42

GiPHouse, a professional student software house

Tom J. van Weert

School of Informatics
University of Nijmegen
The Netherlands

ABSTRACT

New competencies are required for academic informatics professionals. Content and organization of informatics education will have to change. The University of Nijmegen offers application oriented informatics Master programmes aiming at these new competencies. In these programmes a professionally run student software house plays an important role. This software house has real customers for whom real-life problems are solved by professionally applying software engineering techniques. The software house is a knowledge intensive student oriented learning network supported by information technology. Learning results in this network are excellent.

Main conference themes: informatics as study topic
Educational areas: higher education
Study topics: computer science/informatics
Secondary keywords: collaborative learning, electronic mail, problem solving, software engineering, teaching methods

J. D. Tinsley et al. (eds.), *World Conference on Computers in Education VI*

CURRENT EDUCATION OF ACADEMIC INFORMATICS PROFESSIONALS

The education of academic informatics professionals is currently undergoing critical examination. For example, Peter J. Denning in his paper "Educating a New Engineer" [1] quotes commentators who say that universities "do not know how to educate graduates who know how to succeed in the new kinds of organisations and shifting world wide markets that are emerging". It is the author's experience that industry, when asked to give an opinion on young informatics graduates, in general complains that these graduates are:

- unable to close the gap between theory and practice, and unable to cope with real-life problems;
- too narrowly specialized and unable to integrate informatics with other disciplines;
- unable to adequately communicate, and not used to work in teams;
- unable to manage projects and unable to finish projects in time with the desired results.

These serious points of critique must be understood in the wider context of developments in society which require new professional competencies.

The organization of work in society is changing. Professionals are more and more working in networks of teams and need new competencies. Education will have to change to allow students to develop these competencies [2].

Organizations are changing

The organization of activities in society is changing rapidly from a hierarchical, industrial type of organization to a network type of organization. Business organizations are 're-engineered' [3]: work is re-organized with a focus on fundamental business processes. The resulting network organization shows less fragmentation of work, and team work is predominant. The organization can therefore be managed without much bureaucracy through horizontal team coordination and management by participation. Re-engineering is enabled by modern information technology [3, pp. 83-101], because it supports horizontal coordination [4].

Graduates need new competencies

In the new type of organization new competencies are needed [3, pp. 83-101]: for problem solving, for decision making as part of every job, for handling of dynamic situations, for working in more or less autonomous teams, etc. For instance, in problem solving there is a shift from deductive towards inductive

thinking (first recognize a powerful solution and then seek the problems it might solve). But students also need to develop higher conceptual, communication and cooperation skills to cope with their complex, distributed working environment.

Education has to change

Content and organization of education have always served the needs of society [5]. The traditional curriculum satisfies the needs of traditional industrial organization. And school organization, as it is today, reflects the hierarchical industrial organization of yesterday. But education as a production driven system, with the teacher as producer and the pupil as consumer, looses its usefulness. Emerging are educational organizations which allow students to develop the new competencies needed. These take the form of student oriented learning networks, supported by integrated information technology [6]. These networks allow students to take responsibilities, make decisions and work in teams, while learning to solve real problems. There is a shift in responsibility from teacher to student: from being taught to being responsible for your own learning.

Educating the new professional

The undergraduate School of Informatics (Faculty of Mathematics and Informatics, University of Nijmegen) seeks to provide students with opportunities to develop the needed professional competencies in two Master's degree programmes in Application Oriented Informatics. Extensive project work on real-life problems is one of the features of these programmes. In their Master's project (half a year of full time study) students individually design innovative solutions for real problems. Over a period of two years they also work for the equivalent of half a year of full-time study as programmer, designer, manager and quality manager in a student software house. The software house is organized as a student oriented learning network which is supported by information technology. In the following, the aims of the Master's degree programmes are explained and the functioning of the student software house is described.

