MOCURIS – MODERN CURRICULUM IN INFORMATION SYSTEMS AT MASTER LEVEL

Albertas Caplinskas

Vilnius Gediminas Technical University, Sauletekio al. 11, LT-2040 Vilnius, Lithuania
phone: +370 2 626 107, fax: +370 2 619 905
alcapl@ktl.mii.lt

Olegas Vasilecas

Vilnius Gediminas Technical University, Sauletekio al. 11, LT-2040 Vilnius, Lithuania
phone: +370 2 767 876, fax: +370 2 700 495
olegas@fm.vtu.lt

ABSTRACT

A lot of curricula in information systems, also at master level, exists today. However, the strong need in new approaches and new curricula still exists, especially, in European area. The paper discusses the modern curriculum in information systems at master level that is currently under development in the Socrates/Erasmus project MOCURIS. The curriculum is oriented to the students of engineering schools of technical universities. The proposed approach takes into account integration trends in European area as well as the transformation of industrial economics into knowledge-based digital economics The paper presents main characteristics of the proposed curriculum, discuses curriculum development techniques used in the project MOCURIS, describes the architecture of the proposed curriculum and the body of knowledge provided by it.

1. INTRODUCTION

A lot of curricula in information systems, also at master level, exists today. Even 125 Graduate IS Programs are registered in IS World Net [Galletta, 2001]. Additionally, ACM and AIS developed so called Model Curriculum [Gorgone, Gray, 1999] that intends to serve as a set of standards upon which individual schools can base their curriculum. However, the strong need in new approaches and new curricula still exists, especially, in European area. It is for the following reasons.

First at all, integration trends in European area, intensive international student mobility, and orientation to European-wide labour market for IS graduates demand universities to adapt current curricula to the new needs and to unify the body of taught knowledge and a set of trained abilities. However, curricula in mostly of European universities have been developed in bottom-up manner, evolutionary and are influenced by university traditions, faculty skills, and regional needs. Consequently, they should be changed and adapted to the context of new Europe.

Further, the transformation of industrial economics into knowledge-based digital economics, the globalisation of information systems, changes in their nature, and the program of eEurope pose new challenges that should answer modern curricula. Modern curriculum should be constructed in such a way that it be easily changed because of constantly changing technology and business environment. A number of European universities developed already modern curricula for their own use. However, these curricula cannot be used directly by other universities because they reflect the specific of universities, which developed these curricula.

It is difficult also to apply directly the standards proposed by the Model Curriculum [Gorgone, Gray, 1999]. According to American tradition, the Model Curriculum emphasises the business aspects of the discipline. In the European area, especially, in new associated states a different tradition exists. Here a significant part of professionals in IS field is educated by engineering departments of technical universities. So, appropriate master level programs should emphasise engineering aspects of the discipline.

The Socrates/Erasmus project *MOCURIS - Modern curriculum in Information Systems at Master Level*[1] - aims to develop a curriculum in informatics with strong emphasis on information systems. The curriculum is oriented to the students of engineering schools of technical universities. We will refer further to this curriculum as to the Joint Curriculum.

The Joint Curriculum is thought as a set of standards upon which individual universities participating in the project can base their individual curricula in information systems. In the Joint Curriculum the information system is understood as a computer-based system that aims to support the business and to enhance the work and business results by means of using of information technology as an integral part of daily operation of one or more (may be distributed) organisations [Scheer, 1998], [Eriksson, Penker, 2000]. Information system includes not technologies only, but people, processes, and organisational mechanisms as well [Stohr, Kosynsky, 1992]. It should provide and maintain an integrated information flow throughout the enterprise, so that the right information should be available whenever and wherever needed, in the quality and quantity needed [Bernus, Schmidt, 1998]. Advanced information systems should support operations at multiple locations and different time zones, be distributed, multimedia, network- and agent-based, use multidimensional data analysis, data warehousing, data mining and knowledge discovery, knowledge management, mobile computing, and other advanced technologies. The software used to implement an information system can be designed as a number of custom applications, purchased as off-the-shelf standard solution, or assembled from componentware.

