Specifying and comparing informatics curricula through UCSI®

Fred Mulder
Faculty of Engineering (Informatics), Open University, PO Box 2960 6401 DL Heerlen, The Netherlands, e-mail: fred.mulder@ouh.nl

Anneke E.N. Hacquebard
Consultancy for Informatics and Education Hacquebard bv Keppelseweg 48, 6999 AR Hummelo, The Netherlands
e-mail: hacque@pi.net

Abstract

UCSI is a classification system primarily aiming at informatics education and training. Since its conception in 1992 it has been used in various pilot projects in different educational sectors in the Netherlands. In this paper we discuss the origin, the scope and possible applications of UCSI. Also some results are presented of a study in which curricula of four Dutch educational programmes for informatics engineers were compared. These results both exemplify the approach and give indications for other relevant curriculum studies, especially in an international context.

Keywords

Informatics, other disciplines, curriculum (general), taxonomies, levels of competence, educational profiles

Informatics in Higher Education F. Mulder & T. van Weerts (Eds.)

1 INTRODUCTION

Not satisfied with the continuing struggle on terminology in our field the Unified Classification Scheme for Informatics (UCSI) was introduced in 1992 as a new classification system for the discipline 'informatics' (Mulder, 1992; Mulder *et al.*, 1992). Another reason for its introduction was the fact that the various available systems, although of reputation and certainly adequate on their own, were too specific in their approach of the discipline and therefore virtually incompatible.

In the beginning of the nineties we were running a project to characterize and compare in a quantitative manner higher informatics education in the Netherlands. However, the needed overall 'measuring instruments' were lacking. Of course the more familiar qualitative descriptive method could have been followed, but that implied drawbacks such as the impossibility to achieve objectivity, reproducibility, accuracy and depth. UCSI was introduced to avoid these drawbacks.

2 UCSI: MERGE OF EXISTING SYSTEMS

UCSI has been developed on the basis of a number of recognized, specific source systems or schemes and merges them into one overall and broader system.

2.1 Six sources, three approaches

UCSI is based on six sources:

- **Computing Reviews 1991,**
  the well-known and very detailed taxonomy for the classification of literature of the journal Computing Reviews, which is a four level system (we have used the version presented in Coulter (1991));
- **Computer Abstracts 1990,**
  the global system for the classification of literature of the journal Computer Abstracts, which exists since 1957 and is a one page, two level system (we have used the substantially updated 1990 version, see Computer Abstracts (1990));
- **DPMA 1991,**
  the three level knowledge tree, which is the basis of DPMA's model curriculum for 'information systems' (see Longenecker and Feinstein (1991); DPMA stands for the Data Processing Management Association from the USA);
- **ACM and IEEE-CS 1991,**
  the matrix with knowledge elements on the basis of which a whole spectrum of model curricula varying between 'computer science' and 'computer engineering' has been developed by ACM and IEEE-CS (see ACM/IEEE-CS (1991); ACM = Association for Computing Machinery, and IEEE-CS = Institute of Electrical and Electronics Engineers Computer Society, both from the USA);

- **IEEE-CS 1983,**
  the older, comprehensive modular domain description of 'computer science and engineering' of IEEE-CS, which was the starting-point for various educational programmes (IEEE-CS, 1983);
- **IFIP 1987,**
  the older, modular curriculum for 'information systems', developed by BCS and adopted by IFIP (see Buckingham *et al.* (1987); BCS = British Computer Society, and IFIP = International Federation for Information Processing).

In selecting the sources, we have taken care of finding a fair balance between the three 'classical' and originally rather separate approaches of the discipline in the USA:

- computer science;
- computer (science and) engineering;
- information systems.

