usklajenega dogovora odgovornih partnerjev je Laboratorij za telekomunikacije izpeljal večino predlaganih tehničnih rešitev. Portal je bil tudi vsebinsko dopolnjen (število spletnih povezav in informacij je od 300 v letu 2003 naraslo na 582 v jeseni 2004).

V preglednici 2 prikazujemo strukturo portala in število spletnih povezav oz. informacij po posameznih vsebinskih sklopih oz. podsklopih portala.

Preglednica 2: Število spletnih povezav/informacijskih enot po posameznih sklopih in podsklopih v portalu o e-izobraževanju (<http://www.e-studij.net>)

Vsebinski sklop	Vsebinski podsklop	Število povezav oz. informacij
1. Ponudniki	domači	2
	tujji	90
2. Združenja	nacionalna	11
	mednarodna	20
3. Publikacije	elektronske	16
	tiskane	28
4. Elektronske revije	specializirane za ŠND	33
	s prispevki o ŠND	38
5. Tiskane revije	specializirane za ŠND	30
	s prispevki o ŠND	18
6. Članki	slovenski	0
	drugi	21
7. Konference	2003	10
	2004	23
	2005	14
8. Viri informacij	portali	56
	diskusijske skupine	12
9. Razvoj	projekti	31
	demo spletni učbeniki	9
	izobraževanje	30
	slovarji in pojmovniki	8
10. Specializirane teme	razvoj programov	13
	študijska podpora	6
	menedžment	3
	kakovost	19
	ostalo	17
11. Domače iniciative		24
SKUPAJ		582

Tehnološka podpora

Pri snovanju tehnološke rešitve spletnega portala smo izhajali iz osnovnih izhodišč strategije razvoja in finančnih omejitev projekta. To je zahtevalo učinkovito implementacijo z upoštevanjem naslednjih usmeritev:

- tehnološka rešitev je morala biti robustna ter nadgradljiva in odprta za enostavno dodajanje novih funkcionalnosti;
- tehnološka rešitev je zahtevala enostavno administriranje in urejanje.

Pri razvoju portala smo upoštevali tudi druge zahteve, ki so značilne za implementacijo spletnih aplikacij in bodo opisane v nadaljevanju prispevka.

Pristop k razvoju tehnološke rešitve portala je temeljil na izkušnjah in znanju, ki smo ga v Laboratoriju za telekomunikacije Fakultete za elektrotehniko v preteklosti pridobili z razvojem kompleksnih spletnih aplikacij.¹² Izhajali smo iz splošnega tehnološkega koncepta internetnih portalov (Zebec, 2001).

Razvoj tehnološke rešitve je potekal v štirih fazah:

- funkcionalna specifikacija in načrtovanje podatkovnega modela,
- implementacija tehnološke rešitve,
- testiranje,
- evalvacija.

Za uspešno implementacijo je bila najbolj pomembna prva faza razvoja portala, v kateri so bile izbrane tehnologije portala.

Izhodišče za izbor tehnologij portala so predstavljali osnovni koncept portala in njegova funkcionalna opredelitev. Tehnologije portala so sestavljali aplikacijski, spletni in podatkovni strežnik.

Pri izboru tehnologij smo upoštevali še predvidene obremenitve portala ter zahteve po hitrem in enostavnem razvoju rešitve. Zaradi enostavnosti ter možnosti učinkovite nadgradnje so bile izbrane rešitve komercialnega ponudnika Microsoft. Celotna rešitev deluje na strežniku Microsoft 2000 Server. Strežnik je bil izbran zaradi možnosti uporabe storitev indeksiranja in iskanja, ki so že vključene v rešitev. Še bolj pomemben razlog je bil predvidena migracija na strežnik Microsoft server 2003, ki kot del funkcionalnosti portala omogoča po potrebi enostavno vključitev številnih storitev. Kot spletni strežnik je bil uporabljen

¹² Te aplikacije so predstavljene na portalu o telekomunikacijah in tehnologijah Laboratorija za telekomunikacije na Fakulteti za elektrotehniko (<http://www.ltfe.org>), na spletni strani E-CHO: internetna platforma za e-izobraževanje (<http://dl.ltfe.org/>) in na spletni strani Laboratorija za telekomunikacije na Fakulteti za elektrotehniko (<http://www-lt.fe.uni-lj.si/>).

Microsoft IIS (Internet Information Server). Relacijska baza podatkov je bila implementirana s programom Microsoft Access 2000. Razlog za implementacijo v omenjenem programu je predvsem izjemna enostavnost implementacije. Po drugi strani pa predvideno število uporabnikov portala v začetni fazi ne zahteva kompleksnejše baze podatkov. Program omogoča tudi sorazmerno preprosto migracijo na zmogljivejši relacijski podatkovni strežnik Microsoft SQL 2000.

V drugi fazi razvoja portala so bile izvedene naslednje aktivnosti: postavitve tehnološke infrastrukture (strežniki), implementacija poslovne logike in implementacija relacijske baze podatkov.

Implementacija poslovne logike je bila izvedena s pomočjo tehnologije ASP (Active Server Pages), ki se izvaja na IIS spletnem strežniku. Poslovna logika skrbi predvsem za:

- obdelavo zahtev in podatkov uporabnikov ter upravljanje s podatki (branje, vpis in urejanje podatkov v bazi);
- procesiranje odzivov (izdelava dinamičnih spletnih strani za uporabnike portala);
- personalizacijo (prilagoditev vsebine in storitev portala, uporaba skritih polj, sej in piškotkov).

Pomemben poudarek je bil tudi na implementaciji administracije portala – urejanju vsebine. Celotno urejanje se izvaja s pomočjo spletnega vmesnika, ki omogoča nadzorovano, enostavno urejanje vsebine z naprednimi možnostmi dodajanja šifrantov, dodatnih polj v tabelah s podatki ipd. Datoteke, kot so članki, prispevki, predstavitev, je možno na strežnik nalagati s pomočjo spletnega vmesnika, prenos pa poteka preko FTP (File Transfer Protocol) protokola.

Veliko število informacij, ki so dostopne na portalu, zahteva strukturiran in sistematski dostop, kar smo dosegli z domišljeno izdelavo funkcionalne specifikacije. Ob tem so zelo pomembni mehanizmi iskanja po vsebini portala. V fazi implementacije je bilo potrebno implementirati dve vrsti iskanja: po podatkih v bazi ter po vsebini datotek (člankov, prispevkov in predstavitev). V prvem primeru je bil implementiran iskalni mehanizem na dveh ravneh, in sicer na nivoju celotne baze podatkov ter na nivoju posameznih funkcionalnih komponent portala. Za iskanje po vsebini je bil integriran že obstoječi programski modul, ki omogoča iskanje po vsebini .pdf, .doc, .ppt ter drugih tipov dokumentov.

Testiranje portala je bilo opravljeno skupaj s partnerji v projektu, ki so predvsem ocenjevali funkcional-

nost portala in preprostost njegove uporabe. Testna uporaba portala je prinesla tudi precej izboljšav in dopolnitev z vidika predstavitve vsebine portala.

Perspektive

Portal je bil zasnovan z namenom, da bo redno vzdrževan in polnjen z dodatnimi informacijami (s kratkim opisom informacij, ki so na voljo v portalu, kot npr. vloga posameznega mednarodnega združenja ipd.). Sedaj vsebuje portal skoraj 600 spletnih povezav oz. informacij, vendar pa to še zdaleč ne zajema vseh relevantnih informacij s tega področja (npr. niso zajete vse relevantne domače iniciative). Poleg tega se ves čas pojavljajo nove informacije, ki bi bile zanimive za objavo v portalu, kot npr. izid novih publikacij, objava novih konferenc, izvajanje novih nacionalnih in tujih projektov, pojav novih ponudnikov programov e-izobraževanja ipd. Redno vzdrževanje portala zahteva tudi preverjanje obstoječih spletnih povezav, saj nekatere spletne povezave čez čas niso več aktivne ali pa se spremeni njihova namembnost (npr. neka institucija, ki smo jo uvrstili med portale, preneha delovati kot portal, če niso zagotovljena redna finančna sredstva in reducira svojo dejavnost na svetovanje).

Redno vzdrževanje portala zahteva določena finančna sredstva, ki pa zaenkrat za nadaljevanje dela na portalu še niso bila zagotovljena. Bodočnost portala je v veliki meri odvisna od dogovora med ustreznimi partnerji o potrebnosti tovrstnega portala v Sloveniji in rednega dotoka finančnih sredstev, kar bi omogočalo tekoče vzdrževanje portala in tehnično spopolnjevanje.

Možnosti za tehnično izboljšavo portala vidimo predvsem v prilagajanju funkcionalnosti portala zahtevam tako imenovanega uporabnika naslednje generacije, ki ima ob osnovni zahtevi po dostopu informacij na portalu tudi dodatne zahteve (po mobilnosti, enakih pogojih dostopnosti, personalizaciji in večjezikovnosti). Uresničevanje teh zahtev predpostavlja integrirano, kompleksnejšo in hkrati preprosto tehnično rešitev.

Mobilnost pomeni povečanje dostopnosti informacij portala tudi iz različnih mobilnih terminalov, kar je že danes zahteva uporabnikov mlajše generacije.

Enaka dostopnost za vse in uporabnost portala bo prav tako temelj bodočega razvoja. Številne raziskave kažejo, da se povprečna starost uporabnikov veča, nove generacije starejših pa bodo dobro seznanjene z

uporabo sodobnih tehnologij. V ta namen bo potrebno razviti tehnično rešitev, ki bo omogočala uporabo portala za različne ciljne skupine uporabnikov. Izbrana tehnološka rešitev omogoča migracijo na nove tehnologije, predvsem XML (eXtensible Markup Language), XSL (eXtensible Stylesheet Language) in CSS (Cascading Style Sheets), ki omogočajo raznovrstne predstavitve brez potrebe po izdelavi različnih oblik enakega dokumenta.

Personalizacija se ob prilagajanju vsebine nanaša tudi na predstavitev vsebine uporabnikom s posebnimi potrebami (npr. slepi in slabovidni, uporabniki s slabimi motoričnimi sposobnostmi). Ob uporabi primernih XSL predlog je mogoče predstavitev slik zamenjati s tekstovnimi opisi ter na ta način omogočiti uporabo tekstovnih brskalnikov, povezanih s sistemi za sintetiziranje govora TTS (Text-To-Speech System) ali uporabo Braillovih prikazovalnikov. Podobno omogočajo XSL predloge prikaz kontrastnih grafičnih vmesnikov, vmesnikov z večjim besedilom in ikonami, ki uporabnikom z motoričnimi težavami omogočajo lažjo navigacijo in iskanje informacij po portalu.

Ne nazadnje je pomembna lastnost portala večjezičnost vmesnika in vsebine. Podpora večjezičnosti je mogoče doseči s shranjevanjem besedil, opisov ikon in slik grafičnega vmesnika različnih jezikov v posebne XML datoteke.

Viri in literatura

Bregar, L.; Zagmajster M.:

Portals as a Means of Innovating Education (Adult Education Colloquium 2003. Current issues in adult learning and motivation. 7th Adult Education Colloquium; editors: Marko Radovan, Neda Đorđević. Ljubljana : Andragoški center Republike Slovenije = Slovenian Institute for Adult Education, 2004.

Butcher, Neil:

Best Practice in Education Portals. Research Document Prepared for the Commonwealth of Learning and School Net Africa. Commonwealth of Learning. October, 2002.
<http://www.col.org/Consultancies/Q2EducationPortals.htm> (10. 02. 2003).

Butters, Geoff:

What Features in a Portal? Ariadne Issue 35, 2003.
<http://www.ariadne.ac.uk/issue35/butters/> (1. 09. 2003).

CISCO. Internet Learning Solutions Group E-learning Glossary.
http://www.cisco.com/warp/public/10/wwtraining/elearning/pdf/elearn_glossary.pdf (12. 09. 2003).

Conolly C., Christopher:

From Static Web Site to Portal, Villanova University from EDUCAUSE Review
<http://faculty.weber.edu/deisler/syllabus2002.htm> (12. 09. 2003).

Dolphin, Ian; Miller, Paul in Sherratt, Robert:

Portals, Portals, Everywhere. Ariadne Issue 33, 2002.
<http://www.ariadne.ac.uk/issue33/portals/>

E-CHO: internetna platforma za e izobraževanje.

<http://dl.ltfe.org/> (1. 12. 2004)

Glossary of Technical and Distance Education Terms.

<http://www.knowledgeability.biz/weblearning/> (10. 09. 2003).

Islovar. Slovar informatike. Slovensko društvo

INFORMATIKA. Poskusni snopič. Ljubljana, 2004.

Kaplan-Leiserson, E. E-Learning Glossary. 2002

<http://www.learningcircuits.org/glossary.html> (9. 01. 2002).

Laboratorij za telekomunikacije na Fakulteti za elektrotehniko.

<http://www-lt.fe.uni-lj.si/>. (1. 12. 2004)

Looney, Michael in Lyman, Peter:

Portals in Higher Education, EDUCAUSE Review, July/August, 2000.
<http://faculty.weber.edu/deisler/syllabus2002.htm> (25. 08. 2003).

Norman N., Michele:

Portal Technology: Into the Looking Glass, Converge Magazine. Special Publication, 2003.

Olsen, Florence:

The Power of Portals. The Chronicle of Higher Education. August, 2002.

Portal o telekomunikacijah in informacijskih tehnologijah.

<http://www.ltfe.org/> (1. 12. 2004)

JISC. Joint Information Systems Committee. UK Council for Higher Education.

<http://www.jisc.ac.uk/dner/development/lestategy.html>, (25. 08. 2003).

Project Distance Learning in Slovenia

www.ltfe.org/crp/ (5. 09. 2003).

Project Proposal Distance Learning in Slovenia (Poučevanje in učenje na daljavo, Celovita uvedba e-izobraževanja na nacionalni ravni). Laboratory for Telecommunications, Faculty for Electrical Engineering, University of Ljubljana). 2002.

Zebec, Luka:
Internetni portali; Diplomsko delo, Fakulteta za elektrotehniko, Ljubljana 2001.

Lea Bregar je doktorica ekonomskih znanosti, zaposlena na Univerzi v Ljubljani, kjer kot izredna profesorica poučuje statistiko na Ekonomski fakulteti. Bila je pobudnica in vodja projekta uvajanja študija na daljavo na Ekonomski fakulteti v Ljubljani in vodja prve faze projekta Phare programa študija na daljavo v Sloveniji. Je avtorica številnih člankov s področja študija na daljavo, predstavljenih na srečanjih strokovnjakov v Sloveniji in v tujini ter objavljenih v domačih in tujih strokovnih ter znanstvenih publikacijah. Razvila je vrsto študijskih gradiv s področja statistike med njimi tudi spletni učbenik ekonomske statistike v slovenskem in angleškem jeziku. Leta 1999 je za svoje delo na področju študija na daljavo prejela Sokratovo odličje za visokošolsko didaktiko.

Mag. Margerita Zagmajster se že nekaj let ukvarja s študijem na daljavo. Od leta 1991 do 1993 je na Centru za razvoj univerze delala kot raziskovalka v projektu Uvajanje visokošolskega študija na daljavo na slovenske visokošolske ustanove. Leta 1994 se je zaposlila na Nacionalni projektni enoti za študij na daljavo (NCP) na Ekonomski fakulteti. Konec leta 1997 je bila imenovana za vodjo NCP. Od začetka leta 2002 je zaposlena na Andragoškem centru Republike Slovenije, kjer se ukvarja z e-izobraževanjem. Od aprila 2003 dalje je namestnica direktorice. Je avtorica vrste člankov s področja študija na daljavo in e-izobraževanja, objavljenih oz. predstavljenih na konferencah doma in v tujini.