This paper presents main characteristics of the Joint Curriculum, curricula development techniques used in the project MOCURIS, and describes the architecture of the curriculum and the body of knowledge provided by it

2. MAIN APPROACH

The Join Curriculum is designed to meet the needs of universities participating in the MOCURIS project and does not pretend to meet the requirements of all educational systems in the European area. It is based on a master degree structure in the participating universities. The master degree in informatics with strong emphasis on information systems is thought as a professional degree that integrates information engineering, business engineering, software engineering and entrepreneurial cultures. The Joint Curriculum provides currently the following career tracks: *IS engineer*, *Knowledge engineer*, *Data manager*, *Systems integrator*, *Systems (business) analyst*, *IS consultant*, *Project manager*, *Chief information officer* (*CIO*), *E-business specialist*, *Academic* (track to Doctorate). Some career tracks have sub-tracks, for example, the track *IS engineer*, have sub-tracks *Business IS engineer*

[1] European Commission, contract 69077-IC-1-1999-1-LT-ERASMUS-CDA-2, supports the project. It is co-ordinated by Vilnius Gediminas Technical University (Lithuania). Other universities participating in the project are Braunschweig Technical University (Germany), Catalonian Technical University (Spain), Klaipeda University (Lithuania), Riga Technical University (Latvia), Stralsund University of Applied Sciences (Germany), Tallinn Technical University (Estonia), Växjö University (Sweden), and Vytautas Magnus University (Lithuania).

and *Technical IS engineer*. Besides, each university may form additional career tracks or even allow for students to form individual tracks.

The basic philosophy behind the Joint Curriculum is its European dimension. The curriculum aims ensure that graduates be prepared for positions in all European area and leave universities with the knowledge and skills required for the future competitiveness of Europe. It aims also to support student mobility between universities participating in the MOCURIS project and takes into account that Erasmus programme enables students to supplement their courses with three months to a year of additional study in one of the European countries.

Second basic principle of the Joint Curriculum is borrowed from the Model Curriculum [Gorgone, Gray, 1999]. It argues that the master's programme should include a solid body of fundamental knowledge and should not aim to teach everything that will be needed by the graduate. It should provide a basis for lifelong learning. Of course, it does not mean that training in practical issues should be ignored. The master’s programme should support productive employment and the graduate should be skilled in information systems design, maintenance, operation, and other related issues. However, the graduate should understand also the theoretical underpinnings of the discipline and its philosophical backgrounds, be aware of historical and social context of informatics, and be able to serve as a mentor to people with lowers levels of education.

Third basic principle is that all modules including theoretical ones should be IS-oriented. It contradicts to current practice when part of modules (mathematics, management, etc.) are taught by departments belonging to others faculties (schools) and prepared without orientation to the individual curriculum.

It should be noted also that the Joint Curriculum emphasises intelligent information systems and, as a consequence, agent-based technologies and other AI related methods and technologies.

The Joint Curriculum is strongly influenced by the philosophy of the Model Curriculum [Gorgone, Gray, 1999]. In accordance to the recommendations of the Model Curriculum, this curriculum has been used in MOCURIS project as “a useful reference” [Gorgone, Gray, 1999]. We follow the suggestions of Model Curriculum to integrate through the curriculum “essential career development skills including oral, written, and presentation skills; people and business skills; and ethics and professionalism” [Gorgone, Gray, 1999]. We accepted also the suggestion of the Model Curriculum that the master’s programme courses should not be treated as independent entities and that students should understand how the pieces integrate into whole. However, the Joint Curriculum cannot be viewed as an implementation of the Model Curriculum. The architecture of the Joint Curriculum differs from the architecture of the Model Curriculum, the bodies of knowledge are different, and curricula are developed using different methodologies.