An interesting source is the fourth one, ACM/IEEE-CS (1991), which represents a successful common effort to bring together the first two of the three approaches, building on the influential paper by Denning *et al.* defining 'computing' as a discipline. A similar joint activity of ACM, DPMA (which now is called AITP = Association of Information Technology Professionals) and AIS (= Association for Information Systems) has explored the third approach or field of 'information systems' (IS) and has been reported on recently (Davis *et al.*, 1997). The resulting model curriculum IS'97 is based on an IS body of knowledge which still has to be added as a seventh, new source underlying UCSI. A first quick study, however, shows a large similarity with the earlier approach of DPMA (1991), so that there is no need to change UCSI in its latest three level release (as presented in Figure 1).

2.2 Design process, outcome and maintenance

UCSI has resulted from an iterative process of going through the six source schemes and their related terminologies (which sometimes differ substantially). In order to manage the rather extensive information contained in the descriptions of the existing schemes as well as in the new scheme UCSI, a database has been designed.

In each iteration step a particular level of UCSI was specified by refinement into the next level. Input for each step was a carefully chosen set of UCSI descriptors; sometimes the terms used are similar to existing ones, sometimes not. Other input consisted of the items collected from the various source schemes. Allocation of these items to the UCSI descriptors yields a comprehensive specification for each descriptor. The next iteration step then starts with the conversion of these descriptor specifications into a limited number of new UCSI descriptors, one level deeper.

The final outcome of the process is a specification of the discipline 'informatics' in a four level knowledge domain tree. Figure 1 shows four (numbered) main domains at the highest UCSI level. Each main domain - apart from 'miscellaneous' - is further specified by at most five domains, which in turn are specified by not more than four subdomains. The subdomains are specified at the fourth level (not shown in Figure 1) by sets of selected descriptors. Except for the fourth level, it is easy to show the relations between UCSI and the source schemes, thanks to the use of the database mentioned.

1 Computer systems

1.1 Hardware structures and digital systems
- 1.1.1 Digital components
- 1.1.2 Circuits and structures
- 1.1.3 Digital systems
- 1.1.4 Integrated circuits
- 1.1.9 Miscellaneous

1.2 Computer architecture
- 1.2.1 Memory systems
- 1.2.2 Processor architectures
- 1.2.3 Instruction sets and data representation
- 1.2.4 Assembly languages
- 1.2.9 Miscellaneous

1.3 Interfacing and peripherals
- 1.3.1 Interfacing technology
- 1.3.2 Input/output systems
- 1.3.3 Storage systems
- 1.3.4 Peripheral devices
- 1.3.9 Miscellaneous

1.4 Communication and networks
- 1.4.1 Communication technology
- 1.4.2 Network architectures
- 1.4.3 Network protocols
- 1.4.4 Network management
- 1.4.9 Miscellaneous

1.5 Operating systems and system software
- 1.5.1 File and device management
- 1.5.2 Process management
- 1.5.3 System management
- 1.5.4 System software
- 1.5.9 Miscellaneous

1.9 Miscellaneous

2 Software systems

2.1 Programming languages and environments
- 2.1.1 Language constructs
- 2.1.2 Specific languages and environments
- 2.1.3 Language processors
- 2.1.4 Language concepts
- 2.1.9 Miscellaneous

2.2 Software architecture
- 2.2.1 Data structures
- 2.2.2. Algorithms
- 2.2.3 Programming techniques and strategies
- 2.2.9 Miscellaneous

2.3 Software engineering (SE)
- 2.3.1 Software requirements and specification
- 2.3.2 Software development process
- 2.3.3 Software exploitation
- 2.3.4 SE methods, techniques and tools
- 2.3.9 Miscellaneous

2.4 Artificial intelligence (AI)
- 2.4.1 AI fields
- 2.4.2 AI methods, techniques and tools
- 2.4.3 AI concepts
- 2.4.9 Miscellaneous

2.5 Theory of computing
- 2.5.1 Formal languages and automata
- 2.5.2 Computability and complexity
- 2.5.3 Semantics of programs
- 2.5.4 Information and coding theory
- 2.5.9 Miscellaneous