Marko Papič je diplomiral leta 2000 in magistriral leta 2003 na Fakulteti za elektrotehniko Univerze v Ljubljani s temo Celovite rešitve e-izobraževanja. Že od leta 1997 dela v Laboratoriju za telekomunikacije na Fakulteti za elektrotehniko Univerze v Ljubljani (LTFE). Njegovo pedagoško, raziskovalno in razvojno delo je povezano z načrtovanjem in implementacijo internetnih aplikacij. Je avtor več strokovnih prispevkov in soavtor sistema za upravljanje e-izobraževanja in upravljanje izobraževalnih vsebin E-CHO, razvitega na LTFE. Aktivno sodeluje tudi pri izvedbi več razvojno-raziskovalnih projektov in aplikativnih projektov, povezanih z industrijo, in pri izobraževalnih aktivnostih LTFE.

Uporaba podatkovnih skladišč v univerzitetnem okolju: primer Univerze v Ljubljani

Viljan Mahnič, Marko Pozenel

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Tržaška 25, Ljubljana

Viljan.Mahnic@fri.uni-lj.si, Marko.Pozenel@fri.uni-lj.si

Povzetek

V članku opisujemo projekt izgradnje podatkovnega skladišča, ki poteka na Univerzi v Ljubljani z namenom, da bi vzpostavili enoten vir podatkov za razne analize študijskega procesa. V prvem delu članka je opisan namen projekta in arhitektura predlaganega podatkovnega skladišča. Nato sledi podrobna predstavitev dimenzijskih podatkovnih modelov za posamezna področna podatkovna skladišča in opis orodij, ki jih uporabljamo za realizacijo. Na koncu podajamo še pregled možnih analiz in nekaj tipičnih vzorcev poročil, ki so na voljo uporabnikom.

Ključne besede: podatkovno skladišče, področno podatkovno skladišče, dimenzijski podatkovni model.

Abstract

Data warehousing in university environment: the case of the University of Ljubljana

We describe a data warehouse project that has been started at the University of Ljubljana with the aim of obtaining unified and integrated source of data for various analyses of the educational process. A description of the scope of the project and the overall data warehouse design is first given, followed by a detailed description of the underlying dimensional model and implementation tools. Finally, a survey of possible analyses is given and some typical end user reports are described.

Keywords: data warehouse, data mart, dimensional model.

1 Uvod

Od začetka devetdesetih let prejšnjega stoletja, ko je Bill Inmon [6] izoblikoval koncept podatkovnih skladišč, je skladiščenje podatkov postalo ena izmed najpomembnejših vej na področju informacijskih sistemov [2]. Mnoga velika podjetja so pričela graditi obsežna podatkovna skladišča, ki služijo kot vir podatkov za sisteme za podporo odločanju, in mnogim izmed njih se investicije bogato obrestujejo [17].

Tudi v univerzitetnih okoljih smo priča različnim projektom s tega področja [14, 1, 5, 3, 11]. Medtem ko Stevenson [14] opisuje predvsem izkušnje in kritične dejavnike uspeha pri izgradnji podatkovnega skladišča na eni izmed avstralskih univerz, je v [1] predstavljen vsedržavni projekt izgradnje podatkovnega skladišča za potrebe francoskih univerz. V [3] je osrednja pozornost namenjena zgodnejšim fazam v življenjskem ciklu razvoja podatkovnega skladišča, ostali projekti pa obravnavajo podatkovna skladišča za posamezna področja. Tako je v [5] predstavljeno podatkovno skladišče za spremljanje raziskovalne dejavnosti, v [11] pa podatkovno skladišče za analizo učinkovitosti e-izobraževanja.

V prispevku bomo predstavili projekt izgradnje podatkovnega skladišča, ki poteka na Univerzi v Ljubljani z namenom, da bi vzpostavili enoten vir podatkov za razne analize študijskega procesa. Članek je sestavljen iz petih delov. V drugem delu opisujemo namen projekta in arhitekturo celotnega podatkovnega skladišča. V tretjem in četrtem delu sta podrobno predstavljene področni podatkovni skladišči s podatki o vpisanih študentih in diplomantih, ki sta bili realizirani najprej. V petem delu so opisana orodja, ki smo jih uporabili pri realizaciji, v šestem delu pa je podan pregled možnih analiz in nekaj tipičnih vzorcev poročil, ki so na voljo uporabnikom.

2 Namen projekta in zasnova podatkovnega skladišča

Končni cilj našega projekta je izgradnja celovitega podatkovnega skladišča za področje študijske informatike, ki bo sestavljeno iz verige štirih področnih podatkovnih skladišč (angl. data marts): področnega podatkovnega skladišča za analizo prijav za vpis,

področnega podatkovnega skladišča za analizo vpisnih podatkov, področnega podatkovnega skladišča za analizo opravljanja študijskih obveznosti in področnega podatkovnega skladišča za analizo podatkov o diplomantih [10]. Ker je Univerza v Ljubljani sestavljena iz 26 članic, ki imajo vsaka svoj informacijski sistem, je za izgradnjo podatkovnega skladišča potrebno najprej izločiti ustrezne podatke iz informacijskih sistemov posameznih članic, jih prečistiti in preoblikovati ter nato naložiti v posamezna področna podatkovna skladišča. Arhitektura predlaganega podatkovnega skladišča je prikazana na sliki 1.

Za realizacijo smo izbrali strategijo "od spodaj navzgor" [16, 9, 15], ki omogoča postopno izgradnjo posameznih področnih podatkovnih skladišč, ki jih nato povežemo v celovito rešitev s pomočjo med seboj skladnih skupnih dimenzijskih tabel (angl. conformed dimensions). V primerjavi s strategijo "od zgoraj navzdol", ki jo zagovarja Inmon [7], nam ta pristop omogoča, da hitreje, z manj stroški in manj tveganja pridemo do otipljivih rezultatov. Seveda pa je treba že od začetka posvetiti posebno pozornost načrtovanju skupnih dimenzijskih tabel [10].

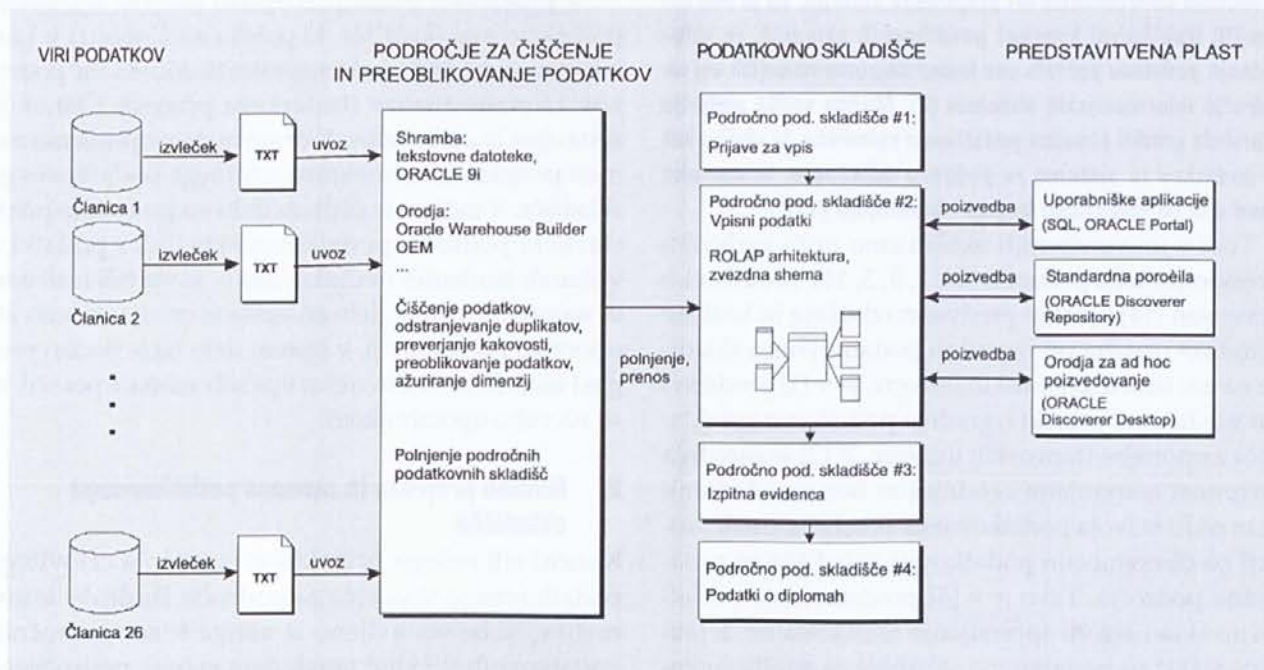
Izgradnjo celotnega podatkovnega skladišča smo tako zasnovali kot iterativen proces, v okviru katerega je moč priti do končne rešitve s pomočjo manjših 4-8-mesečnih projektov, ki dajejo otipljive rezultate

takoj in (vsaj v začetni fazi) temeljijo na dobro definiranih poslovnih procesih s stabilnim virom podatkov [9]. V skladu s temi priporočili smo najprej realizirali področno podatkovno skladišče za analizo vpisnih podatkov. To področje smo izbrali iz dveh razlogov:

- Obstaja enoten vir podatkov – obrazec "Vpisni list", ki je za vse članice enak, čeprav ga posamezne članice obdelujejo na različne načine in z različnimi programi.
- Na ravni univerze je bila jasno izražena potreba po pregledu nad stanjem vpisa na posameznih članicah.

Na ta način smo dosegli, da je bil projekt na začetku osredotočen na en sam poslovni proces z dobro definiranim virom podatkov, ki lahko pomembno prispevajo k boljšemu poslovanju univerze.

Nato smo realizirali še področno podatkovno skladišče za analizo podatkov o diplomantih, medtem ko je realizacija preostalih področnih podatkovnih skladišč (za analizo prijav za vpis in opravljenih študijskih obveznosti) predvidena v nadaljevanju projekta. Ocenjujemo, da realizacija področnega podatkovnega skladišča s podatki o prijavah za vpis ne bi smela predstavljati velikih težav, saj gre za poslovni proces, ki je strogo centraliziran in enoten za celotno univerzo. Veliko več problemov pa pričakujemo pri realizaciji področnega podatkovnega skladišča za spremljanje



Slika 1: Arhitektura podatkovnega skladišča

opravljanja študijskih obveznosti, saj bodo tu prišle do izraza vse specifičnosti posameznih članic.

Za realizacijo podatkovnega skladišča smo izbrali orodja podjetja Oracle. Podatke iz informacijskih sistemov posameznih članic dobimo v obliki tekstovnih datotek, ki jih očistimo, preoblikujemo in naložimo v podatkovno skladišče s pomočjo orodja Oracle Warehouse Builder [12]. Za dostop do podatkov uporabljamo Oracle Discoverer [13], zahtevnejša poročila pa smo sprogramirali v programskem jeziku PL/SQL.

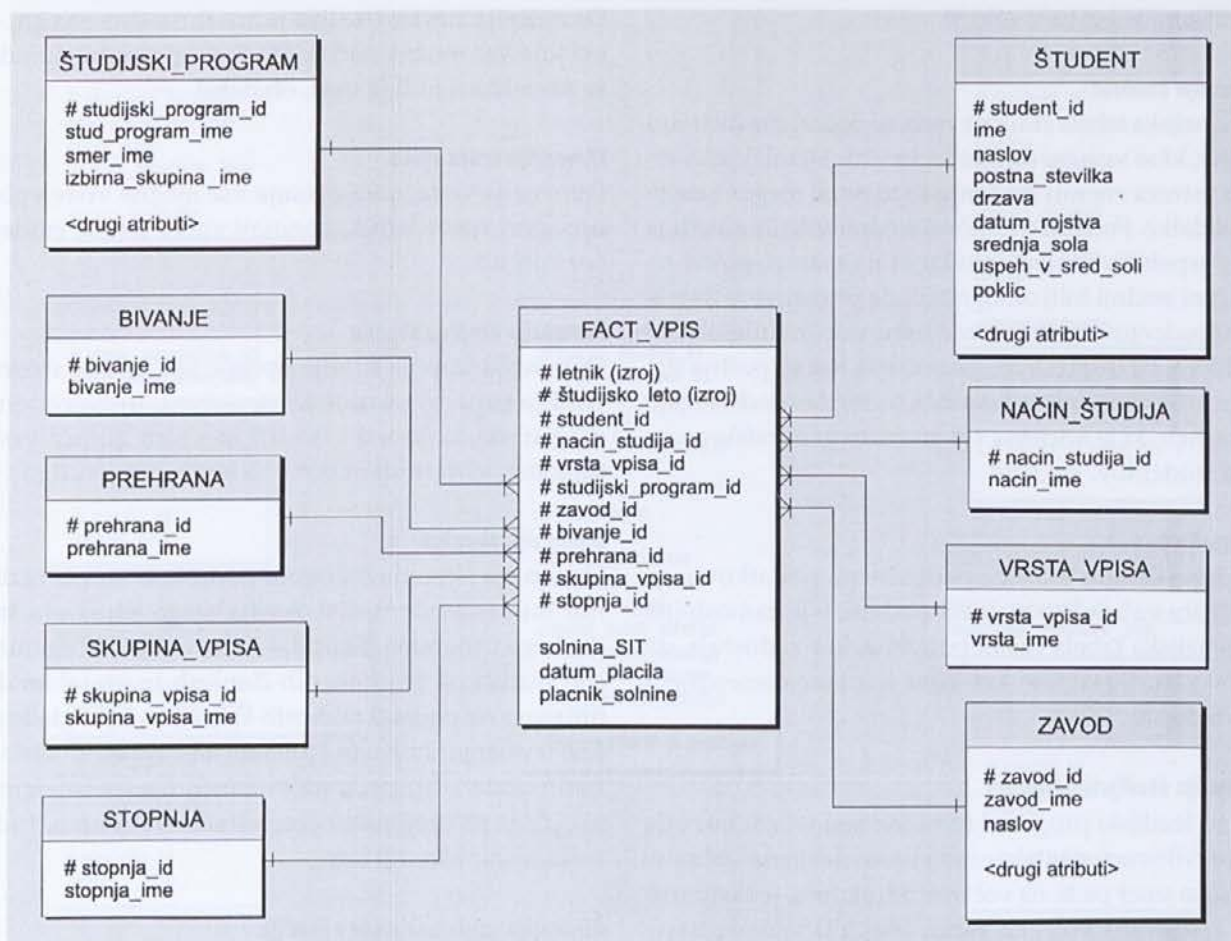
3 Področno podatkovno skladišče za analizo vpisnih podatkov

Logični načrt področnega podatkovnega skladišča za analizo vpisnih podatkov prikazuje zvezdna shema na sliki 2. Sestavljajo jo tabela dejstev in 11 dimenzijskih tabel. Za zrno tabele dejstev smo izbrali vpis enega študenta, ki je prikazan kot dogodek na prese-

čišču naslednjih dimenzij: študent, čas (tj. študijsko leto), zavod (tj. fakulteta, akademija ali visoka šola, na katero se je študent vpisal), študijski program, letnik, način študija (redni ali izredni), stopnja študija (visokošolski strokovni, univerzitetni, magistrski, doktorski), skupina (če so študenti v okviru istega letnika in študijskega programa razdeljeni na več skupin), vrsta vpisa, prehrana v času študija in bivanje v času študija. Dimenziji čas in letnik nimata atributov, zato sta predstavljeni kot izrojene dimenziji (angl. degenerate dimensions). Zanju obstaja samo ključ v tabeli dejstev, ne pripada pa jima samostojna dimenzijska tabela.

Tabela dejstev

Tabela dejstev predstavlja množico razmerij M:N med prej omenjenimi dimenzijami, vsebuje pa samo dve merljivi dejstvi: znesek šolnine in datum, ko je bila šol-



Slika 2: Zvezdna shema področnega podatkovnega skladišča za analizo podatkov o vpisu

nina plačana. To pomeni, da večina aplikacij, ki uporabljajo to področno podatkovno skladišče, izvaja samo štetje zapisov. Kljub majhnemu številu merljivih dejstev (tabele dejstev, ki ponazarjajo dogodke, običajno nimajo merljivih dejstev) pa lahko s pomočjo te tabele dobimo odgovore na številna vprašanja, npr.:

- Koliko študentov se je vpisalo na posamezno članico ali v posamezen študijski program?
- Kakšna je struktura vpisanih študentov glede na končano srednjo šolo, poklic (ki ga imajo po končani srednji šoli), uspeh v srednji šoli, način študija, vrsto vpisa ipd.?
- Se število študentov na posameznih članicah oziroma študijskih programih povečuje ali zmanjšuje?
- Kako napreduje v višje letnike izbrana generacija študentov?
- Iz katerih krajev prihajajo študentje, ki so vpisani na posamezno članico ali študijski program?
- Kje študentje prebivajo in se hranijo v času študija (v študentskem domu, pri starših, v podnajemniškem stanovanju itd.)?