In the MOCURIS project systems engineering techniques [Caplinskas, 2002] have been applied to develop the Joint Curriculum. This approach suggests that anyone study program can be seen as a system. The components of this system are courses, modules, labs, projects, etc. Each component has its requirement specification. For example, module description specifies the content, teaching considerations and other module requirements. So, any curriculum is a collection of specifications. Consequently, the curriculum development process can be treated as a requirement engineering process. In order to develop the curriculum in a systematic way, it is necessary to formulate mission of the curriculum, to gather or/and to develop mission-related materials, to analyse these materials, to formulate curriculum requirements, to define curriculum architecture, to define a system hierarchy (abstraction levels and functional decomposition for each level) for the curriculum, in top-down manner, applying iterative requirements allocation and flowdown techniques, allocate curriculum requirements in its modules, to do verification and validation of allocated requirements and, in bottom-up manner, to propagate necessary changes up to top-level of hierarchy. The Joint Curriculum was developed following this approach [Caplinskas, Vasilecas, 2002].

As in any top-down development the “hot problem” of the outlined approach is the implementation, more exactly, staffing of the curriculum. No one of the participating universities is able to implement such a curricula separately because of the shortage of human and other resources. The solution of this problem in the MOCURIS project is to accumulate resources of the partners and implement the curriculum jointly using students and teachers mobility.

3. ARCHITECTURE OF THE CURRICULUM

Different architectures are used to implement curricula. First at all, majority of modern curricula have a modular structure. Further, the abstraction levels (introductory modules, core modules, advanced modules) are also recognized in the majority of modern curricula. However, decomposition and abstraction rarely are applied in the systematically way required by the discipline of systems engineering. As a result the curriculum is insufficiently integrated, non-flexible and requires surplus resources for teaching because of unreasonable repetitions of taught knowledge in different modules. In order to enhance the integration of the knowledge, two approaches are used commonly: case-based teaching and project-oriented teaching. Both approaches are well motivated and enhance the integration of the knowledge. However, in our opinion, these approaches should be combined with modular approach and used as supporting teaching strategy only. The curriculum architecture based on one of these approaches entirely requires too many efforts to adapt it to new requirements.

The architecture of the Model Curriculum [Gorgone, Gray, 1999] provides four interrelated components: foundations, core, integration, and career tracks. The foundations component provides backgrounds, mainly business and IS foundations courses[2], required as prerequisite to the rest of the curriculum. The business foundations provide three courses: *Financial accounting*, *Organisational behaviour*, and *Marketing*. The IS foundations provide also three courses: *Fundamentals of information systems*, *Information technology hardware and software*, and *Programming, data and object structures*. The core component provides a set of primary courses required for all graduates. It includes five courses: *Data management*, *Analysis, modelling, and design, Data communications and networking*, *Project and change management*, and *IS policy and strategy*. The integration component "offers the students the opportunity to synthesize the ideas presented earlier" [Gorgone, Gray, 1999]. Integration is viewed from three perspectives: integrating the enterprise, integrating the IS function, and integrating IS technologies. Each perspective could merit a course of its own. The curriculum recommends that schools offer one of these courses or create a course that looks at all three perspectives. The career tracks component consists of four or more related electives that prepare a student for a specialisation. The tracks can be multidisciplinary, involving courses in two or more departments.

Thus, the components of Model Curriculum can be seen as the abstraction levels. At the foundations level the introductory modules are allocated, at the core level - the core modules, and at the integration and career tracks levels - the advanced modules. It seems that this architecture is sufficiently flexible because of strong modularity and of separation of core and advanced knowledge. However, for the reasons explained below, we select different architecture.

The main design objectives of the architecture of the Joint Curriculum were the flexibility and the adaptability of the curriculum. In order to achieve these objectivities, we aimed strongly separate contents and concerns at the all levels of the architecture. The top-level components of the Joint Curriculum are five independent bodies of knowledge called courses[3]: advanced concepts of information systems, theoretical fundamentals of informatics, engineering fundamentals of informatics, management issues of informatics, and philosophical and interdisciplinary issues of informatics.