Higher education in the Netherlands

The Dutch system of Higher Education is a two tier system. Students either study for four years in higher vocational education (after five years of vocationally oriented secondary education) or they follow a study of four years at the university (after six years of university oriented secondary education). University education has a first phase of four years leading to a Master's degree (Doctorandus). After graduation a student may find a job in business, industry

or the government, but may also enter a second phase of again four years of study and research work leading to a Doctor's degree.

In vocational higher education informaticians and information engineers are educated to operate professionally in conventional and current application environments in business and industry. They are immediately effective in these areas, but their professional capabilities are less effective in new problem environments or in environments subject to rapid technological change.

In university education informaticians and information engineers are educated to think in terms of abstract models. University students learn to develop "State of the Art" models for complex, innovative situations. Academic professionals need time to adapt to the professional environment of business or industry, but are then able to design innovative applications and to keep up with scientific and technological developments.

Application Oriented Informatics

Informatics or Computer Science is a branch of knowledge or study, especially concerned with establishing and systematizing facts, principles, and methods, as by experiments and hypotheses (Webster's Dictionary). Its field of study, the design and realization of programmable systems, is depicted in Figure 1. Informatics takes two forms: **Pure Informatics**, purely oriented onto itself, without reference to applications, and **Application Oriented Informatics**, oriented towards the design of applications [7]. Pure Informatics encompasses such topics as: automata theory, compiler construction, operating systems, complexity, etc. Application Oriented Informatics deals with the eventual realization of application systems in broad, economically important application areas. Two examples of such systems are:

- **Information systems** which support business or organizational processes.
- **Technical systems**, such as real-time systems and computer aided design systems, which control and support industrial processes.

Masters programmes in Application Oriented Informatics

The undergraduate School of Informatics of the University of Nijmegen offers two Master programmes in Application Oriented Informatics. One deals with information systems, the other with technical systems.

These programmes aim to educate academic informatics professionals who are able to manage, and participate in, the development of innovative software systems and their implementation into the organization; who are able to keep up to date with theoretical and practical developments in informatics and the application area, and to master specific topics at specialist level.

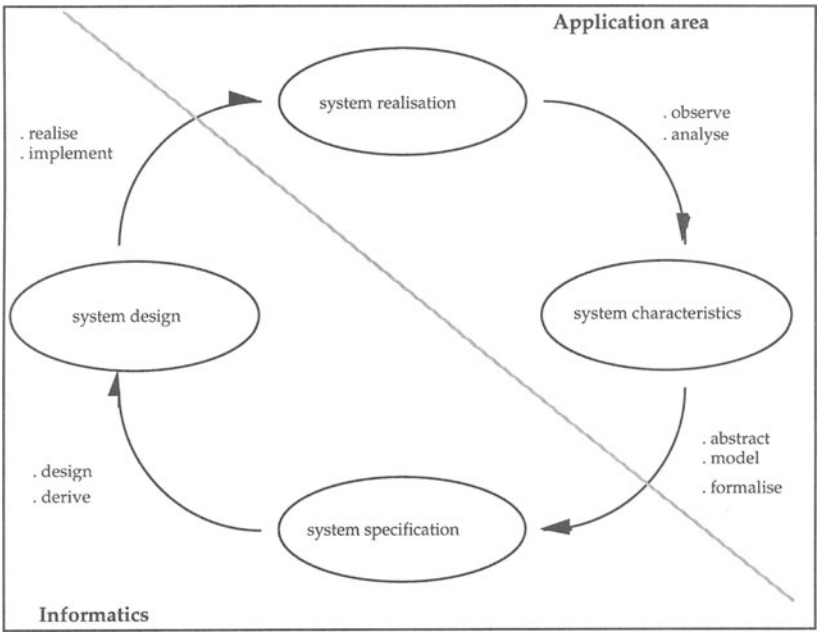


Fig. 1 Informatics as a field of study

In particular these professionals are able to recognize and analyze open problems, to abstract from these problems and to conceptually model the problems on the basis of theoretical insights (possibly extending the theoretical basis). The professional is further able to design and validate a solution in terms of the model, and to realize this solution in a software system taking into account the limits and dangers of applying informatics in real-life situations. The academic professional is able to participate in and manage teamwork, and to adequately communicate orally and in written form.