Dimenzija študent

Dimenzijska tabela študent vsebuje podatke o vseh študentih, ki so vpisani na Univerzo v Ljubljani. Vsaka vrstica ustreza enemu študentu in vsebuje njegove osebne podatke. Podatki o končani srednji šoli (ime srednje šole, uspeh v zadnjem letniku in na maturi, poklic po končani srednji šoli) omogočajo, da primerjamo uspešnost študentov na univerzi z njihovim srednješolskim profilom. Po drugi strani pa podatki, kot so poštna številka, občina stalnega bivališča in regija predstavljajo hierarhijo, ki je koristna pri analizah geografskega izvora študentov.

Dimenzija čas

Gledano samo s stališča področnega podatkovnega skladišča za analizo vpisnih podatkov je samostojna dimenzijska tabela čas nepotrebna, ker zadostuje, da imamo študijsko leto kot ključ izrojene dimenzije v tabeli dejstev.

Dimenzija študijski program

Ker so študijski programi navadno sestavljeni tako, da se (praviloma v višjih letnikih) lahko delijo na več smeri, vsaka smer pa še na več izbirnih skupin, je ta dimenzija zasnovana tako, da vsaka vrstica tabele predstavlja eno izbirno skupino. Več izbirnih skupin, ki pripadajo isti smeri, lahko povežemo med seboj na nivoju

smeri, več smeri, ki pripadajo istemu študijskemu programu, pa na nivoju študijskega programa. Opisana hierarhija študijski program – smer – izbirna skupina omogoča enostavno izdelavo poročil na različnih ravneh podrobnosti, ki je v terminologiji podatkovnih skladišč znana pod imenom vrtanje navzdol (angl. drill down) oziroma vrtanje navzgor (angl. drill up).

Dimenzija zavod

Dimenzija zavod opisuje posamezne članice ljubljanske univerze. Trenutno univerzo sestavlja 26 članic.

Dimenzija letnik

Dimenzija letnik je izrojena dimenzija, saj je atribut letnik študija, ki je sestavni del primarnega ključa v tabeli dejstev, edini atribut te dimenzije. Uporablja se kot ključ za grupiranje tistih študentov, ki so vpisani v isti letnik.

Dimenzija način študija

Dimenzija način študija je majhna dimenzija, ki opisuje vse možne načine študija, npr. redni, izredni (v preteklosti tudi iz dela, ob delu).

Dimenzija vrsta vpisa

Dimenzija vrsta vpisa opisuje vse možne vrste vpisa, npr. prvi vpis v letnik, ponovni vpis v letnik, evidenčni vpis itd.

Dimenzija stopnja študija

Dimenzija stopnja študija opisuje vse možne stopnje študija (npr. visokošolski strokovni, univerzitetni, magistrski, doktorski študij) in s tem določa vrsto diplome, ki jo študent dobi po končanem študiju.

Dimenzija skupina

Dimenzija skupina omogoča podrobnejši opis različnih skupin študentov v okviru istega letnika in študijskega programa. Skupine so odvisne od organizacije študija na posameznih članicah in jih lahko definiramo na podlagi različnih kriterijev, npr. glede na kraj izvajanja študija (v Ljubljani ali v različnih dislociranih oddelkih), glede na vrsto študija (redni, izredni), glede na predhodni potek študija (običajno študij, nadaljevalni študij) itd.

Dimenzija prehrana v času študija

Dimenzija prehrana v času študija opisuje različne možnosti, ki so na voljo študentom za prehranjevanje

v času študija, npr. v študentskem domu, pri starših, v lastnem gospodinjstvu itd.

Dimenzija bivanje v času študija

Dimenzija bivanja v času študija opisuje različne vrste nastanitve v času študija, npr. v študentskem domu, pri starših, v podnajemniškem stanovanju itd.

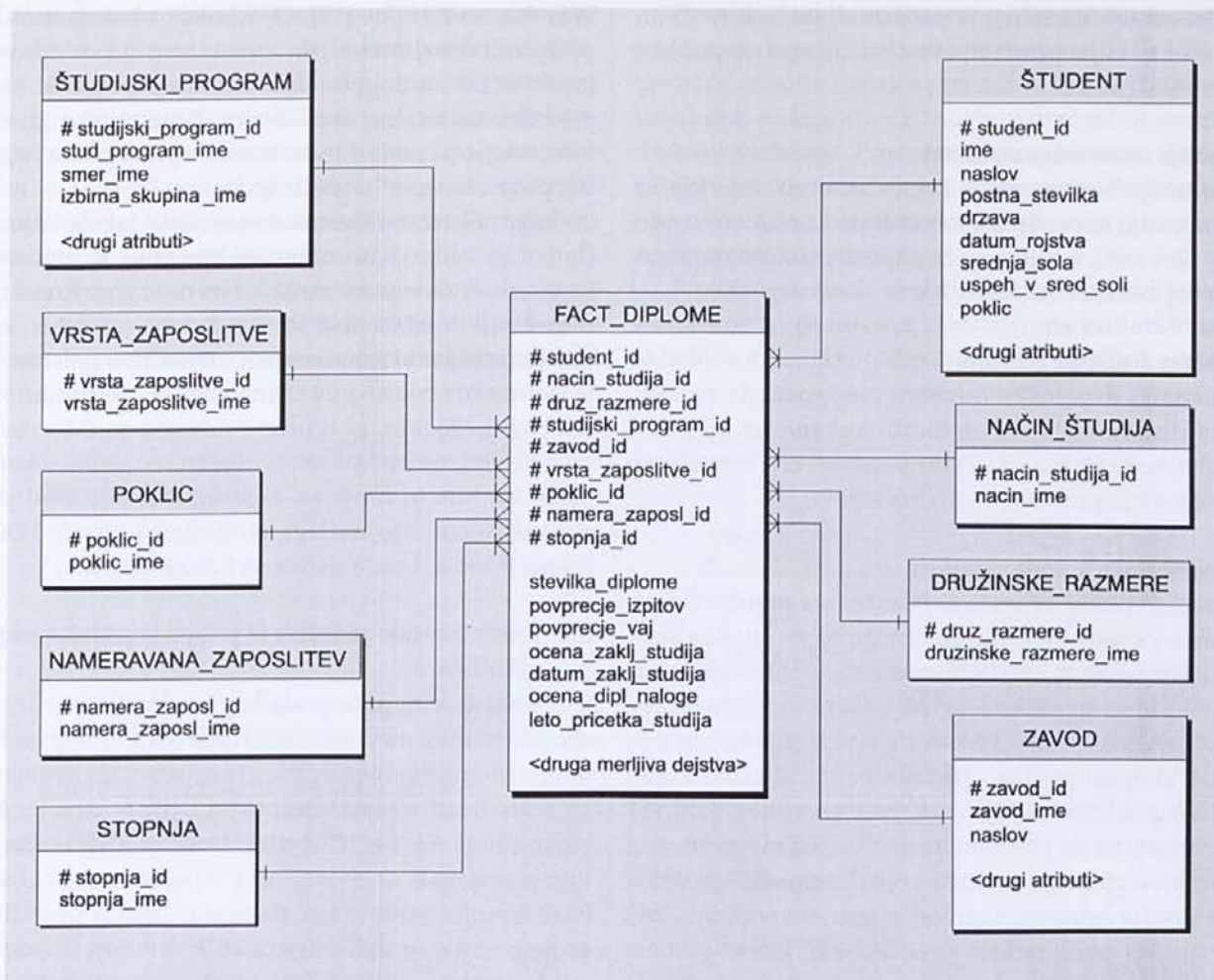
4 Področno podatkovno skladišče za analizo podatkov o diplomantih

Logični načrt področnega podatkovnega skladišča za analizo podatkov o diplomantih je prikazan na sliki 3. Sestavljajo ga tabela dejstev in 9 dimenzijskih tabel. Zrno tabele dejstev predstavlja diploma enega študenta, ki je prikazana kot dogodek na presečišču devetih dimenzij. Tabela dejstev vsebuje naslednja merljiva dejstva: številko diplome, datum zagovora, oceno

diplomske naloge, oceno zagovora, povprečno oceno izpitov in vaj, leto pričetka študija itd. Dimenzijske tabele pa predstavljajo dimenzije študent, zavod, študijski program, način študija, stopnja študija, poklic, vrsta zaposlitve, nameravana zaposlitev in družinske razmere.

S pomočjo tega področnega podatkovnega skladišča lahko dobimo odgovore na različna vprašanja, povezana s podatki o diplomantih, npr.:

- Koliko študentov je diplomiralo na posameznih članicah univerze?
- Kolikšno je število diplomantov na posameznih študijskih programih, smereh in izbirnih skupinah?
- Kolikšen je povprečen čas študija pri posameznih generacijah študentov (na posameznih članicah oziroma študijskih programih)?



Slika 3: Zvezdna shema področnega podatkovnega skladišča za analizo podatkov o diplomantih

- Kako uspešni so izredni študenti v primerjavi z rednimi?
- Kakšna je povprečna ocena študentov na različnih članicah, študijskih programih, smereh in izbirnih skupinah?
- Kje se nameravajo diplomanti zaposliti?
- Kakšne so družinske razmere diplomantov?

Dimenzijske tabele študent, zavod, študijski program, način študija in stopnja študija so skupne za obe področni podatkovni skladišči, zato so fizično realizirane samo enkrat in delujejo kot "lepilo", s katerim sta obe področni podatkovni skladišči povezani med seboj. Z njihovo pomočjo lahko izvajamo t. i. vrtnanje počez (angl. drill across) in tako generiramo poročila, ki vključujejo podatke iz obeh zvezdnih shem.

Poleg že omenjenih skupnih dimenzij nastopajo v področnem podatkovnem skladišču s podatki o diplomantih še štiri nove dimenzije: nameravana zaposlitev, družinske razmere, poklic in vrsta zaposlitve. Vsaka od teh dimenzij je predstavljena z ustreznim šifrantom, ki ga uporablja Statistični urad Republike Skovenije (SURS).

Dimenzija nameravana zaposlitev

S pomočjo te dimenzije lahko analiziramo, kje se nameravajo zaposliti diplomanti po končanem študiju: v Sloveniji, v tujini, so že zaposleni, se ne nameravajo zaposliti, še ne vedo, kje se bodo zaposlili.

Dimenzija družinske razmere

Dimenzija družinske razmere omogoča, da za vsakega diplomanta zabeležimo, kakšne so njegove družinske razmere v času diplome: ali živi sam ali pri starših, ali je poročen in ali ima otroke.

Dimenziji poklic in vrsta zaposlitve

Dimenziji poklic in vrsta zaposlitve sta pomembni za analizo podatkov o diplomantih, ki so študirali ob delu. Omogočata, da diplomante razvrstimo glede na delo, ki so ga opravljali med študijem in glede na vrsto zaposlitve. Vsaka vrstica v dimenziji poklic ustreza eni skupini poklicev, skladno s standardno klasifikacijo poklicev SURS. Vsaka vrstica v dimenziji vrsta zaposlitve pa predstavlja eno izmed mogočih vrst zaposlitve glede na to, kje so bili diplomanti zaposleni: v družbi oziroma podjetju, v lastnem podjetju, kot samostojni podjetniki – posamezniki, v svobodnem poklicu itd.

5 Opis orodij za realizacijo

V splošnem velja, da za realizacijo podatkovnega skladišča potrebujemo sistem za upravljanje podatkovne baze, orodja za izločanje, preoblikovanje in nalaganje podatkov (angl. ETL tools) in orodja za predstavitev podatkov končnim uporabnikom (angl. data access tools) [9].

5.1 Sistem za upravljanje podatkovne baze in fizični načrt

Pri izboru sistema za upravljanje podatkovne baze se navadno odločimo za tisti sistem, ki je predpisan kot interni standard organizacije, za katero razvijamo podatkovno skladišče. Vsako odstopanje namreč zahteva tehtno utemeljitev. Zato smo izbrali Oracle DBMS, ki je predlagan kot interni standard za razvoj vseh novih aplikacij na ljubljanski univerzi.

Fizični načrt podatkovnega skladišča in polnjenje skladišča s podatki smo realizirali z orodjem Oracle Warehouse Builder [12]. Oracle Warehouse Builder podpira tako dimenzijski kot relacijski podatkovni model in tako omogoča, da kreiramo vse objekte, ki so potrebni za fizično realizacijo: dimenzijske tabele, tabele dejstev, podatkovne kocke, uporabniške poglede, povezave med podatki ipd.

Najprej je treba kreirati dimenzijske tabele. V vsaki dimenziji lahko definiramo več hierarhij, ki predstavljajo temelj za vrtnanje navzdol in navzgor. Kreiranju dimenzijskih tabel sledi kreiranje tabele dejstev, kjer določimo ključne vse dimenzij in merljiva dejstva. Ko je definirana celotna zvezdna shema, definiramo še relacijske objekte, ki tvorijo področje za čiščenje in preoblikovanje podatkov. Ti objekti so tabele, sekvence, procedure in funkcije, ki se uporabljajo med preoblikovanjem. Na podlagi teh definicij Oracle Warehouse Builder kreira ustrezne fizične objekte.

5.2 Preoblikovanje podatkov in polnjenje podatkovnega skladišča

Viri podatkov za naše podatkovno skladišče so transakcijski sistemi na posameznih članicah univerze. Ker se informacijski sistemi članic med seboj razlikujejo in so realizirani z različnimi orodji (npr. Oracle, Microsoft SQL Server, Clipper), smo se odločili, da ne bomo posegali neposredno v njihove podatkovne baze, ampak zahtevamo, da vsaka članica posreduje svoje podatke v obliki tekstovnih datotek v vnaprej predpisanem formatu. Te datoteke zbiramo in z njimi

polnimo podatkovno skladišče v vnaprej predpisanih časovnih intervalih. Za področno podatkovno skladišče z vpisnimi podatki so ti intervali bolj gosti v času vpisa (tj. septembra in v začetku oktobra), medtem ko se med študijskim letom vpisni podatki praktično ne spreminjajo. Področno podatkovno skladišče s podatki o diplomantih pa osvežujemo vsak mesec.

Odločitvi za uporabo tekstovnih datotek sta botrovala še dva razloga: tekstovne datoteke so preproste in se jih da enostavno kreirati, Oracle pa ima na voljo učinkovita orodja za njihovo nalaganje v relacijske tabele.

Med postopkom preoblikovanja podatkov se tekstovne datoteke s podatki posameznih članic najprej naložijo v relacijske tabele s pomočjo orodja Oracle SQL*Loader. Pri tem SQL Loader preveri format podatkov in prisotnost vseh obveznih atributov. Nepopolne vrstice in vrstice z neustreznim formatom se izločijo in zabeležijo v posebnem dnevniku obdelave, ki služi kot osnova za odpravo napak.

V naslednjem koraku preverimo veljavnost podatkov (ali so njihove vrednosti znotraj predpisane domene in ustrezajo dogovorjenim poslovnim pravilom) in referenčno integriteto. Nazadnje opravimo še kontrolo podvajanja in izločimo duplikate. Tudi v tem koraku se napačni podatki skupaj z ustreznimi sporočili o napakah zapišejo na posebno datoteko, ki se potem uporabi pri njihovem čiščenju.