Theoretical fundamentals of informatics, advanced concepts of information systems, and engineering fundamentals of informatics are three-semester courses, management issues of informatics is two-

[2] The term *course* is used in [Gorgone, Gray, 1999] as synonym of the term *module* used in this paper.
[3] We use the term *course* in another sense then it is used in [Gorgone, Gray, 1999].

semester course, and philosophical and interdisciplinary issues of informatics is one-semester course. So, four courses are taught in parallel during three study semesters.

The solution to divide the curriculum into autonomous bodies of knowledge increases adaptability of the architecture because new bodies of knowledge can be added and existing bodies of knowledge modified easily.

Each course has a multi-modular structure and implements three levels of abstraction: foundations, core, and advanced. Career tracks are implemented by groups of introductory, core and advanced modules. It means that only part of introductory and core modules are common to all career tracks. It is important to note, that curriculum provides also elective modules, which relate the discipline of information systems to the wider social, academic and business context.

The modules of the Joint Curriculum are defined in terms of abilities. It means that any module is specified in terms of prerequisite abilities, developed abilities, and trained abilities. The set of trained abilities may be empty. So, any module adds new value to the graduate. The value is added in two ways. Firstly, the module develops some new abilities. Secondly, it may train some earlier gained abilities, improve them and increase in result the level of professionalism of the students (see Table 1). Such approach increases flexibility of the architecture because it facilitates forming of new career tracks. It increases also the adaptability because the modules are seen as a kind of black boxes where implementation of modules is hidden and knowledge is encapsulated into abilities.

To define the level of abilities we introduced the following classification system:

Table 1. Classification of knowledge and abilities

No	Level of knowledge	Abilities	Level of professionalism
1	Foundations	Be able to describe basic concepts.	Be aware
2	Core	Be able to explain basic concepts and methods.	Understand
		Be able to apply general theoretical knowledge to solve model problems.	Be able to participate in student projects
3	Advanced	Be able to apply career-oriented theoretical knowledge and skills to solve real-life problems.	Be professional
		Be able to predict consequences and impacts of professional decisions.	Be expert
		Be able to propose innovative solutions of real-life problems.	Be pioneer

This system introduces two sublevels of the core level and three sublevels of the advanced level. The Joint Curriculum aims to prepare professionals only. However, it requires also that graduates gained some abilities inherent to experts and pioneers.

The proposed architecture of Joint Curriculum does not provide any integration component. We argue that the idea to provide special courses on integration of enterprise, IS functions, and IS technologies is not enough motivated. Firstly, the integration issues are wider and should be considered jointly. Apart the integration of enterprise, IS functions, and IS technologies, the curriculum should provide the integration of abilities and knowledge. It is questionable that courses on integration of enterprise, IS functions, and IS technologies can ensure for the students the opportunity to synthesize the developed abilities and ideas presented earlier in other courses. Secondly, the discipline of the information systems is multi-aspect one by its nature. The integration is one of the aspects only. Although it is very important aspect, it rather unjustified to devote separate component to this aspect. It violates the homogeneity of the curriculum. Our opinion is that usage of the aspect-oriented components requires, being consequent, constructing the whole architecture of the curriculum as the aspect-oriented one.

Our solution of the problems of integration is as follows. Firstly, we see the integration of enterprise as one of the problems of business engineering and the integration of IS technologies as one of the problems of IS engineering. Consequently, we include these issues into course *Engineering fundamentals of informatics*. Secondly, the integration of IS functions we regard as a part of main ideology of information systems and include it into course *Advanced concepts of information systems*. Thirdly, the problem of integration of knowledge we solve by including in each module the part that explains the purpose of module and its role in the context of whole curriculum. Finally, the problem of the integration of abilities we solve by providing team-based projects as a part of the course *Engineering fundamentals of informatics.*