Programme content

The emphasis in the programmes is on high level knowledge and skills for developing and applying theory in the solving of complex, innovative problems. In particular, skills for analyzing, abstracting/modelling, design and realization are developed. These skills are based on knowledge of a particular application area and its integration with informatics. Basic topics in the programmes are: programming, computer technology, theoretical informatics, theoretical mathematics, applied mathematics, system design, software engineering. For the programme in information systems, this knowledge base is extended with

specializations in information storage and retrieval, information systems, management, cognitive ergonomics and business organization. For the programme in technical systems, the specialist extension is: real-time systems, telematics, image processing, process interfacing and the modelling of continuous physical processes in discrete mathematical formalisms.

Development of professional competencies

Part of the Master programmes is organized as a flexible learning network of more or less selfmanaging student teams, supported by information technology. This organization allows students to integrate old and new knowledge and skills into a set of professional competencies.

This learning network for professional competencies is realized in GiPHouse, a professional student software house, where the students play the professional roles of programmer, designer, project manager, technical manager and quality manager going through the software development cycle for a real-life problem. Professional applications are designed, realized and tested, including appropriate user interfaces and professional user and system documentation. The applications are realized for real customers within the university. Students from other departments of Nijmegen University (e.g. from Cognitive Science) play a role as information engineers in appropriate phases of the software life cycle. The organizational set-up of GiPHouse reflects that of a professional software house, and it is run in a similar way.

GiPHouse, a professional student software house

GiPHouse projects are accepted from customers within Nijmegen University, and sometimes from businesses. These customers may be research staff, management or administrative staff working in the academic hospital, in departments of Liberal Arts, Mathematics, Psychology, etc. Customers do not have to pay for the developed systems, but have to invest a considerable amount of time. Clearly they only are willing to do so, if the results are of real value in their work. It takes an academic year to complete a project. In September projects are accepted and by Christmas the system design will have been completed by teams of third-year students (GiPHouse 2). These teams will have been managed by a project manager and technical managers who are fourth-year students (GiPHouse 4). Quality assurance will also have been implemented by teams of fourth-year students (GiPHouse 4). After Christmas the design is realized by a team of second-year students (GiPHouse 1), managed and subjected to quality control by third-year students (GiPHouse 3). By the end of May, after an acceptance test by the customer, the systems are delivered.

Working in the student software house is a part-time job. During each semester students have other courses to follow and therefore are forced to do effective time management.

GiPHouse, a knowledge intensive organization

Let us again look at GiPHouse, but now from the student side. Second-year students start in GiPHouse in spring with the realization of a given design (GiPHouse 1), then go on in the fall semester of their third year to design a new system in a new project (GiPHouse 2). In the next spring semester they manage the realization of their own design by a team of second-year students, and they end in their fourth year by managing the design of yet another new system. This makes GiPHouse a knowledge intensive organization in which accumulated knowledge is shared between the students.

GiPHouse products

Some examples of systems realized in GiPHouse.

Go-Tournament System

This system allows to schedule and administrate Go-tournaments, it keeps track of results, produces ratings etc. An intricate scheduling algorithm, developed by a staff member as part of his Doctor's thesis, is at the heart of this system. The system is in use at international Go-tournaments in Europe.

Slides Lending System

This system supports the administration of art slides which are lend out to staff and students. There are tens of thousands of slides which are lent out. The system is in actual use at the department of History of Art and found of much value.