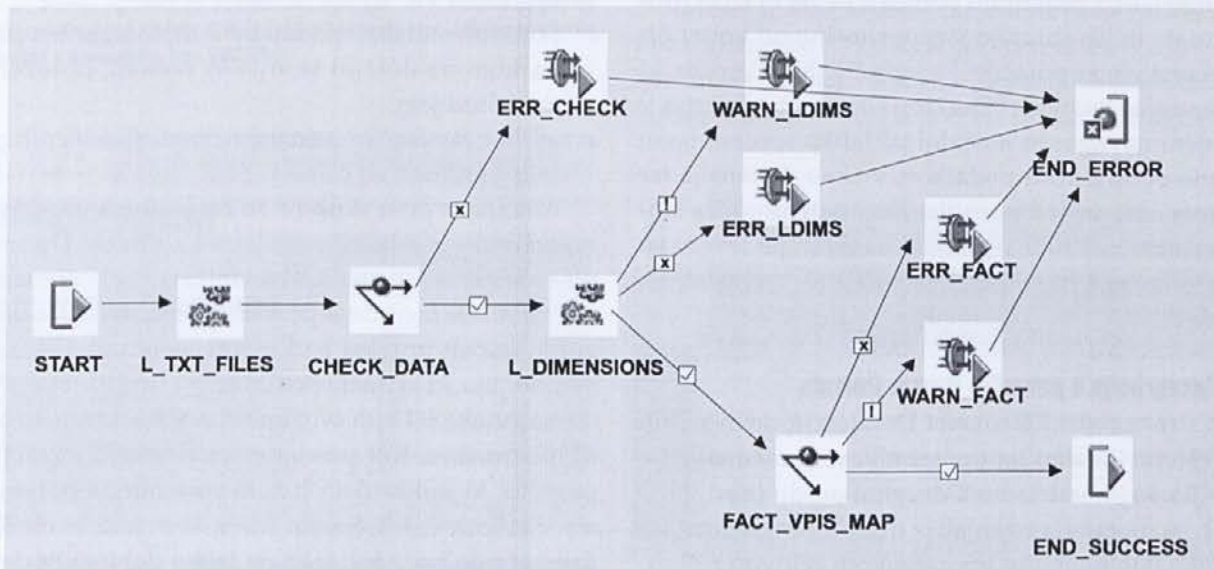
Podatki, ki so uspešno prestali vse kontrole, se na koncu naložijo v podatkovno skladišče. Med nalaganjem se najprej osvežijo dimenzijske tabele, nato pa še

tabela dejstev. V primeru sprememb v dimenzijskih tabelah nove vrednosti "povezijo" stare, kar ustreza tipu 1 počasi spreminjajočih se dimenzij [8]. Do takih sprememb največkrat pride v dimenziji študent, medtem ko so ostale dimenzije relativno majhne in se le redko spreminjajo.

Celoten postopek preoblikovanja in nalaganja podatkov v področno podatkovno skladišče s podatki o vpisu je prikazan na sliki 4. Proces `l_txt_files` sproži nalaganje tekstovnih datotek v relacijske tabele. Preslikava `check_data` preveri veljavnost podatkov, referenčno integriteto in morebitna podvajanja. V primeru napake se sproži procedura `err_check`, ki v poseben dnevnik obdelave zapiše vrsto napake in podatke, pri katerih je bila napaka ugotovljena. Proces `l_dimensions` sproži osveževanje dimenzij. Sestavljen je iz več podprocesov (angl. child processes) in preslikav (angl. mappings), ki osvežujejo posamezne dimenzije. Preslikava `fact_vpis_map` na koncu osveži tabelo dejstev.

Proceduri `err_ldims` in `warn_ldims` služita za obravnavo napak pri osveževanju dimenzijskih tabel. V primeru usodne napake procedura `err_ldims` prekine postopek polnjenja in zapiše ustrezno obvestilo v dnevnik obdelave. Procedura `warn_ldims` pa samo zapiše opozorilo in nadaljuje s polnjenjem. Podobno nalogo opravljata proceduri `err_fact` in `warn_fact` pri polnjenju tabele dejstev.

Na enak način je realizirano tudi preoblikovanje podatkov in polnjenje področnega podatkovnega skladišča s podatki o diplomantih.



Slika 4: Postopek preoblikovanja in nalaganja podatkov

5.3 Dostop do podatkov

Podatkovno skladišče samo po sebi ni koristno, če ne moremo na primeren način prikazati podatkov, ki so shranjeni v njem. V našem primeru so poročila, ki jih potrebujejo končni uporabniki, realizirana na tri načine:

- kot vnaprej definirane uporabniške aplikacije, napisane v jeziku PL/SQL;
- kot vnaprej definirana poročila, zgrajena z orodjem Oracle Discoverer in shranjena v repozitoriju;
- kot ad hoc poizvedbe, ki jih po potrebi generiramo z Oracle Discovererjem.

Prvo možnost uporabljamo za zahtevna poizvedovanja, ki jih je težko realizirati z običajnimi orodji za povpraševanje in generiranje poročil. Druga možnost je primerna za večino standardnih poročil, medtem ko se tretje možnosti poslužujemo v primeru, ko se zahteve za neko poročilo pojavijo spontano.

Orodje za dostop do podatkov Oracle Discoverer [13] sestavlja dve komponenti: Discoverer Administrator in Discoverer Desktop. Discoverer Administrator je namenjen računalniškim strokovnjakom in se uporablja za vrsto nalog, kot so npr. vzdrževanje meta podatkov (tj. določanje dimenzij in kock), nadzor nad dostopom in vzdrževanje agregatov. Vsi metapodatki so shranjeni v repozitoriju, imenovanem Discoverer End User Layer, ki predstavlja jedro sistema za dostop do podatkov v podatkovnem skladišču.

Končni uporabniki in analitiki uporabljajo Discoverer Desktop. Le-ta dostopa do metapodatkov repozitoriju, ki jih je pripravil Discoverer Administrator. Analitiki lahko generirajo poročila, ne da bi znali programirati, in jih shranijo v repozitoriju, od koder jih lahko ponovno uporabijo, kadarkoli želijo. Poročila, ki jih generira Discoverer Desktop so parametrizirana, z vrtnjem navzgor in navzdol pa lahko spreminjamo stopnjo podrobnosti podatkov, vrtimo dimenzije ter spreminjamo izgled poročila. Rezultate poročila lahko predstavimo tudi v grafični obliki in jih izvozimo v druga orodja (npr. Excel, glej sliko 9) za nadaljnjo obdelavo.

5.4 Integracija s pomočjo Oracle Portala

Slaba stran orodja Discoverer Desktop je, da mora biti nameščeno lokalno na uporabnikovem računalniku, poročila, ki jih izdelamo z drugimi orodji (npr. v PL/SQL), pa moramo poganjati iz drugih programov. Da bi lahko dali končnim uporabnikom celovito rešitev, ki bi omogočala dostop do podatkovnega skladišča

prek ene same vstopne točke, smo se odločili za integracijo v okolju Oracle 9iAS Portal [4].

V okviru portala smo izdelali aplikacijo, ki omogoča dostop do podatkovnega skladišča prek svetovnega spleta in povezuje v enovito rešitev vsa poročila ne glede na to, s katerim orodjem so bila izdelana. Vsa poročila (vnaprej sprogramirana v PL/SQL in avtomatično generirana z orodjem Discoverer) so dostopna prek enotnega uporabniškega vmesnika, tako da uporabniku ni treba vedeti, kakšna je njihova dejanska realizacija.

Oracle 9iAS Discoverer uporablja iste metapodatke kot Discoverer Desktop, zato lahko do poročil, ki so shranjena v repozitoriju, dostopamo prek svetovnega spleta, ne da bi jih spreminjali. Analitiki in programerji lahko pripravljajo poročila in jih shranjujejo v repozitoriju s pomočjo ustrezne okenske aplikacije, medtem ko jih končni uporabniki lahko uporabljajo prek svetovnega spleta. Zahtevna poročila pa lahko realiziramo kot dinamične strani v PL/SQL.

6 Pregled analiz in tipičnih poročil za končne uporabnike

V tem delu podajamo pregled najpomembnejših analiz, ki jih omogoča naše podatkovno skladišče. Zaradi večje preglednosti smo analize grupirali v 4 skupine:

- analize, ki omogočajo pregled nad številom vpisanih študentov in številom diplomantov na posameznih članicah,
- podrobne analize vpisnih podatkov s poudarkom na analizi prehodnosti iz letnika v letnik in trendih vpisa,
- podrobne analize podatkov o diplomantih s poudarkom na dolžini študija in ocenah, doseženih med študijem,
- ad hoc analize, ki nastajajo zaradi specifičnih potreb posameznih članic.

Analize iz prve skupine so razmeroma enostavne in se jih da učinkovito realizirati z Oracle Discovererjem, vendar so za vodstvo univerze zelo pomembne, saj je šele izgradnja podatkovnega skladišča omogočila celovit pregled nad vsemi članicami. Glede na avtonomijo, ki jo imajo posamezne članice, in dejstvo, da ima vsaka od njih svoj informacijski sistem, to prej ni bilo možno. Kot primer te vrste analiz navajamo poročila, ki prikazujejo število vpisanih na posameznih članicah v določenem študijskem letu in omogočajo vrtnje navzdol, tako da lahko dobimo podatke o vpisu na različnih stopnjah podrobnosti:

- za članico kot celoto,
- po posameznih študijskih programih, ki jih izvaja določena članica,
- po smereh znotraj študijskega programa,
- po izbirnih skupinah znotraj vsake smeri,
- po letnikih in študijskih programih.

Primer vrtnja navzdol prikazuje slike 5, 6 in 7. Medtem ko so na sliki 5 prikazani samo sumarni podatki o številu vpisanih na posameznih članicah, je iz slike 6 razvidno število vpisanih po posameznih študijskih programih, iz slike 7 pa tudi število vpisanih na posameznih smereh študija.

Število vpisanih študentov po članicah			
Page Item: Študijsko leto: 2003 ▼			
	Število vpisanih		
	redni študij	izredni študij	Skupaj
Akademija za glasbo Ljubljana	402	15	417
Akademija za gled., radio, film in TV	87		87
Akademija za likovno umetnost Lj.	395	16	411
Biotehniška fakulteta	3.155	264	3.419
Ekonomski fakulteta Ljubljana	5.693	3.167	8.860
Fakul. za gradbeništvo in geodezijo	1.558	249	1.807
Fakulteta za arhitekturo	847	52	899
Fakulteta za družbene vede Lj.	3.651	784	4.435
Fakulteta za elektrotehniko	2.125	59	2.184
Fakulteta za farmacijo	1.040	32	1.072
Fakulteta za kemijo in kem. tehn.	1.369	132	1.501
Fakulteta za matematiko in fiziko	963	57	1.020
Fakulteta za pomorstvo in promet	1.277	784	2.061
Fakulteta za računalništvo in informatiko	1.404	49	1.453
Fakulteta za socialno delo	607	528	1.135
Fakulteta za strojništvo Ljubljana	1.754	73	1.827
Fakulteta za šport Ljubljana	945	123	1.068
Fakulteta za upravo	1.267	2.113	3.380

Slika 5: Število vpisanih po posameznih članicah. Prvi stolpec prikazuje število rednih, drugi stolpec število izrednih in tretji stolpec skupno število študentov v študijskem letu 2003/04.

Število vpisanih po članicah in študijskih programih			
Page Item: Študijsko leto: 2003 ▼			
	Št. vpisanih		
	redni študij	izredni študij	Skupaj
Fakulteta za elektrotehniko	2.125	59	2.184
▶ ELEKTROTEHNIKA UN	1.102		1.102
▶ ELEKTROTEHNIKA VS	1.023	59	1.082
Fakulteta za računalništvo in informatiko	1.404	49	1.453
▶ RAČUNAL. IN INFORMATIKA UN	808	24	832
▶ RAČUNAL. IN INFORMATIKA VS	596	25	621
Medicinska fakulteta Ljubljana	1.685		1.685
▶ MEDICINA - VIS	1.321		1.321
▶ STOMATOLOGIJA - VIS	364		364
Skupaj vsi	5.214	108	5.322

Slika 6: Primer vrtnja navzdol: število vpisanih v posamezne študijske programe na izbranih članicah

Število vpisanih po članicah in študijskih programih			
Page Item: Študijsko leto: 2003 ▼			
	Št. vpisanih		
	redni študij	izredni študij	Skupaj
Fakulteta za elektrotehniko	2.125	59	2.184
▶ ELEKTROTEHNIKA UN	1.102		1.102
▶ Avtomatika	194		194
▶ Elektronika	124		124
▶ Močnostna elektrotehnika	106		106
▶ NI smeri	536		536
▶ Telekomunikacije	142		142
▶ ELEKTROTEHNIKA VS	1.023	59	1.082
▶ Avtomatika	150	10	160
▶ Elektronika	133	26	159
▶ Energetska teh. in avt. post.	145	11	156
▶ NI smeri	394		394
▶ Telekomunikacije	137	12	149
▶ Zagotavljanje kakovosti	64		64
Fakulteta za računalništvo in informatiko	1.404	49	1.453
▶ RAČUNAL. IN INFORMATIKA UN	808	24	832

Slika 7: Nadaljevanje vrtnja navzdol: število vpisanih po posameznih smereh študija

V to skupino sodijo tudi posebne analize števila vpisanih na Filozofsko fakulteto, za katero je značilen dvopredmetni študij, ki omogoča, da študenti izberejo poljubno kombinacijo dveh študijskih programov. Te analize omogočajo pregled nad vsemi izbranimi kombinacijami in vrtnje navzdol po obeh študijskih programih.

Analize iz druge skupine skušajo dati odgovore na dve vrsti vprašanj:

- Kako določena generacija študentov, ki so vpisani v neki študijski program, napreduje iz letnika v letnik?
- Kako se spreminjajo trendi vpisa oziroma zanimanje študentov za študij na posameznih fakultetah?

Primer analize, ki daje odgovor na prvo vprašanje, je prikazan na sliki 8. Iz slike je jasno razviden tipičen problem univerzitetnega študija v Sloveniji, to je izredno velik osip študentov na prehodu iz prvega v drugi letnik.

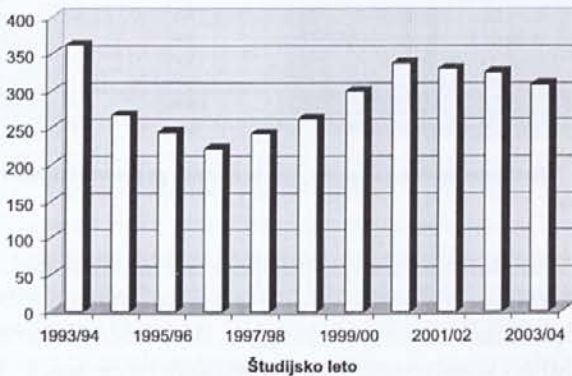
Spremljanje napredovanja generacije						
Page Item: Zvezi: Fakulteta za računalništvo in informatiko ▼ VIP: RAČUNAL. IN INFORMATIKA III ▼ Leto						
	Število vpisanih					
	1. letnik	2. letnik	3. letnik	4. letnik	5. letnik	Absolvent UNI
1995/96	163					163
1996/97	25	73				98
1997/98	6	34	55			95
1998/99		3	24	51		78
1999/00			1	2	22	51
2000/01				1	3	22
2001/02				1		2
2002/03						2
2003/04					2	2
Skupaj	194	111	83	78	75	605

Slika 8: Napredovanje izbrane generacije študentov

Primer analize, ki daje odgovor na drugo vprašanje, pa je na sliki 9. Slika jasno kaže na upad zanimanja za študij tehnike po osamosvojitvi Slovenije in spremembi družbenega sistema. Šele v zadnjih letih se je situacija nekoliko popravila.

V tretjo skupino spadajo analize, ki prikazujejo podatke o dolžini študija in uspešnosti posameznih generacij diplomantov. Primer take analize je prikazan na sliki 10.

Število vpisanih na študijski program ELEKTROTEHNIKA



Slika 9: Število vpisanih na univerzitetni študij elektrotehnike v letih 1993–2003

Univerza v Ljubljani

Čas trajanja študija

Page Item: Leto diplomanja: 2003 Vrsta študija: visokoš. (univ.) št.