3. THE BODY OF KNOWLEDGE

3.1. Advanced concepts of IS

The course *Advanced concepts of IS* is thought as a basic course of the Joint Curriculum that introduces the conceptual framework for all career tracks and presents essential aspects of information systems and their components. The course includes following advanced-level modules: *Modern information systems*, *Advanced database systems*, *Spatio-temporal data management*, *Knowledge management, Document management systems*, and *Artificial Intelligence*. The modules *Modern information systems*, *Advanced database systems* are addressed to all career tracks. The module *Spatio-temporal data management* is addressed to the career track *Data manager*, the module *Knowledge management* - to the career track *Knowledge Engineer,* the module *Document management systems* - to the career tracks *IS engineer*, *Knowledge engineer*, *Data manager*, *Systems (business) analyst*, *IS consultant*, *Project manager*, and *CIO,* and the module *Artificial Intelligence* - to the career tracks *IS engineer*, *Knowledge engineer*, *Data manager*, *Systems integrator*, *Systems (business) analyst*, *IS consultant*, and *Academic.*

The module *Modern information systems* discusses concepts of modern information systems and issues of professionalism including the history of computing and IS, conception of Information Society and eEuropea, the role of IS in the context of Information society and knowledge-based economy. The module *Advanced database systems* continues undergraduate database course and provides topics on advanced data management concepts including databases with incomplete information, scalability; data warehousing; intelligent query processors; OO systems; logic-based systems; active DB; agents and DB; web-based DB; multimedia databases; repositories, and other new trends in the field of databases systems. The main philosophy of the module *Spatio-temporal data management* is strongly influenced by the curriculum [Graduate, 2001]. It provides topics on the nature of spatio-temporal data, the role of spatio-temporal data in business and decision making, computational treatment of time, functionality and architectures of GIS and temporal systems, languages for spatio-temporal DB, conceptual modelling for spatio-temporal DB (geo-objects, spatial aggregation, topological relationships, spatial integrity constraints, time models, life-cycle and history, object generation and transformation), and spatio-temporal processes. The module *Knowledge management* includes topics on knowledge capital and knowledge-based economy, management approaches to knowledge management, organisational memories, and other knowledge management concepts. The module *Document management systems* adresses the issues of the creation, management, distribution, and updating of document-based information. It presents an indepth overview of the conceptual framework of the electronic document management systems, basic technologies, and development methodology. The module *Artificial Intelligence* presents a framework for understanding the role of AI in IS. It discusses advanced AI methods and techniques, especially, agents and their applications in IS.

3.2. Theoretical fundamentals

The course *Theoretical fundamentals of informatics* is thought as a course that introduces conceptual apparatus to be used in other courses, especially, in *Engineering fundamentals of informatics* and in *Management issues of informatics*. However, it provides also several modules that are taught at advanced level and are used to form career tracks data manager, system (business) analyst, IS engineer and academic. It includes following modules: *Mathematical logic, Algorithm theory, Abstract algebra, Operation research, Data analysis and applied statistics*, and *Mathematical modelling*. According to the principles behind the Joint Curriculum, all modules are IS-oriented. From one hand, it means that the module does not aims to present everything that is important from the point of view of the taught discipline (e.g. algorithm theory) and, from other hand, it can include topics that are not taught in the traditional courses of the discipline. Another peculiarity is change of the focus. All modules are focused rather to concepts and application aspects then to the proofs and other technical details. It should be noted also that all disciplines, except of the *Mathematical modelling*, are part of the Undergraduate Curriculum. So, they should be taught now at the core or at the advanced level. In this case the term *advanced* means broader in the scope but not obligatory significantly deeper in the content. Let us outline shortly the content of the modules.