Knee Joint Visualization System

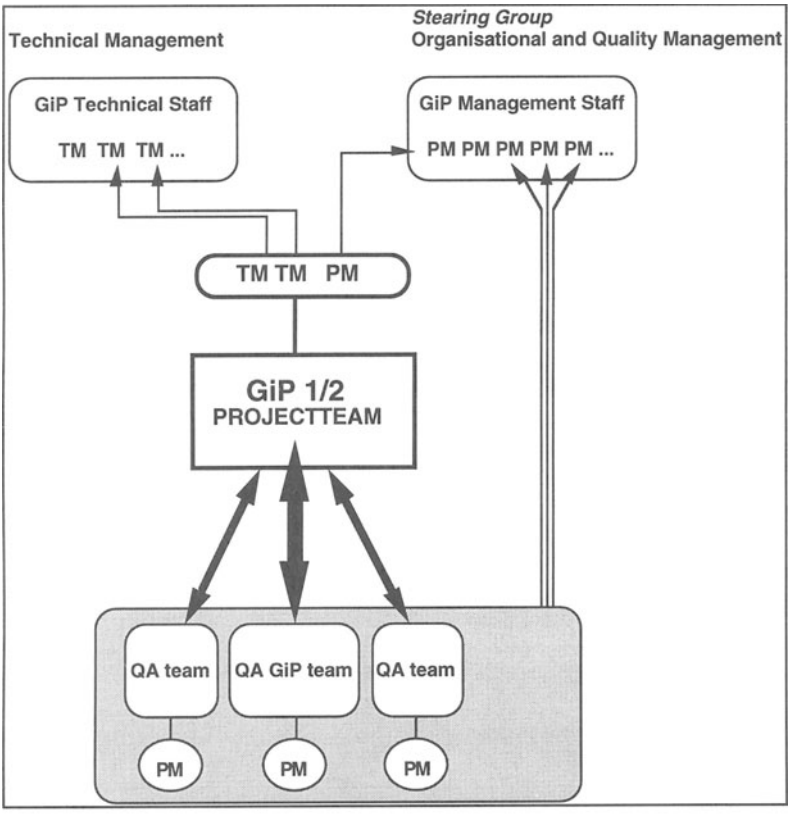
At the academic hospital in Nijmegen, research is done on a functional model of the knee joint and its ligaments. A visualisation of this model allows surgeons to decide on their strategies in knee surgery. The system visualizes the results of parameter changes in the model.

Diagnosis Comparison System

In the department of Orthopedagogy research is carried out in which a comparison is made between different models for diagnosing child behaviour. This comparison is supported by the system. A GiPHouse prototype is used in student research projects. On the basis of this prototype a system for field use is developed by the customer.

GiPHouse techniques

The actual techniques used in GiPHouse projects are selected by the students to fit the particular project. Among these are: Data Flow Diagrams (Yourdon, De Marco), structured analysis/real time (SA/RT), Data Dictionary, Structured Analysis and Design Technique (SADT), OSIS (Organization Systems and Information Systems Metamodel), NIAM (Nijssens Information Analysis Method), Conceptual Schema Design Procedure (CSDP), HOOD (Hierarchical Object Oriented Design), techniques for white-, grey- en blackbox testing.



PM	Project Manager (GiP 3/4)
TM	Technical Manager (GiP 3/4)
Project Team	Design or Realisation Team (GiP 2 or GiP 1)
QA GiP Team	Primary Quality Assurance Team
QA Team	Secondary Quality Assurance Team

Fig. 2 Management structure of GiPHouse

And furthermore: interview techniques, techniques for effective meetings, Work Break Down Structure technique, Gantt Chart, and GiPHouse defined process documentation standards, system documentation standards, user documentation standards.