Zavod	Študijski program	Trajanje študija
Akademija za glasbo i ljubljana	CEFKOVNA GLASBA - VIS	6,75
	GLASBENA PEDAGOGIKA -VIS	7,50
	GOGALA IN DR INSTR. S STR -VIS	6,65
	INSTRUMENTI S TIFKAMI - VIS	6,46
	KOMPOZICJA IN GLASB.TEOR.-VIS	6,06
	PETJE - VIS	4,25
Akademija za gled.,radio,film in TV	PIHALA, TROBILA IN TOLKALA-VIS	7,15
	DRAMATURGIJA	6,41
	DRAMSKA IGRA IN UMETNJSKA BEG	7,32
	FILMSKA IN TELEVIZJSKA REŽIJA	7,75
	GLEDALJSKA IN RADJSKA REŽIJA	7,75
		Povprečje: 7,31

Slika 10: Trajanje študija v letih za posamezne študijske programe na izbranih članicah

Zadnjo skupino tvorijo analize, ki nastajajo spontano kot rezultat zahtev, ki se pojavljajo med vsakodnevnim poslovanjem. Te analize skušamo realizirati takoj, ko se pojavi tovrstna potreba, z uporabo Oracle Discovererja. Primer take analize prikazuje slika 11. Analiza je nastala kot posledica zahteve Fakultete za računalništvo in informatiko, ki je želela podrobneje analizirati strukturo vpisanih v prvem letniku visokošolskega strokovnega študija glede na poklic, ki ga

Univerza v Ljubljani

Število vpisanih študentov glede na srednješolski poklic

Page Item: Zavod: Fakulteta za računalništvo in informatiko Študijsko leto: 2003/04 VIP: RACUNA

	Št. vpisanih		Skupaj
	redni študij	izredni študij	
5.70.06 EKONOMSKO-KOMERC.TEHNİKAD 1991	35	3	38
5.26.01 RAČUNALNIŠKI TEHNİK	22	3	25
5.25.38 ELEKTROTEHNİK ELEKTRONİK	20	4	24
5.99.11 GIMNAZJSKI MATURANT	20	7	27
5.11.01 LEŠARSKI TEHNİK	8		8
5.73.15 POSLOVNI TEHNİK (3+2)	7		7
5.20.01 STROJNI TEHNİK	6	2	8
5.70.08 EKONOMSKI TEHNİK - PR	5		5
5.94.01 ZDRAVSTVENI TEHNİK	5		5
5.99.31 TEHNIŠKA GIMNAZIJA	4	1	5
5.45.01 GRAFIČNI TEHNİK	3		3

Slika 11: Struktura vpisanih v 1. letnik visokošolskega strokovnega študija računalništva glede na poklic po končani srednji šoli

imajo študenti po končani srednji šoli. Analiza je dala zanimive rezultate: največ študentov prihaja iz ekonomske srednje šole, šele nato sledijo gimnazijski maturanti in računalniški tehniki. Ta in podobne analize dajejo vodstvu fakultete možnost, da sprejme ustrezne odločitve: lahko se odloči za prilagoditev vsebine nekaterih tehničnih predmetov (npr. osnove programiranja in osnove arhitekture računalniških sistemov) predznanju študentov, ki so končali program ekonomsko-komercialni tehnik, ali pa predlaga spremembo pogojev za vpis v ta študijski program tako, da bi se zmanjšal vpis študentov, ki nimajo predhodne tehnične izobrazbe.

7 Sklep

Zaradi decentralizirane organizacije je izgradnja podatkovnega skladišča za Univerzo v Ljubljani še posebej pomembna, saj predstavlja podatkovno skladišče sredstvo za integracijo podatkov, ki sicer nastajajo in se obdelujejo v okviru informacijskih sistemov posameznih članic. Na ta način je vodstvu univerze omogočen vpogled v poslovanje univerze kot celote, ki pred izgradnjo podatkovnega skladišča ni bil možen.

Zaradi heterogenih virov podatkov (informacijski sistemi posameznih članic se med seboj razlikujejo) pa je izgradnja podatkovnega skladišča zahtevna in tvegana naloga. Izkušnje namreč kažejo, da napor za pripravo podatkov narašča eksponentno s številom različnih izvornih sistemov [9]. Da bi zmanjšali tveganje in čim prej prišli do otipljivih rezultatov, smo izbrali pristop "od spodaj navzgor", ki predvideva postopno izgradnjo posameznih področnih podatkovnih skladišč in njihovo povezovanje prek skupnih

dimenzijskih tabel. Najprej smo realizirali področno podatkovno skladišče z vpisnimi podatki, saj za to področje obstaja enoten in razmeroma stabilen vir podatkov – obrazec Vpisni list. Temu smo nato dodali še področno podatkovno skladišče s podatki o diplomantih. V članku je podrobno predstavljen logični podatkovni model za obe področni podatkovni skladišči in opisan postopek njune realizacije s pomočjo orodij iz družine Oracle.

Končni cilj projekta pa je izgradnja verige področnih podatkovnih skladišč, s pomočjo katerih bo moč spremljati študijski proces kot vrednostno verigo, ki se prične s prijavo za vpis na univerzo, nadaljuje z vpisom in opravljanjem izpitov v posameznih letnikih ter konča z diplomom.

Literatura

- [1] Desnos, J.-F. A Data Warehouse for French Universities. V: J. Knop, P. Schirmbacher in V. Mahnič (ur.). *The Changing Universities – The Role of Technology*, Gesellschaft für Informatik, Bonn, 2002, str. 146–152.
- [2] Eckerson, W. W. *Evolution of Data Warehousing, The Trend Toward Analytical Applications*, Boston, MA: The Patricia Seybold Group (April 28, 1999), str. 1–8.
- [3] Cardoso, E., Galhardas, E., Trigueiros, M. J., Silva, R. A. *Decision Support System for IST Academic Information*. V: *Beyond the Network – Innovative IT Services*, Proceedings of the 9th International Conference of European University Information Systems, Amsterdam, julij 2003, str. 416–427.
- [4] Greenwald R., Milbrey J., *Oracle 9iAS Portal Bible*, John Wiley & Sons, 2001.
- [5] Flory, A., Soupirot, P., Tchounikine, A. *Design and Implementation of a Data Warehouse for Research Administration Universities*. V: J. Knop, P. Schirmbacher in V. Mahnič (ur.). *The Changing Universities – The Role of Technology*, Gesellschaft für Informatik, Bonn, 2002, str. 259–267.
- [6] Inmon, B. *Building the Data Warehouse*, QED Publishing Group, 1992.
- [7] Inmon, B. *Building the Data Warehouse, Third Edition*, John Wiley & Sons, 2002.
- [8] Kimball, R. et al. *The Data Warehouse Toolkit*, John Wiley & Sons, 1996.
- [9] Kimball, R. et al. *The Data Warehouse Lifecycle Toolkit*, John Wiley & Sons, 1998.
- [10] Mahnič, V. *Data warehouse: a tool for analysing study process as a value chain*. V: *Beyond the Network – Innovative IT Services*, Proceedings of the 9th International Conference of European University Information Systems, Amsterdam, julij 2003, str. 124–130.
- [11] Oliveira, C. C., Domingues, M. *Multidimensional Analysis of Administrative Data in eLearning Systems*. V: V. Mahnič in B. Vilfan (ur.) *IT Innovation in a Changing World*, Proceedings of the 10th International Conference of European University Information Systems, Bled, Slovenija, 29. junij–2. julij 2004, str. 172–178.
- [12] Oracle, *Oracle9i Warehouse Builder User's Guide, Release 2 (9.0.4)*, 2003.
- [13] Oracle, *Oracle9i Discoverer Plus User's Guide, Version 9.0.2*, 2002.
- [14] Stevenson, D. *Data Warehouse and Executive Information Systems – Ignoring the Hype*. V: J.-F. Desnos in Y. Epelboin (ur.). *European Cooperation in Higher Education Information Systems*, Grenoble, Francija, September 1997, str. 202–207.
- [15] Watson, H. J., Wixom, B. H., Buonamici, J. D., Revak, J. R. *Sherwin-Williams' Data Mart Strategy: Creating Intelligence across the Supply Chain*, Communications of the Association for Information Systems, Vol. 5, Article 9, May 2001.
- [16] Watson, H. J. *Recent Developments in Data Warehousing*, Communications of the Association for Information Systems, Vol. 8, 2001, pp. 1–25.
- [17] Watson, H. J., Goodhue, D. L., Wixom, B. H. *The Benefits of Data Warehousing: Why Some Organizations Realize Exceptional Payoffs*, Information & Management, Vol. 39, 2002, pp. 491–502.

Marko Požnel je diplomiral leta 2001 in magistriral leta 2004 na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer je tudi zaposlen kot asistent. Ukvarja se s podatkovnimi skladišči in razvojem spletnih informacijskih sistemov. Poleg pedagoškega in raziskovalnega dela aktivno sodeluje tudi pri prenovi informacijskega sistema za področje študijske informatike na ljubljanski univerzi.

Viljan Mahnič je izredni profesor in predstojnik Laboratorija za tehnologijo programske opreme na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. V letih 1999–2003 je bil tudi prodekan za pedagoško delo. Ukvarja se z razvojem programske opreme za računalniško podprte informacijske sisteme s posebnim poudarkom na informacijskih sistemih za področje visokega šolstva. Od leta 1996 je predstavnik Slovenije v EUNIS (European University Information Systems Association), od leta 2002 pa tudi član sveta direktorjev omenjene organizacije.

Obvladovanje tveganj pri uporabi informacijske tehnologije v univerzitetnem okolju

Nataša Žabkar, Viljan Mahnič
Univerza v Ljubljani, Fakulteta za računalništvo in informatiko
natasa.zabkar@email.si, viljan.mahnic@fri.uni-lj.si

Povzetek

Vse večja uporaba informacijskih tehnologij (IT) v univerzitetnem okolju zahteva, da zagotovimo ustrezno stopnjo kakovosti, varnosti in zaupanja v rezultate pri uporabi IT. Vse to lahko dosežemo z upravljanjem IT. Pomemben del le-tega pa je obvladovanje tveganj, ki izvirajo iz uporabe IT. Namen tega prispevka je predstaviti pristop k obvladovanju IT tveganj, ki temelji na ogrodju za upravljanje informacijske tehnologije COBIT in je nadgrajen z uporabo "Standarda za obvladovanje tveganj" in standarda BS 7799. Najprej podamo pregled navedenih treh pristopov in potem opišemo naš pristop. Na koncu podamo še zgled, kako bi lahko uporabili predlagani pristop na primeru informacijskega sistema univerze za področje študijske informatike.

Ključne besede: COBIT, "Standard za obvladovanje tveganj", BS 7799.

Abstract

IT Risk Management in the University Environment

The increasing use of information technology (IT) in the university environment raises the importance of achieving quality and security, as well as satisfying fiduciary requirements of IT usage. These goals can be achieved through IT governance, an important part of which is IT risk management. The aim of this paper is to present an approach to IT risk management that is based on the IT governance framework COBIT and refined using of the standards "A Risk Management Standard" and BS 7799. We first give an overview of these three approaches and then describe our approach. Finally, we give an example of its usage in the Student Records Information System.

Keywords: COBIT, "A Risk Management Standard", BS 7799.

1 Uvod

Univerzitetni informacijski sistem je zelo pomemben za doseganje ciljev univerze, zato predstavlja pomembno sredstvo, ki ga je treba ustrezno varovati. Upravljanje informacijske tehnologije (angl. information technology governance) in obvladovanje tveganj informacijske tehnologije (angl. IT risk management) nam pri tem lahko pomagata. Za upravljanje IT je odgovorno poslovodstvo. Upravljanje IT zajema vodenje, organizacijsko strukturo in procese, ki zagotavljajo, da IT univerze ohranja in razširja strategijo in cilje univerze [16].

Dva glavna cilja upravljanja IT sta:

- zagotoviti, da bodo pričakovanja glede uporabe IT izpolnjena;
- zagotoviti, da bodo tveganja, povezana z uporabo IT ustrezno obvladovana.

V tem prispevku se bomo osredotočili na drugi cilj.

Obvladovanje tveganj IT dodaja in ohranja vrednost organizacije [01]. Ohranjanje ali varovanje vred-

nosti sredstev zajema tudi kontrole, s katerimi znižujemo tveganja na stopnjo, ki jo je opredelilo poslovodstvo v procesu obvladovanja tveganj. Eden izmed pogojev za ustrezno upravljanje IT je sistematični pristop k obvladovanju tveganj. Poznamo več pristopov, ponavadi pa se poslovodstvo odloči za tisto kombinacijo obstoječih pristopov, ki je po njegovi presoji za organizacijo najbolj ustrezna.

V finančnih organizacijah je obvladovanje tveganj določeno z zakonom, za razliko od univerzitetnega okolja, kjer je odločitev glede obvladovanja tveganj prepuščena poslovodstvu. Slovenska zakonodaja med drugim¹ določa, da morajo banke v poslovanju upoštevati slovenska standarda SIST BS 7799-2:2003 in SIST ISO/IEC 17799:2003, ki ju izdaja slovenski inštitut za standardizacijo oziroma drug pooblaščen organ (UL RS št. 83/2004, 29.07.2004). Zahteva po obvladovanju tveganja je opredeljena v standardu BS 7799-2:2002

[05] v poglavju A.4.2 "Vzpostavitev in upravljanje SUVI (sistema za upravljanje varovanja informacij)". Obvladovanje tveganj je velikega pomena v bankah, kjer kakovost obvladovanja tveganj vpliva na zahteve glede najnižje zakonske kapitalske obveznosti za kritje tveganja. Tveganja IT uvrščamo med operativna tveganja, ki so predmet Novega baselskega sporazuma (New Basel Capital Accord) oziroma Basel II [04]. Spremljevalne dokumentne odprtega tipa, kot so "Postopki dobre prakse za obvladovanje in nadziranje operativnega tveganja" [03], je mogoče uporabiti tudi v drugih okoljih, npr. prvo načelo: "Nadzorni svet bi moral pomembnejše vidike operativnih tveganj banke obravnavati kot posebno in obvladljivo skupino tveganj ter sprejeti in občasno preverjati strategijo banke v odnosu do operativnih tveganj. Strategija naj bi odražala odpornost banke do teh tveganj in njeno poznavanje in razumevanje posebnih značilnosti tovrstnih tveganj. Potrditi oz. sprejeti bi moral tudi osnovno obliko organiziranosti za obvladovanje operativnega tveganja in zagotoviti, da uprava dejansko izvaja svoje naloge v zvezi z obvladovanjem tega tveganja."

Namen tega prispevka je predstaviti metodo za obvladovanje tveganj IT, ki bi bila primerna za univerzitetno okolje. Čeprav za univerze uporaba prej omenjenih standardov ni predpisana z zakonom, menimo, da je uporaba le-teh lahko v pomoč pri uvajanju obvladovanja tveganj IT v univerzitetnem okolju.

V naslednjem razdelku bomo opredelili obvladovanje tveganj in predstavili nekatere možne rešitve za zmanjšanje tveganj IT. Potem bomo predstavili naš pristop k obvladovanju tveganj IT kot eno izmed mogočih rešitev (razdelek 3) in podali zgled uporabe tega pristopa (razdelek 4). Na koncu bomo podali sklepne ugotovitve.

2 Obvladovanje tveganj IT

Obstaja več definicij tveganja, npr. "možnost, da se zgodi nekaj, kar bo vplivalo na cilje" [02]. V tem prispevku bomo uporabili naslednjo definicijo: "kombinacija verjetnosti dogodka in njegovih posledic (ISO/IEC Guide 73)" [01], ker vključuje pozitiven in negativen vidik tveganja.

Obstaja tudi več definicij obvladovanja tveganj. Ena izmed njih je "kultura, procesi in strukture, ki so usmerjeni k uspešnemu obvladovanju možnih priložnosti in neželenih učinkov" [02]. V tem prispevku bomo uporabili naslednjo definicijo: "proces, v katerem se organizacija metodično ukvarja s tveganji za vsako aktivnost in sicer z namenom doseganja stalne koristi v okviru vsake posamezne aktivnosti ter za portfelj vseh aktivnosti" [01] in sicer zato, ker poudarja doseganje ciljev.