The disciplines *Mathematical logic* and *Algorithm theory* are taught both at the core and at the advanced levels. The core level modules are addressed to all career tracks and advanced level modules are addressed to the career track *Academic*. The *Mathematical logic* includes topics on conceptualisation; concepts forming; the role of mathematical logic in databases and knowledge representation; reasoning including spatial, and temporal ones. The core level module of the *Algorithm theory* aims to explain different approaches how to construct an algorithm[4]: pattern matching, evolutionary (genetic algorithms), classification (neural networks), production rules, logic programming. It also explains the concept of virtual machine as the generalisation of a Turing machine. Advanced level module is devoted to the issues of complexity and algorithms analysis. The module *Abstract algebra* is an introductory one. It is addressed to the career track *Academic* and emphasizes the role of algebra in databases and informatics. The module *Operation research* (OR) is core level module addressed to all career-tracks. It explains relations between the OR and problem solving and introduces elements of the decision theory, risk management, utility theory; queuing theory, scheduling theory; and game theory with emphasis on applications in agent technologies. The module *Data analysis and applied statistics* is core level module addressed to the career track *Data manager* and provides topics on the basic data analysis methods including application of data mining and knowledge discovery methods to data warehousing. The module *Mathematical modelling* is core level module addressed to the career track *IS engineer*. It provides topics on the system conceptualisation in terms of causal relationships and functional roles of its components, and on the dynamic systems modelling concepts and methods.

3.3. Engineering fundamentals

The course *Engineering fundamentals of informatics* is thought as a cardinal course of the Joint Curriculum. First at all it is oriented to career track IS engineering, however it is very important to all other tracks, too, because the curriculum on the information systems taught by technical universities should be focused on engineering aspects of the activity in IS field. The philosophy behind this course is that the main purpose of IS is to support business activities. On the other hand, modern IS should be supported by software systems implementing components of IS. So, the students should be taught business and software engineering as well as IS engineering and concerns of each discipline should be separated. Besides, all business engineering, IS engineering, and software engineering are branches of the system engineering. Consequently, it is purposeful to introduce main concepts of engineering at

[4] Turing machines, recursive functions, and Markov algorithms should be taught at the undergraduate level.

the level of system engineering and then to explain how to apply these concepts to business systems, information systems, and software. We hope that such approach significantly reduces the time needed to teach the engineering fundamentals and increases the transparency of this course. It is worth to note, that after analysis of a number registered in [Galletta, 2001] curricula on IS, we was deeply surprised that we did not find any module indeed devoted to IS engineering issues. Although many universities have courses called *IS engineering*, mainly these courses are devoted to software engineering, data engineering, network engineering and related issues. Even the Model Curriculum [Gorgone, Gray, 1999] is not consequent from this point of view. It provides the course MSIS2000.2 *Analysis, Modelling and Design*, however, this course is a mixture of IS engineering, project management, software engineering, and even computer ethics issues.

The course *Engineering fundamentals of informatics* includes following modules: *Systems engineering*, *Business and enterprise engineering*, *Information systems engineering*, *Data engineering*, *Interfaces engineering*, *Networks engineering*, *Process engineering*, *Advanced software engineering*, *Agent technologies*, and *IS implementation methods*. The module *Systems engineering* is an introductory module addressed to all career tracks. It provides topics on systems analysis; systems conceptualisation; fact finding techniques; information analysis, evaluation, and recording; life cycle concepts; requirement engineering; systems quality metrics; systems modelling methods; systems architecture; feasibility and trade-off analysis; systems design approaches and methods; systems implementation; process engineering and reengineering; configuration management; systems integration; systems evaluation; and systems maintenance. The module *Business and enterprise engineering* is a core level module addressed to the career tracks *Systems (business) analyst*, *IS consultant*, *E-business specialist, Business IS engineer*, *Knowledge engineer*, and *Data manage*. It deepens in the business context the concepts introduced by the module *Systems engineering* and provides topics on business foundations including e-business, and virtual and networked enterprises. The module emphasises the role of ERP and IT in the business infrastructure, application of operation research methods in business, and integration of enterprise-wide information infrastructure into global information infrastructure. The module *IS engineering* is an advanced module addressed to the career tracks *IS engineer*, *Systems integrator*, *Systems (business) analyst*, *IS consultant*, and *Project manager.* On one hand, it continues the core level module taught at the undergraduate level, on other hand it relates IS engineering issues to the general systems engineering context. Additionally, it discusses issues of the development of cross-lingual and multilingual IS, spatio-temporal IS, multimedia IS and digital libraries. A number of topics is devoted to the specific of engineering of intelligent IS. The module *Data engineering* is an advanced level module addressed to the career tracks *Data manager*, *IS engineer* and *CIO*. It provides topics on enterprise data architecture, data requirements, and advanced methods of design of databases, expert databases, warehouses, and data repositories. The module continues issues presented in undergraduate DB course.