GiPHouse infrastructure

The GiPHouse infrastructure has both a technical and a social side. On starting their informatics study students are provided with a personal working environment on a UNIX platform. Students may also connect to this environment from home through the television cable network (an experiment) or through the telephone network. One of the features of this working environment is a system (Andrew Message System) for electronic communication with fellow-students, staff and others (locally, nationally or internationally). This communication system is heavily used and without it an organization such as GiPHouse cannot be run. There is a special room for the GiPHouse student teams with dedicated equipment (UNIX based, Intel 486 based, 68000 based), software (programming environments, management tools, case tools) lockers for storing materials, etc. This room also serves as a social meeting point.

Learning to help yourself

Students in GiPHouse have to learn to "help themselves". They start their projects with a general description of half a page, the address of the customer, a GiPHouse manual explaining the basics of the organization [8] and of techniques to be used, and in addition some GiPHouse standards. After that it is up to the students to find suitable techniques to tackle their project tasks, to organize, plan, manage their teams etc. However, students who have been involved in earlier phases of GiPHouse, bring their knowledge over on younger students, thus providing a real learning network. This also applies to communication and presentation skills which students develop during their work in GiPHouse.

Helping yourself to techniques is supported by Software Engineering courses which run in parallel with GiPHouse. These courses deal with traditional software engineering methods and techniques, object oriented methods and techniques, facilities management and information planning. And of course there is staff support to the students in GiPHouse. This includes the support by a GiPHouse Director who is a professor sponsored by a well-know Dutch software house (BSO/Origin) with extensive experience in development of large software systems and quality control.

Managing GiPHouse to success

GiPHouse is mainly managed by the students themselves (GiPHouse 3 and GiPHouse 4). Some deal with the technical aspects of the work, others with the management aspect. Members of staff work together with the students as one management team to realize the GiPHouse goals: to deliver on time an effective, quality software system to the customer and to meet the learning goals. The management structure of GiPHouse is indicated in Figure 2.

In GiPHouse a well-documented, industry accepted waterfall method of project process management is used. Such a waterfall method was selected because it provides management with milestone products which can be assessed. At the moment also other methods, such as object oriented methods, are being tried out.

The managing of GiPHouse to success is a complicated and critical process. It has taken several years to find the right set of checks and balances for realizing both the production goal and the learning goals of GiPHouse.

Students find GiPHouse to be a demanding exercise, but are very satisfied by their accomplishments, both with respect to the product, a real-life software system, and with respect to what they have learned. The academic staff is impressed by what the students are capable of learning, how quickly they can learn and by their motivation. All in all, GiPHouse provides for a very stimulating learning and teaching environment.

REFERENCES

1. Denning, Peter J. (1992) *Educating a New Engineer*, Communications of the ACM, **35** (12) pp. 83-97.

2. van Weert, Tom J. (1994) *Education and computers: Who is in control?*, in 13th World Computer Congress 94, Volume 2 (eds. K. Brunnstein and E. Raubold), Elsevier Science B.V. (North-Holland), Amsterdam, pp. 619-626.

3. Hammer, M. and Champy, J. (1993) *Re-engineering the corporation*, Nicolas Brealy Publishing, London.

4. Quinn, J. B. and Paquette, P. C. (1990) *Technology in services: Creating organizational revolutions*, Sloan Management Review (winter 1990), pp. 66-77.

5. Toffler, A. (1970) *The third wave*, Bantam Books, New York, pp. 29-30.

6. van Weert, Tom J. (1992) *Informatics and the organization of education*, in The impacts of informatics on the organization of education (eds. B. Samways and T. J. van Weert), Elsevier Science Publishers B. V., Amsterdam, pp. 15-24.

7. van Weert, Tom J. (1992) *Application Oriented Informatics and Informational Disciplines: A symbiosis bridging the gap*, in Information Processing 92, Volume II (ed. R. Aiken), Elsevier Science Publishers B. V., Amsterdam, pp. 144-150.

8. Manuals for GiPHouse 1, GiPHouse 2, GiPHouse 3, GiPHouse 4, the GiPHouse Customer, and GiPHouse Staff (all in Dutch).