Obvladovanje tveganj v splošnem je opisano v dokumentu "Standard za obvladovanje tveganj" (angl. "A Risk Management Standard") [01], ki je odprt standard in so ga razvile večje organizacije za obvladovanje tveganj v Veliki Britaniji. Podobno velja za avstralsko-novozelandski standard AS/NZS 4360:1999 "Obvladovanje tveganj" [02], ki pa je zaprt standard. Oba standarda podajata splošno ogrodje za obvladovanje tveganj, ki ni odvisno od posamezne panoge. Pred kratkim je bil objavljen še en dokument, ki opisuje ogrodje za obvladovanje tveganj v podjetjih – COSO-ERM: "Enterprise Risk Management Framework" [09] in je prav tako splošne narave.

Do sedaj smo opredelili obvladovanje tveganj v splošnem. V tem prispevku se bomo ukvarjali z obvladovanjem tveganj IT, ki je del splošnega obvladovanja tveganj. Obvladovanje tveganj IT je usmerjeno predvsem na operativna tveganja, ki so v Basel II [03] opredeljena na naslednji način: "tveganje izgube zaradi neustreznih ali neuspešnih notranjih postopkov, ljudi in sistemov ali zaradi zunanjih dogodkov". Nekateri primeri operativnih tveganj so: prekinitev poslovanja in sistemske napake (napake v strojni in programski opremitvi, telekomunikacijski problemi, izpadi sistema itn.), napake pri izvedbi procesov in upravljanju postopkov (napačen vnos podatkov, vzdrževanje ali polnjenje podatkov v bazo, napačno delovanje sistema/modela, vzdrževanje referenčnih podatkov, napačni podatki o komitentu itn.), poškodbe fizičnih sredstev (potresi, požari, poplave itn.) in drugi.

Obstaja večje število dokumentov, ki opisujejo proces obvladovanja tveganj IT, nekateri pa se nanašajo predvsem na varnostna tveganja IT:

¹ Slovenska zakonodaja določa, da morajo prireditelji pri trajnem prirejanju klasičnih iger na srečo pri uporabi programske, računalniške, mrežne in telekomunikacijske opreme upoštevati določila slovenskega standarda s področja varovanja informacij (UL RS št. 70/2000, 8. 8. 2000); družbe za upravljanje morajo pri svojem poslovanju smiselno upoštevati slovenski standard PSIST BS 7799 Kodeks varovanja informacij (UL RS št. 80/2003, 14. 8. 2003); klirinška družba mora pri poslovanju upoštevati slovenski standard PSIST BS 7799 (UL RS št. 7/2003, 23. 01. 2003); pri družbah za izdajo elektronskega denarja mora revizor v dodatku revizorjevega poročila oceniti zlasti usklajenost s slovenskim standardom PSIST BS 7799 (UL RS št. 87/2002, 17. 10. 2002).

- COBIT: Kontrolni cilji za IT in sorodne tehnologije (angl. "Control Objectives for Information and Related Technology") [07];
- PD 30002:2002: Vodič za oceno tveganj po BS 7799 (angl. "Guide to BS 7799 Risk Assessment") [23];
- NIST 800-300: Vodič za obvladovanje tveganj za sisteme informacijske tehnologije – priporočila Nacionalnega inštituta za standarde in tehnologijo [21];
- ISO/IEC TR 13335-n: Informacijska tehnologija – Smernice za obvladovanje varnosti IT (angl. "Information Technology – Guidelines for the management of IT security" [13];
- HB 231:2000: Smernice za obvladovanje tveganj informacijske varnosti (angl. "Information Security Risk Management Guidelines") [10];
- CCTA Metoda za analizo in obvladovanje tveganj [25];
- Microsoft Operations Framework v3.0: Obvladovanje tveganj v operativi [20].

Podrobnejši pregled drugih dokumentov s področja obvladovanja IT je podan v [18], opis enega izmed pristopov ("Analiza varnostne ogroženosti") pa v [17].

V naslednjem razdelku bomo podali kratek opis treh standardov oziroma smernic, ki smo jih uporabili pri oblikovanju našega pristopa: COBIT [07], "Standard za obvladovanje tveganj" [01] in BS 7799 [05]. Želeli smo uporabiti samo odprt pristop, opisan v COBIT-u, vendar smo ugotovili, da potrebujemo bolj podroben opis procesa obvladovanja tveganj. Tega smo našli v odprtem pristopu, opisanem v "Standardu za obvladovanje tveganj". Vendar pa je ta standard splošne narave, mi pa smo potrebovali smernice za obvladovanje tveganj IT. Le-te smo našli v zaprtem pristopu BS 7799, ki je zelo razširjen in velja za najboljšo prakso na področju varnosti IT.

2.1 COBIT: PO9 "Oceniti tveganja"

COBIT [07] je izdal Inštitut za upravljanje IT in predstavlja zbirko dokumentov, ki so najboljša praksa za upravljanje IT, kontrolo in zagotavljanje. Večina komponent COBIT je odprtega tipa.

COBIT ima 34 procesov in za vsak proces so opredeljeni podrobni cilji. Eden izmed teh procesov je PO9: "Oceniti tveganja" (oznaka PO9 je okrajšava za "Planiranje in organizacijo", kar je ena od štirih domen, v katere so razvrščeni vsi procesi v COBIT-u). Proces PO9 ima na nižjem nivoju 8 kontrolnih ciljev [07]:

- PO9.1: Ocena poslovnih tveganj (vzpostavitev ogrožja za sistematično oceno tveganj);
- PO9.2: Pristop k oceni tveganj (opredelitev obsega, metodologije, odgovornosti in veščin);
- PO9.3: Opredelitev tveganj (poslovnih, regulativnih, pravnih, tehnoloških, povezanih s poslovnimi partnerji in s človeškimi viri);
- PO9.4: Merjenje tveganj (kvalitativno in kvantitativno rangiranje tveganj);
- PO9.5: Plan aktivnosti za obravnavanje tveganj (izogibanje, blažitev, prevzem ali prenos tveganja);
- PO9.6: Prevzem tveganja (formalen prevzem preostalega tveganja);
- PO9.7: Izbira kontrol (kontrolne, ki imajo največje koristi in najmanjše stroške – angl. "quick win");
- PO9.8: Zavezanost oceni tveganj (spodbujanje uporabe ocene tveganja).

Primerjava COBIT-a in standardov ISO/IEC 17799:2000 ter ISO/IEC TR 13335 (in drugih) je podana v [08].

Prednosti uporabe pristopa COBIT za obvladovanje tveganj IT sta:

- primernost za organizacije, ki že uporabljajo COBIT;
- primernost za brezplačen prenos znanja (odprt pristop).

Pomanjkljivosti uporabe pristopa COBIT pa sta:

- zahtevnost in pomanjkanje formalnega izobraževanja za organizacije, ki še ne uporabljajo COBIT-a;
- uporaba COBIT-a ni tako razširjena kot uporaba BS 7799.

2.2 "Standard za obvladovanje tveganj" (SOT)

Standard za obvladovanje tveganj (SOT) [01] je odprt standard, ki so ga razvile večje organizacije za obvladovanje tveganj v Veliki Britaniji: (1) Inštitut za obvladovanje tveganj (IRM), (2) Združenje menedžerjev za zavarovanja in tveganja (AIRMIC) in (3) ALARM Nacionalni forum za obvladovanje tveganj v javnem sektorju (ALARM). Standard za obvladovanje tveganj predstavlja najboljšo prakso, ki jo organizacije lahko uporabijo za primerjavo oziroma merjenje. V skladu s tem standardom je proces obvladovanja tveganj sestavljen iz naslednjih korakov:

- Korak 1: Strateški cilji organizacije;
- Korak 2: Ocena tveganja;
- Korak 3: Poročanje in komuniciranje o tveganjih (Grožnje in priložnosti);

- Korak 4: Odločitev;
- Korak 5: Obravnavanje tveganj (izbira in uvedba ukrepov za obravnavanje tveganj, stroški in koristi kontrol);
- Korak 6: Poročanje o preostalem tveganju (notranje, zunanje);
- Korak 7: Nadzor in pregled procesa obvladovanja tveganj (spremembe, zagotavljanje).
Drugi korak ima naslednje podkorake:
- Korak 2.1: Analiza tveganj;
- Korak 2.1.1: Opredelitev tveganj (strateška, operativna, finančna, obvladovanje znanja, skladnost);
- Korak 2.1.2: Opis tveganj (strukturiran format, ki zajema naziv, obseg, naravo, deležnike, kvantifikacijo, tolerance/odnos do tveganj (angl. risk tolerance/appetite), obravnavanje tveganj in kontrolne mehanizme, možne aktivnosti za izboljšavo, razvoj strategije in politike);
- Korak 2.1.3: Ocena tveganja (kvantitativna, semikvantitativna, kvalitativna);
- Korak 2.2: Ovrednotenje tveganj (primerjava ocenjenih tveganj glede na izbrani kriterij, stroški/koristi, regulativne zahteve, socioekonomski in okoljski dejavniki).
Prednosti uporabe SOT sta:
- enostavnost;
- primernost za brezplačen prenos znanja (odprt pristop).
Pomanjkljivosti pristopa SOT pa sta:
- ne vsebuje posebnosti obvladovanja tveganj IT;
- uporaba SOT ni tako razširjena kot uporaba BS 7799.

2.3 Standard BS 7799-2:2002

Standard BS 7799-2:2002 "Sistemi za upravljanje varovanja informacij – specifikacija s smernicami za uporabo" [05] opredeljuje naslednje korake vzpostavitve sistema za upravljanje varovanja informacij (SUVI):

- Opredeliti namen SUVI;
- Opredeliti politiko SUVI;
- Opredeliti sistematičen pristop k ocenjevanju tveganja;
- Določiti tveganja;
- Oceniti tveganja;
- Določiti in ovrednotiti možnosti za obravnavanje tveganja;
- Izbrati kontrolne cilje in kontrole za obravnavanje tveganj;
- Pripraviti izjavo o primernosti;

- Pridobiti odobritev vodstva za preostala predlagana tveganja in odobritev za uvedbo in izvajanje SUVI.

V dokumentu PD 3002: 2002 [23] so podane smernice za uporabo standarda BS 7799. V skladu s PD 3002:2002 je proces obvladovanja tveganj sestavljen iz naslednjih korakov:

- Korak 1: Opredelitev sredstev (izhod: popis sredstev, lokacij in lastnikov);
- Korak 2: Vrednotenje sredstev (izhod: pomen vsakega sredstva);
- Korak 3: Opredelitev varnostnih zahtev (izhod: seznam groženj in ranljivosti za vsako sredstvo, pravne/pogodbene in poslovne zahteve za vsako sredstvo);
- Korak 4: Ocena varnostnih zahtev (izhod: ocena za vsako varnostno zahtevo);
- Korak 5: Izračun varnostnih tveganj (izhod: seznam izmerjenih tveganj za vsako sredstvo);
- Korak 6: Opredelitev in ovrednotenje možnosti za obravnavanje tveganj (izhod: možnosti obravnavanja tveganja za vsako tveganje);
- Korak 7: Izbor varnostnih kontrol (izhod: izbrane kontrole za vsako tveganje).

Prvi korak je v [12] predstavljen z uporabo procesnega pristopa:

- Korak 1.1: Opredelitev poslovnih procesov znotraj obsega;
- Korak 1.2: Opredelitev povezav med izbranimi procesi;
- Korak 1.3: Opredelitev sredstev za izbrane procese: informacijski vhodi, izhodi, zapisi in viri (človeški viri, okolje, oprema, orodja, komunikacije in drugo).

Razumevanje in komunikacijo pri začetnih korakih je mogoče izboljšati z grafično predstavitev procesov in sredstev.

Prednosti pristopa BS 7799 so:

- razširjenost uporabe in določila v zakonodaji;
- podprtost z ustrezno dokumentacijo in izobraževanjem;
- možnost certificiranja.

Pomanjkljivosti pristopa BS 7799 pa sta:

- ni primeren za brezplačen prenos znanja (zaprt pristop);
- omejen je samo na varnostna tveganja.

3 Opis pristopa "UN-OT"

Pristop "UN-OT" (univerzitetno okolje – obvladovanje tveganj), ki ga predlagamo za obvladovanje tveganj IT

v univerzitetnem okolju, temelji na pristopu COBIT in je nadgrajen z uporabo pristopov SOT in BS 7799. Pristop "UN-OT" vključuje tiste dele prej omenjenih pristopov, ki so po našem mnenju najbolj primerni za uporabo v univerzitetnem okolju.

Pristop COBIT smo izbrali za osnovo zato, ker je odprt in ker imamo na univerzi že nekaj izkušenj z njegovo uporabo. Da bi zmanjšali pomanjkljivost glede zahtevnosti, smo ga nadgradili z uporabo pristopa SOT, ki je zelo enostaven za uporabo. Ker pa SOT ni prilagojen za tveganja IT, smo končno prilagoditev izvedli z uporabo pristopa BS 7799.

V okviru pristopa "UN-SOT" smo proces obvladovanja tveganj razdelili na 7 korakov, katerih opis sledi v nadaljevanju. Pri vsakem koraku je v oklepaju naveden pristop, ki je bil uporabljen kot osnova za njegovo oblikovanje.

- Korak 1: Strateški cilji organizacije na področju IT (pristop SOT);
- Korak 2: Opredelitev in ovrednotenje sredstev (pristop BS 7799);
- Korak 3: Ocena groženj in ranljivosti (pristop BS 7799);
- Korak 4: Merjenje tveganj (pristopi COBIT, SOT, BS 7799);
- Korak 5: Prezem tveganj (pristopi COBIT, SOT, BS 7799);
- Korak 6: Obravnavanje tveganj (pristopi COBIT, SOT, BS 7799);
- Korak 7: Nadzor in pregled (pristop SOT).

3.1 Strateški cilji organizacije na področju IT

Proces obvladovanja tveganj se prične z opredelitvijo strateških ciljev organizacije na področju IT [01], ki so ponavadi navedeni v ustreznih strateških dokumentih.

V tem koraku opredelimo cilje in jim določimo prioriteto: zelo visoka (ZV), visoka (V), srednja (S) in nizka (N). Za cilje z najvišjo prioriteto opredelimo procese. En cilj ima lahko več procesov: za vsak proces najprej opredelimo njegovo prioriteto, nato pa določimo še povezave med procesi [12]. Pri tem nam je lahko v pomoč grafična predstavitev obsega v obliki diagrama procesov.

Rezultat prvega koraka je register ciljev in procesov. Register ciljev in procesov vključuje številko cilja, naziv cilja in njegovo prioriteto. Za vsak cilj so v registru navedeni procesi, ki so potrebni za doseganje tega cilja. Opis vsakega procesa obsega številko procesa, naziv procesa in njegovo prioriteto.

Viri za obvladovanje tveganja so pogosto omejeni in ponavadi ni mogoče izvesti ocene tveganj za vse cilje že pri prvem poskusu. Zato izberemo cilje in procese, ki imajo najvišjo prioriteto in so v našem obsegu.

3.2 Opredelitev in ovrednotenje sredstev

V tem koraku najprej opredelimo sredstva za izbrane procese: informacijske vhode, izhode, zapise in vire (človeške vire, okolje, opremo, orodja, komunikacije in drugo). Potem določimo pomen vsakega sredstva: zelo visok (ZV), visok (V), srednji (S) ali nizek (N). Pravilnost ocene pomena preverijo lastniki procesov in poslovodstvo [12].

Rezultat opredelitve in ovrednotenja sredstev je register sredstev, v katerem so za vsak proces iz koraka 1 navedena pripadajoča sredstva (številka in naziv sredstva), njihova lokacija, lastnik in pomen.

3.3 Ocena groženj in ranljivosti

Standardi, ki obravnavajo obvladovanje tveganj IT, vključujejo opredelitev groženj in ranljivosti. Povezava med tveganjem, grožnjo in ranljivostjo je podana v [13]: tveganje je možnost, da bo določena grožnja izrabila ranljivost sredstva ali skupine sredstev in povzročila škodo ali uničenje sredstev.