The module *Knowledge engineering* is an advanced level module addressed to the career track *Knowledge engineer*. It provides topics on ontological engineering, intelligent information storage and retrieval including semantic Web, and intelligent help-desks. The module discusses key stages, architectures, and methods in the development of a knowledge-based and knowledge management systems, and emphasises the integration of knowledge-base applications and knowledge management components into IS. The philosophy behind this module has been influenced by [Knowledge, 1998] and [Course, 2001]. The module *Interfaces engineering* is an advanced level module addressed to the career tracks *Business IS engineer*, *Systems (business) analyst*, *IS consultant,* and *CIO.* It provides topics on human-computer interaction, ergonomics, features of usability, styles of user interfaces, language engineering, user assistance, methods of user interfaces and intelligent front-end design, and methods of the evaluation of the usability of a system. The module *Process engineering* is an advanced level module addressed to the career tracks *CIO*, *IS consultant*, and *IS engineer.* It discusses issues of design and implementation IS development, put to operation, maintenance, and service processes. The module *Advanced software engineering* is a continuation of undergraduate level module and is addressed to the career tracks *IS engineer* and *Systems integrator.* It emphasises component-based engineering, Web-based engineering, security engineering, and software process issues including topics on software reuse, maintenance and reengineering. The module *Networks engineering* is an introductory module addressed to the career tracks *IS engineer*, *Data manager*, *Systems integrator*, *IS consultant*, and *CIO*. It provides topics on network architectures, protocols, and standards, emphasises the intranet and extranet issues. It discusses also computer platforms, strategies of hardware and software vendors, and the role of the middleware.

The module *Agent technologies* is an advanced module addressed to the career tracks *IS engineer*, *Systems integrator*, and *Academic* career tracks. It discusses issues of wireless technologies; mobile computing; intelligent agents, multi-agent systems; reasoning of an intelligent agent using game theory; cooperative, non-cooperative and real-time situations, telepresence, and modelling of virtual bodies and clones. The module *IS implementation methods* is an advanced level module addressed to the career tracks *IS engineer*, *Systems integrator*, *CIO*, and *Academic* career tracks. It concentrates on Web-based programming, component-based programming, XML, and scripting languages.

3.4. Management issues

The course *Management issues of informatics* is thought as a course that first at all discuses classical decision making, project management, and quality management issues in the context of IS projects. These issues are oriented to career tracks *Project manager*, *CIO*, *E-business specialist, Systems (business) analyst*, and *IS consultant.* Additionally, it provides topics on the operation management in the wider business context. These topics are oriented to career tracks *Systems (business) analyst*, *E-business specialist,* and *IS consultant.* Finally, we included also topics on e-business management which are oriented only to career track *E-business specialist.* So, the course includes modules *Decision making*, *Project management*, *Quality management*, *Operation management,* and *E-business management*. The peculiarity of the course is that it emphasises the role of Euromethod in planning, procurement and management of services and discusses supply chain. Another peculiarity is emphasis on teamworking in collaborative environments.

3.5. Philosophical and interdisciplinary issues

The course *Philosophical and interdisciplinary issues of informatics* is thought as a course that introduces for information systems wider academic and social context. The course is heterogeneous. It includes five modules: *Research principles and methods, IS ethics, Philosophical aspects of information and knowledge, Philosophical aspects of self-organising systems*, and *Mind philosophy.* The discipline *Research principles and methods* is thought as a module that gives a holistic view on research methods. It presents a systematic survey of current research principles, methods, and practices. The part of the topics is taught at the core level and is addressed to all career tracks. Another part of the topics is taught at the advanced level and is addressed to the *Academic* career track. The module *IS ethics* is core level module that is addressed to all career tracks. It presents a framework for understanding ethical issues with empasizis on information systems and discusses ethics and professionalism from both a theoretical and practical perspective. The module *Philosophical aspects of information and knowledge* is an introductory elective module that is addressed to all career tracks. It discusses the concepts of data, information and knowledge, the role of knowledge in information society and knowledge-based economy, differencies between theoretical and expert knowledge, the structure of empirical knowledge differencies between beliefs and knowledge, and contemporary theories of knowledge. The module *Philosophical aspects of self-organising systems* is an introductory elective module that is addressed to all career tracks. It discusses the concepts of synergetics, self-organisation, fractals, and artitificial life and concentrates on philosophical aspects of dynamic systems and the notion of catastroph. The module *Mind philosophy* is an introductory elective module addressed to all career tracks. It presents a framework for understanding the modern mind philosophy.

3. CONCLUSIONS

A lot of the Master Level curricula in Information Systems is developed and implemented. However, the strong need in new approaches and new curricula still exists, especially, in European area. The need exists because of integration trends in European area, intensive international student mobility, and orientation to European-wide labour market for IS graduates. The experience gained in the MOCURIS project demonstrated that systems engineering methods could be successfully applied to curricula engineering. The advantages of such approach are that it systematises the curriculum development process, facilitates the ensuring of curriculum integrity, and allows to explain reasons why some abilities or/and knowledge are required from graduates. In order to ensure the adaptability of the curriculum to fast changing business and technological environment, it is useful separate the knowledge, split knowledge body in blocks of theoretical, conceptual, engineering, management and interdisciplinary knowledge. The Master Level curricula in Information Systems oriented to the students of technical universities should emphasise engineering aspects of the discipline (enterprise engineering, IS engineering, knowledge engineering, data engineering, networks engineering, process engineering). The knowledge on business engineering and software engineering should be also provided. Because all kinds of engineering use common conceptual apparatus, it is purposeful to introduce main concepts of engineering at the level of systems engineering and then to explain how to apply these concepts to business, information, and software systems.

REFERENCES

Bernus P. and G. Schmidt (1998). Architectures of information systems. In P. Bernus, K. Mertins, G. Schmidt (eds.), *Handbook on Architectures of Information Systems*, Springer, 1998, 1-9

Caplinskas A. (2002). Curricula engineering: application of systems engineering methods to the development of university curricula. *Information Technology and Control*, 1 (22) 10-21.

Caplinskas A. and O. Vasilecas (2002). Modern curriculum in information systems: A case study. *Information Technology and Control*, 1 (22) 22-27.

Graduate School in Computer Science (2001). École Polytechnique Fédérale de Lausanne. URL: http://diwww.epfl.ch/w3forma/predoc/ResumesEPD/Spaccapietra.html

Eriksson H.-E. and M. Penker (2000). *Business Modeling with UML*. OMG Press, John Wiley & Sons, Inc.

Galletta D. (section editor) (2001), Graduate IS Programs (Masters Level), *IS World Net*, URL: http://www.pitt.edu/~isprogs/graduate.html#name

Gorgone J.T. and P. Gray (eds.) (1999). *MSIS 2000. Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems*. Association for Computing Machinery, Association for Information Systems.

Knowledge engineering overview (1998). Module desciption. School of Information Systems, Kingston University, URL: http://infosys.kingston.ac.uk/ISSchool/MID/modules/keng.html

Course Index (2001). University of Sunderland, School of Computing, Engineering and Technology, URL: http://www.cet.sunderland.ac.uk/webedit/CET/courses/course_index.htm

Scheer A.-W. (1998). *ARIS - Business Process Frameworks*. Second, completely revised and enlarged edition. Springer-Verlag, Berlin et al.

Stohr, E.A. and B.R. Kosynsky (eds.) (1992). *Information Systems and Decision Processes*. Los Alamitos, CA: IEEE Computer Society Press.