Grožnja je možen vzrok neželene incidenta, ki ima lahko za posledico škodo na sistemu ali v organizaciji (npr. požar v sistemski sobi). Ranljivost je pomanjkljivost sredstva ali skupine sredstev, ki jih lahko grožnja izrabi [13] (npr. pomanjkanje protipožarne zaščite v sistemski sobi). Nekateri sezname splošnih groženj in ranljivosti so odprtega tipa, npr. [22], [19] in [11].

V tem koraku opredelimo grožnje in ranljivosti za sredstva, ki so navedena v registru sredstev. Za namen tega prispevka bomo kombinacijo grožnje in ranljivosti poimenovali škodni dogodek. Za vsak škodni dogodek moramo oceniti verjetnost, da se bo zgodil, ter posledice tega dogodka. V oceno groženj in ranljivosti lahko vključimo oceno priložnosti (pozitivni vidik tveganja). V tem primeru moramo oceniti verjetnost, da se bo priložnost realizirala ter posledice njene realizacije.

Rezultat ocene groženj in ranljivosti je register groženj in ranljivosti, v katerem so za vsako sredstvo iz koraka 2 navedene vse grožnje (številka grožnje in naziv grožnje), za vsako grožnjo pa vse ranljivosti (številka ranljivosti, naziv ranljivosti). Vsako kombinacijo grožnje in ranljivosti označimo kot škodni dogodek ter ocenimo njegovo verjetnost in možne posledice.

3.4 Merjenje tveganj

V četrtem koraku izračunamo tveganje glede na vrednosti sredstev, groženj in ranljivosti. Izračun tveganja je lahko kvalitativen ali kvantitativen. V pristopu UN-OT smo uporabili kvalitativen način, ki je predstavljen v tabeli 1. Vsak element tabele predstavlja stopnjo tveganja v odvisnosti od verjetnosti, da pride do nekega škodnega dogodka, in posledic, ki jih ta škodni dogodek ima za organizacijo.² Če je npr. verjetnost dogodka visoka in posledica srednja, je stopnja tveganja visoka.

Rezultat merjenja tveganj je register tveganj, ki za vsak škodni dogodek iz koraka 3 vsebuje izračunano stopnjo tveganja tega dogodka [12].

Tabela 1: Merjenje tveganj

Verjetnost /Posledica	Nizka	Srednja	Visoka	Zelo visoka
Nizka	Nizka	Srednja	Srednja	Visoka
Srednja	Srednja	Srednja	Visoka	Visoka
Visoka	Srednja	Visoka	Visoka	Zelo visoka
Zelo visoka	Visoka	Visoka	Zelo visoka	Zelo visoka

3.5 Prevzem tveganj

V petem koraku poslovodstvo določi stopnjo tveganja, ki je še sprejemljiva. Stopnja sprejemljivega tveganja določa profil tveganj, ki je osnova za profil kontrol (korak 6).

Uporabili smo stopnjo sprejemljivega tveganja, ki je podana v tabeli 2. Če je stopnja tveganja nizka ali srednja, potem bo poslovodstvo sprejelo tveganje. Če pa je stopnja tveganja visoka ali zelo visoka, potem poslovodstvo tveganja ne bo sprejelo.

Tabela 2: Stopnja sprejemljivega tveganja

Verjetnost /Posledica	Nizka	Srednja	Visoka	Zelo visoka
Nizka	Nizka	Srednja	Srednja	Visoka
Srednja	Srednja	Srednja	Visoka	Visoka
Visoka	Srednja	Visoka	Visoka	Zelo visoka
Zelo visoka	Visoka	Visoka	Zelo visoka	Zelo visoka

Rezultat tega koraka je dopolnjen register tveganj, v katerem je za vsako tveganje dodana še stopnja

sprejemljivega tveganja. Tako dopolnjen register predstavlja t. i. "profil tveganj", ki prikazuje, katera tveganja je poslovodstvo sprejelo in katerih ne [12].

3.6 Obravnavanje tveganj

V tem koraku poslovodstvo izbere način obravnavanja tveganj. Možni načini obravnavanja so štirje (angl. "4T"):

- zmanjšanje tveganja (angl. treat);
- prenos tveganja (angl. transfer);
- izogibanje tveganju (angl. terminate) in
- sprejem tveganja (angl. tolerate).

Za tveganja, za katera je poslovodstvo izbralo prvi način (tj. zmanjšanje tveganja), lahko izberemo kontrole iz standardov oziroma smernic, kot so:³

- BS 7799 [05];
- COBIT [07];
- ISF: "The standard of good practice for information security" [14];
- BSI: "The Baseline Protection Manual" [06];
- OIT: "Information Security Guideline for NSW Government – Part 3: Information security baseline controls" [22];
- SSE-CMM: "System Security Engineering Capability Maturity Model" [15] in drugi.

Rezultat obravnavanja tveganj je register tveganj, ki poleg elementov, navedenih v točki 3.5, vključuje še način obravnavanja in kontrole, stopnjo zagotavljanja ter stopnjo preostalega tveganja.

3.7 Nadzor in pregled

Z nadzorom pridobimo zagotovilo, da so kontrole ustrezne in da zaposleni razumejo in izvajajo predpisane postopke [01]. Tveganja niso nikoli statična, zato je potreben nadzor in pregled tveganj, ki zagotavlja ažurnost registra tveganj.

Rezultat obravnavanja tveganj je register tveganj, ki poleg elementov, navedenih v točki 3.6, vključuje še dopolnitve, pripravljene na osnovi poročil o nadzoru in pregledu.

4 Primer uporabe pristopa UN-OT

Uporabo pristopa UN-OT bomo podali na namišljenem primeru informacijskega sistema za področje študijske informatike.

² Vrstice tabele predstavljajo stopnjo verjetnosti, da pride do škodnega dogodka, stolpci pa resnost posledic, ki jih le-ta povzroči.

³ V ta namen so uporabne tudi naloge za pridobitev certifikatov s področja varnosti informacij [24].

4.1 Strateški IT cilji organizacije

Izhajali bomo iz predpostavke, da je eden izmed ciljev univerze nuditi uporabnikom (predvsem profesorjem in študentom) visoko kakovost administrativnih storitev. Ta cilj je povezan z informacijskim sistemom za področje študijske informatike, ki je del univerzitetnega informacijskega sistema in vsebuje naslednje module:

- obdelava vpisnih podatkov (zajem podatkov o vpisanih študentih, sezname vpisanih študentov, razna statistična poročila);
- izpitna evidenca (razpisovanje izpitnih rokov, prijavljanje na izpite, podatki o doseženih ocenah);
- evidenca diplomantov (administrativni postopki od dviga teme do zagovora, izdelava priloge k diplomam, sezname diplomantov);
- analize uspešnosti študija (prehodnost iz letnika v letnik, napredovanje čiste generacije, analiza opravljanja izpitov pri posameznih predmetih, povprečne ocene, rang lestvica študentov).

Informacijski sistem je realiziran kot spletna aplikacija, ki uporabnikom omogoča oddaljen dostop do podat-

kov, kar je še posebej pomembno za študente in profesorje, ki lahko uporabljajo sistem od koderkoli, pomembno je le, da imajo na razpolago dostop do interneta.

Za doseganje na začetku omenjenega cilja so najpomembnejši tisti procesi, v katere je vključenih največ uporabnikov. V našem primeru sta to obdelava vpisnih podatkov in izpitna evidenca, ker vključujeta komunikacijo s profesorji in študenti, ki pričakujejo dogovorjeni nivo storitev.

Rezultat tega koraka je register ciljev in procesov, podan v tabeli 3. Na enak način bi lahko opisali še druge cilje in za vsak cilj določili procese, ki so potrebni za njihovo realizacijo.

4.2 Opredelitev in ovrednotenje sredstev

Rezultat tega koraka je register sredstev, podan v tabeli 4. Vrednost sredstev se nanaša na pomembnost sredstev pri izvajanju izbranih procesov, kar je možno zapisati v komentarju. Vrednosti sredstev naj bi določili lastniki procesov in poslovodstvo. Na koncu je podan rang vsakega sredstva, ki odraža prioriteto tega sredstva.

ID Cilja	Opis	Prioriteta	ID procesa	Opis	Prioriteta	Komentar
1	2	3	4	5	6	7
C001	Zagotoviti uporabnikom visoko kakovost administrativnih storitev	Zelo visoka	P001	Obdelava vpisnih podatkov	Zelo visoka	
			P002	Izpitna evidenca	Zelo visoka	Problemi v preteklosti
Drugi cilji						

Tabela 3: Korak 1: Strateški IT cilji organizacije

ID Sredstva	Opis	Lastnik	Lokacija	Pomen sredstva	Rang	Komentar
1	2	3	4	5	6	7
S001	Programi za vodenje izpitne evidence	Vodja oddelka IT	Strežnik fakultete	Zelo visoka	1	Ključna komponenta
S002	Strežnik	Vodja oddelka IT	Sistemska soba fakultete	Visoka	2	
S003	Podatki o izpitnih rokih	Vodja študentske pisarne	Študentska pisarna	Visoka	3	
S004	Podatki o študentih, ki so prijavljeni na izpit	Vodja študentske pisarne	Študentska pisarna	Visoka	4	
S005	Podatki o vpisanih študentih	Vodja študentske pisarne	Študentska pisarna	Srednja	5	
S006	Zaposleni v študentski pisarni		Študentska pisarna	Visoka	6	
Druga sredstva						

Tabela 4: Korak 2: Opredelitev in ovrednotenje sredstev za cilj C001 in proces P002

4.3 Ocena groženj in ranljivosti

V tem koraku opredelimo grožnje in ranljivosti za vsako sredstvo. Za vsak škodni dogodek (kombinacija grožnje in ranljivosti) ocenimo verjetnost in posledico. Ocene podajo lastniki in uporabniki sredstev. Zgled za sredstvo S001 je podan v tabeli 5, ki je del registra groženj in ranljivosti.

4.4 Meritev tveganj

Sedaj moramo določiti stopnjo tveganja za vsak škodni dogodek. Stopnjo tveganja izračunamo kot produkt verjetnosti dogodka in posledice dogodka (glej tabelo 1).

V tabeli 6 je podan zgled meritve tveganj za škodne dogodke D001 (napaka v programski opremi kot posledica nizke kakovosti specifikacij programske opreme), D002 (napaka v programski opremi kot posledica nizke stopnje sodelovanja z uporabniki) in D003 (napaka v programski opremi kot posledica nizke stopnje kakovosti testiranja programske opreme), do katerih lahko pride zaradi grožnje G001 (napake v programski opremi) in ranljivosti R001 (nizka kakovost specifikacij programske opreme), R002 (nizka stopnja sodelovanja z uporabniki) in R003 (nizka stopnja kakovosti testiranja programske opreme).

4.5 Prevzem tveganja

V praksi bi bilo predrago, če bi se ukvarjali z vsemi tveganji. Zato poslovodstvo določi pomembnost tveganj in prioriteto porabe pogosto omejenih virov z namenom izboljšanja stopnje tveganja.

ID dogodka	Verjetnost	Posledica	Izračunana stopnjatveganja (2x3)	Komentar
1	2	3	4	5
D001	Zelo visoka	Zelo visoka	Zelo visoka	Ključna komponenta
D002	Zelo visoka	Zelo visoka	Zelo visoka	
D003	Zelo visoka	Zelo visoka	Zelo visoka	Ključna komponenta

Tabela 6: Korak 4: Merjenje tveganj za dogodke D001, D002 in D003

Stopnja sprejemljivega tveganja določa stopnjo tveganja, ki ga je poslovodstvo še pripravljeno sprejeti. Če stopnja sprejemljivega tveganja ni enaka stopnji izračunanega tveganja, nastopi razkorak, ki ga skušamo premostiti v koraku obravnavanja tveganj. Navedeno lahko zapišemo v obliki formule:

$$\text{Razkorak} = + \text{Izračunana stopnja tveganja} - \text{Stopnja sprejemljivega tveganja}$$

V tabeli 7 je podan zgled za dogodke D001, D002 in D003. Za dogodka D001 in D003 je stopnja sprejemljivega tveganja nizka, za dogodek D002 pa je stopnja sprejemljivega tveganja srednja. To pomeni, da bo poslovodstvo moralo ukrepati tako, da se bo izračunana stopnja tveganja znižala in postala enaka sprejemljivi, tj. nizka za dogodka D001 in D003 oziroma srednja za dogodek D002. Da bi premostilo razkorak, lahko poslovodstvo visoka in zelo visoka tveganja zmanjša, prenese ali se jim izogne.

ID Grožnje	Grožnja	ID Ranljivosti	Ranljivost	Dogodek	Verjetnost	Posledica
1	2	3	4	5	6	7
G001	Napake v programski opremi	R001	Nizka kakovost specifikacij programske opreme	D001	Zelo visoka	Zelo visoka
		R002	Nizka stopnja sodelovanja z uporabnikom	D002	Zelo visoka	Zelo visoka
		R003	Nizka kakovost testiranja programske opreme	D003	Zelo visoka	Zelo visoka
G002	Uporabniške napake	R004	Nizka kakovost funkcije nudenja pomoči	D004	Srednja	Srednja
		R005	Nizka kakovost navodil	D005	Srednja	Srednja
G003	Napake v podatkih	R006	Nizka kakovost vhodnih kontrol	D006	Srednja	Zelo visoka
		R007	Nepričakovane vrednosti	D007	Nizka	Zelo visoka
Druge grožnje						

Tabela 5: Korak 3: Ocena groženj in ranljivosti za sredstvo S001

ID dogodka	Izračunana stopnja tveganja	Stopnja sprejemljivega tveganja	Komentar
1	2	3	4
D001	Zelo visoka	Nizka	Razkorak (Zelo visoka - Nizka)
D002	Zelo visoka	Srednja	Razkorak (Zelo visoka - Srednja)
D003	Zelo visoka	Nizka	Razkorak (Zelo visoka - Nizka)

Tabela 7: Korak 5: Stopnja sprejemljivega tveganja za dogodke D001, D002 in D003

4.6 Obravnavanje tveganj

Potem ko je bila v koraku 5 določena stopnja še sprejemljivega tveganja, v koraku 6 izberemo tveganja, s katerimi se bomo najprej ukvarjali. Ker že imamo profil tveganj (korak 5), lahko ustrezno prilagodimo profil kontrol. Kontrole izberemo tako, da je stopnja tveganja, ki še ostane po uvedbi ustrezne kontrole, manjša ali enaka stopnji sprejemljivega tveganja. Navedeno lahko zapišemo v obliki formule:

$$\text{Stopnja preostalega tveganja} = + \text{Izračunana stopnja tveganja} - \text{Stopnja zagotavljanja}$$

Kontrole, s katerimi želimo doseči zahtevano stopnjo zagotavljanja, lahko izberemo iz ogrodja COBIT, standarda BS 7799 ali drugih dokumentov, navedenih v točki 3.6. Izbor ustreznih kontrol opravi poslovodstvo.

ID dogodka	Izračunana stopnja tveganja	Stopnja sprejemljivega tveganja	Kontrole	Stopnja zagotavljanja	Stopnja preostalega tveganja	Komentar
1	2	3	4	5	6	7
D003	Zelo visoka	Nizka	COBIT A15, A16 Distribucija programske opreme je dovoljena samo po izvedenem testiranju	Visoka	Nizka (Zelo visoka – Visoka)	Hitra izboljšava (ang. quick win)
			COBIT A15, A16 Testiranje se izvaja v okolju, ki ustreza produkcijskemu okolju	Zelo visoka	Nizka (Zelo visoka – Zelo visoka)	Višji stroški priprave ustreznega testnega okolja
			COBIT A15, A16 Napake po testiranju se analizirajo in izvedejo se ustrezne aktivnosti	Srednja	Srednja (Zelo visoka – Srednja)	Reaktivni pristop

Tabela 8: Korak 6: Obravnavanje tveganj za dogodek D003

V tabeli 8 je prikazan del registra tveganj, iz katerega so razvidne kontrole, ki lahko pripomorejo k zmanjšanju tveganja zaradi dogodka D003.

Pri določitvi kontrol smo izhajali iz naslednjih procesov ogrodja COBIT [07]:

- COBIT A15: "Namestiti in potrditi sisteme" ter
- COBIT A16: "Obvladovati spremembe".

V teh procesih so vključene naslednje kontrole:

- distribucija programske opreme je dovoljena samo po izvedenem testiranju;
- testiranje se izvaja v okolju, ki ustreza produkcijskemu okolju;
- napake po testiranju se analizirajo in izvedejo se ustrezne aktivnosti.

4.7 Nadzor in pregled

V tem koraku izvajamo nadzor in pregled tako, da analiziramo incidente in ugotovljamo stopnjo skladnosti s standardi in postopki. Eden izmed možnih rezultatov je del registra tveganj, podan v tabeli 9.

5 Sklep

Dva osnovna cilja upravljanja IT sta: zagotoviti, da bo uporaba IT dala pričakovane rezultate, in zagotoviti, da bodo tveganja IT ustrezno obvladovana. Uspešno obvladovanje tveganj IT je pomembno za univerzo, ker pomaga dodati in zaščititi vrednost njenih sredstev, kot je univerzitetni informacijski sistem.

V prispevku smo predstavili pristop k obvladovanju tveganj IT v univerzitetnem okolju, ki smo ga poimenovali UN-OT. Naš pristop temelji na odprtem ogrodju za upravljanje informacijske tehnologije COBIT in je nadgrajen z uporabo odprtega standarda

ID dogodka	Izračunana stopnja tveganja	Stopnja sprejemljivega tveganja	Kontrole	Stopnja zagotavljanja	Stopnja preostalega tveganja	Nadzor in pregled
1	2	3	4	5	6	7
D003	Zelo visoka	Nizka	COBIT AI5, AI6 Distribucija programske opreme je dovoljena samo po izvedenem testiranju	Visoka	Nizka	Testiranje se izvaja vedno.
			COBIT AI5, AI6 Testiranje se izvaja v okolju, ki ustreza produkcijskemu okolju	Zelo visoka	Nizka	Okolje, ki ustreza produkcijskemu, bo pripravljeno do roka X.
			COBIT AI5, AI6 Napake po testiranju se analizirajo in izvedejo se ustrezne aktivnosti	Srednja	Srednja	Vzroki napak niso bili ugotovljeni za X % vseh napak.

Tabela 9: Korak 7: Nadzor in pregled za dogodek D003

“Standard za obvladovanje tveganj” (SOT) in zaprtega standarda za varovanje informacij BS 7799. Da bi ilustrirali ustreznost pristopa UN-OT za univerzitetno okolje, smo predstavili njegovo uporabo na primeru informacijskega sistema za področje študijske informatike. V prikazanem primeru smo se lotili zmanjševanja stopnje tveganja napak v programski opremi z uvedbo kontrol, ki zagotavljajo višjo kakovost testiranja programske opreme.

Literatura

- [01] AIRMIC, ALARM, IRM: “A Risk Management Standard”. The Association of Insurance and Risk Managers (AIRMIC), ALARM The National Forum for Risk Management in Public Sector, The Institute of Risk Management (IRM), Velika Britanija, (2000). www.airmic.com, www.alarm-uk.com, www.theirm.org
- [02] AS/NZS 4360:1999 “Risk Management”. Australian standards, Australija, (1999).
- [03] Basel Committee on Banking Supervision: “Sound Practices for the Management and Supervision of Operational Risk”, (February 2003). www.bis.org/publ/bcbs96.pdf Prevod Banke Slovenije: www.bsi.si/html/basel2/03_aktivnosti/dokumenti/Operativno%20tveganje_bs-nbp.pdf
- [04] Basel Committee on Banking Supervision: “The New Basel Capital Accord - Annex 7: Detailed loss event type classification”, (April 2003). www.bis.org/publ/cp3annex.pdf Prevod Banke Slovenije: www.bsi.si/html/basel2/05_publikacije/dokumenti/C/povzetki/Operativno%20tveganje_CP_3.pdf
- [05] BS 7799-2:2002: “Sistemi za upravljanje varovanja informacij – specifikacije s smernicami za uporabo”, slovenski prevod BS 7799-2:2002: “Information security management systems – Specifications with guidance for use” (British Standard Institute). Inštitut za informacijsko varnost, Šempeter pri Gorici, 2003.
- [06] BSI: “The Baseline Protection Manual”. Das Bundesamt für Sicherheit in der Informationstechnik (BSI), Nemčija. www.bsi.de/gshb/english/download/index.html
- [07] “COBIT 3rd edition: Control Objectives for Information and Related Technology”. IT Governance Institute, ZDA (2000). www.isaca.org, www.itgi.org
- [08] “COBIT Mapping: Overview of International IT Guidance”. IT Governance Institute, ZDA, (2004). www.itgi.org
- [09] COSO-ERM: “Enterprise Risk Management Framework”. The Committee of Sponsoring Organizations of the Treadway Commission, ZDA, 2004. www.coso.org/
- [10] HB 231:2000 “Information Security Risk Management Guidelines”. Australian standards, Australija, (2000).
- [11] ICAT: “Metabase of computer vulnerabilities”. NIST, ZDA. <http://icat.nist.gov/icat.cfm>
- [12] “Implementing BS 7799-2: 2002”. British Standard Institution, Delegate Workbook for IT06, Issue 9, (september 2002).
- [13] ISO/IEC TR 13335-n “Information Technology – Guidelines for the management of IT Security”. ISO/IEC, Switzerland, (1998).
- [14] ISF: “The standard of good practice for information security”. Information Security Forum, ZDA (2003). www.securityforum.org/html/frameset.htm
- [15] ISSEA: “System Security Engineering Capability Maturity Model” (SSE-CMM – ISO/IEC 21827) The International Systems Security Engineering Association (ISSEA). www.sse-cmm.org/lib/lib.asp
- [16] IT Governance Institute: “IT Governance Executive Summary”. IT Governance Institute, USA. www.itgi.org
- [17] Jaušovec, Matjaž, Brumen, Boštjan, Welzer - Družovec, Tatjana: “Analiza varnostne ogroženosti”. Uporabna informatika, Ljubljana, letnik 2003, št. 2, str. (61–67).

- [18] Lalovič, Dušan: "Sistematični pristopi obvladovanja varnosti informacijskih sistemov združb". Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, magistrska naloga (2003).
- [19] MITRE: "Common Vulnerabilities and Exposures". MITRE, ZDA.
<http://cve.mitre.org/cve/>
- [20] MOF: "Risk Management Discipline for Operations". Microsoft Operations Framework v3.0, ZDA (January 2004)
<http://www.microsoft.com/technet/itsolutions/cits/mof/mofrisk.mspx>
- [21] NIST: "Risk Management Guide for Information Technology Systems - Recommendations of the National Institute of Standards and Technology". National Institute of Standards and Technology, ZDA (2001). <http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf>
- [22] OIT: "Information Security Guideline for NSW Government". Department of Commerce, Office of Information Technology, Australia.
www.oit.nsw.gov.au/content/2.3.Guidelines.asp
- [23] PD 3002:2002 "Guide to BS 7799 Risk Assessment". British Standards Institution, UK, (2002).
- [24] SANS-GIAC: "Seminarske naloge". The SANS (SysAdmin, Audit, Network, Security) Institute, GIAC (Global Information Assurance Certification), ZDA.
<http://www.giac.org/GCIH.php>
- [25] Šinigoj, Aleksander: "CRAMM kot orodje za oceno in upravljanje s tveganji". Slovenski inštitut za revizijo in slovenski odsek ISACA, Zbornik referatov 10. mednarodne konference o revidiranju in kontroli informacijskih sistemov (Čatež, 24.-26. 09. 2002), str. 315-324 (2002).

Nataša Žabkar je diplomirala leta 1989 na Fakulteti za računalništvo in informatiko in magistrirala leta 1998 na Ekonomski fakulteti Univerze v Ljubljani. Ukvarja se predvsem z revizijo informacijskih sistemov. Na tem področju je leta 2000 pridobila naziv preizkušeni revizor informacijskih sistemov leta in leta 2001 naziv "CISA – Certified Information Systems Auditor".

Viljan Mahnič je izredni profesor in predstojnik Laboratorija za tehnologijo programske opreme na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. V letih 1999–2003 je bil tudi prodekan za pedagoško delo. Ukvarja se z razvojem programske opreme za računalniško podprte informacijske sisteme s posebnim poudarkom na informacijskih sistemih za področje visokega šolstva. Od leta 1996 je predstavnik Slovenije v EUNIS (European University Information Systems Association), od leta 2002 pa tudi član sveta direktorjev omenjene organizacije.

Spoštovane bralke in spoštovani bralci,

člani uredništva revije Uporabna informatika

vam želimo

uspešno in ustvarjalno novo leto 2005

K RAZVOJU TELECENTROV V SLOVENIJI

K razvoju telecentrov v Sloveniji / Janez Bešter idr. / Vintar, Jelka; Jeraj, Dimitrij (ur.) / Maribor : samozal. D. Jeraj; Ljubljana : Bons / 2004 / ISBN 961-90755-2-8 / 263 str.

Knjiga je nastala v okviru dolgoročnega projekta Telecenter, ki naj bi v slovensko okolje, predvsem v bolj oddaljena naselja, vnesel elemente informacijske družbe. Po besedah urednikov je bil njun namen, da bi problem osvetlili z različnih vidikov in bi bila vsebina knjige dovolj izzivalna in dovolj informativna, da bi pritegnila somišljenike in partnerje, ki jih projekt potrebuje za uresničitev.

Avtorji prispevkov, ki obravnavajo tematiko z različnih vidikov in iz izkušenj v praksi, so informatiki, sociologi, komunikologi, arhitekti, izobraževalci idr., dva tuja avtorja pa poročata o izkušnjah uvajanja telecentrov v tujini.

Telecenter je javni prostor, opremljen z informacijsko in komunikacijsko tehnologijo, ki omogoča občanom medsebojno komuniciranje, pa tudi dostop do podatkov in informacij. V svetu se telecentri pod različnimi imeni in v različnih oblikah (civilna mreža, odprta mreža, komunalni računalniški center, javno dostopna mreža, kiberkavarna, telekoča, televas) uveljavljajo že dvajset let, v Sloveniji pa so malo poznani in pretežno na eksperimentalni ravni. Kot pravita urednika, je osnovna usmeritev uvajanja telecentrov preseganje oddaljenosti, posredno pa je to lahko tudi ena od pomembnejših rešitev za trajnostni razvoj in s tem tudi za izboljšanje kakovosti življenja.

V uvodnem poglavju z naslovom Zaostritev zamisli razpravlja Dimitrij Jeraj o pojavnih oblikah telecentrov in njihovih značilnostih ter procesih razvijanja, tako npr. o avtonomni in virtualni skupnosti, središču virtualne skupnosti, modulih kot programu telecentra, osrednji dejavnosti, telenaselju.

Naslovi poglavij lepo predstavljajo razdelitev vsebine:

1. Problem bivanja zunaj mestnih središč
2. Avtonomna skupnost – poti do nje
3. Telecenter – pot do avtonomije
4. Približevanja telecentra človeku.

Prispevki avtorjev so zelo raznoliki, pač glede na strokovno usmeritev pisca. Prav interdisciplinarni pristop k razvoju informacijske družbe, s številnimi vprašanji, ki jih odpira, je zanimiv. Srečujemo se s številnimi novimi pojmi, kot so socialni kapital, občanske informacijske mreže, ekonaselja, portal lokalne skupnosti, skrita znanja, računalniško posredovana komunikacija. Pomembna začetna vsebina telecentrov je izobraževanje, zato mu je v knjigi posvečena posebna pozornost. Zanimiv je prispevek Billa Murraya, ki predstavlja zaključke iz obsežnega poročila o razvoju telecentrov v svetu in trendih razvoja ter razpravlja o vlogi telecentra v skupnosti, o njegovem delovanju in ekonomski upravičenosti.

Neenavadna oprema in fotografije izredno lepo dopolnjujejo besedilo. Izid knjige je finančno podprlo Ministrstvo za informacijsko družbo.

K. Puc

KOLEDAR PRIREDITEV

IPSI-2005	6.–9. jan. 2005	Big Island Hawaii, ZDA	IPSI	www.internetconferences.net
IPSI-2005	17.–20. feb. 2005	Amalfi, Italija	IPSI	www.internetconferences.net
ASYN-2005	14.–16. mar. 2005	New York, ZDA	IEEE	http://vlsi.cornell.edu/asyn2005/
ISPASS-2005	20.–22. mar. 2005	Austin, ZDA	ISPASS	http://ispass.org
WCCE 2005 - World Conference on Computers in Education	4.–7. jul. 2005	Južna Afrika	IFIP	http://www.wcce2005.org.za

Pristopna izjava

Želim postati član Slovenskega društva INFORMATIKA

Prosim, da mi pošljete položnico za plačilo članarine SIT 6.700 (kot študentu SIT 2.900) in me sproti obveščate o aktivnostih v društvu.

(ime in priimek, s tiskanimi črkami)

(poklic)

(domači naslov in telefon)

(službeni naslov in telefon)

(elektronska pošta)

Datum:

Podpis:

Članarina SIT 6.700,- (plačljiva v dveh obrokih) vključuje tudi naročnino za revijo Uporabna informatika. Študenti imajo posebno ugodnost: plačujejo članarino SIT 2.900,- in za to prejemajo tudi revijo. Izpolnjeno naročilnico ali pristopno izjavo pošljite na naslov:

Slovensko društvo INFORMATIKA, Vožarski pot 12, 1000 Ljubljana.

Lahko pa izpolnite obrazec na domači strani društva: <http://www.drustvo-informatika.si>

Naročilnica na revijo UPORABNA INFORMATIKA

- Revijo naročam(o) s plačilom letne naročnine SIT 5.900
 izvodov po pogojih za podjetja SIT 17.800 za eno letno naročnino in SIT 11.900 za vsako nadaljnjo naročnino
 po pogojih za študente letno SIT 2.800

(ime in priimek, s tiskanimi črkami)

(podjetje)

(davčna številka)

(ulica, hišna številka)

(pošta)

Datum:

Podpis:

Naročnino bomo poravnali najkasneje v roku 8 dni po prejemu računa.

INTERNET

Vse bralce revije obveščamo, da lahko najdete domačo stran društva na naslovu: <http://www.drustvo-informatika.si>

Obiščite tudi spletne strani mednarodnih organizacij, v katere je včlanjeno naše društvo: IFIP: www.ifip.or.at, ECDL: www.ecdl.com, CEPIS: www.cepis.com

Popoln

E-Business Suite



Vse aplikacije zasnovane enotno.

Vse informacije na enem mestu.

ORACLE®

www.oracle.si

II 433 748/2004



900405356,4

COBISS 0

Razprave

Markus Aleksy, Ingo Bayer, Tilo Dickopp, Axel Korthaus
**IB-CUE - Theoretical Concept and Implementation Aspects for
a Software-based Information System for Innovative Budgeting
in a Competitive University Environment**

Pascal Aubry, Vincent Mathieu, Julien Marchal
**ESUP-Portail: open source Single Sign-On with CAS (Central
Authentication Service)**

Jorma Kajava, Rauno Varonen
Information Security e-Learning - from Practice to Theory

Milan Ojsteršek, Dejan Dinevski, Tomaž Klojčnik
Portal Univerze v Mariboru

Jaka Lindič, Aleš Popovič, Jurij Jaklič, Mojca Indihar Štemberger
Vpliv spletnih portalov na kakovost visokošolske izobraževalne institucije

Lea Bregar, Margerita Zagmajster, Marko Papić
**Spletni portal kot informacijski vir za spodbujanje razvoja in
raziskovanja e-izobraževanja v Sloveniji**

Viljan Mahnič, Marko Poženeš
**Uporaba podatkovnih skladišč v univerzitetnem okolju: primer
Univerze v Ljubljani**

Nataša Žabkar, Viljan Mahnič
**Obvladovanje tveganj pri uporabi informacijske tehnologije v
univerzitetnem okolju**

Nove knjige

Koledar prireditev

ISSN 1318-1882



9 771318 188001