2.9 Miscellaneous

Figure 1 (First half) Three level specification of the discipline 'informatics' according to UCSI

3 Information systems

3.1 Information bases (IB)
- 3.1.1 Language constructs
- 3.1.2 Specific languages and environments
- 3.1.3 IB management environments
- 3.1.4 Information base concepts
- 3.1.9 Miscellaneous

3.2 Information systems architecture
- 3.2.1 Information models
- 3.2.2 Process models
- 3.2.3 Event models
- 3.2.9 Miscellaneous

3.3 Information systems (IS) engineering
- 3.3.1 Information analysis
- 3.3.2 IS development process
- 3.3.3 IS exploitation
- 3.3.4 IS engineering methods, techniques and tools
- 3.3.9 Miscellaneous

3.4 Interaction and presentation (IP)
- 3.4.1 IP fields
- 3.4.2 IP methods, techniques and tools
- 3.4.3 IP concepts
- 3.4.9 Miscellaneous

3.5 Theory of information systems (IS)
- 3.5.1 Systems theory for IS
- 3.5.2 Communication and linguistics
- 3.5.3 Human information processing
- 3.5.9 Miscellaneous

3.9 Miscellaneous

4 Context of informatics

4.1 Management and informatics
- 4.1.1 Project management
- 4.1.2 Systems management
- 4.1.3 Quality management
- 4.1.4 Information strategy and planning
- 4.1.9 Miscellaneous

4.2 Domain specific and dedicated systems
- 4.2.1 Dedicated computer systems
- 4.2.2 General purpose software systems
- 4.2.3 Domain specific information systems
- 4.2.9 Miscellaneous

4.3 Informatics operational environment
- 4.3.1 Informatics suppliers
- 4.3.2 Informatics clients
- 4.3.3 Informatics profession
- 4.3.4 Informatics education
- 4.3.9 Miscellaneous

4.4 Informatics in society
- 4.4.1 History of informatics
- 4.4.2 Social and individual issues
- 4.4.3 Economic, political and legal issues
- 4.4.4 Cultural and philosophical issues
- 4.4.9 Miscellaneous

4.9 Miscellaneous

Figure 1 (Second half) Three level specification of the discipline 'informatics' according to UCSI (release 1.2, 1997); there is a detailed fourth level which is not shown.

Once created, maintenance of the system is a simple task. This paper describes UCSI in release 1.2 which implies a minor update on the first three levels:
- a number of descriptors has been changed;
- some subdomains were combined into one subdomain or were swapped;
- 'miscellaneous' is indicated now by a '9' instead of an 'm'.

At the fourth level changes are substantial: for example, to be more complete and precise, descriptors have been added from new sources. A rather high degree of dynamics is however acceptable at this level, since - contrary to the three upper levels - the fourth level does not provide classifying items.

Summarizing: UCSI has been designed to offer a more balanced and broader view of 'informatics' than each of its individual sources. The main advantage of UCSI is that it 'unifies' a number of more specific approaches of the discipline 'informatics'.

3 CURRICULUM SPECIFICATION AND COMPARISON: INGREDIENTS

3.1 Mapping curriculum information

Comparing educational programmes can be done at different moments (for one and the same institute) or between different institutes (at the same moment). In the first case one can use the approach and terminology of that institute and obtain - dependent on the quality of the data - a comparison of the curriculum over successive years. The second case is complicated by the fact that approach and terminology at the various institutes are rarely identical. Such a comparison requires an instrument based on accepted standards.

For each institute the curriculum information about courses, laboratories and projects (referring to aspects as contents, educational goals, examination, study load, position within the programme) should be mapped onto a standardized scheme. On the basis of the mapped data a quantitative comparison of curricula across different institutes is straightforward. As far as the informatics contents is concerned UCSI offers an adequate unified scheme. But that is not sufficient.

3.2 Other disciplines: Dewey

Generally informatics educational programmes also contain noninformatics subjects. This may not be the case for short, specific courses and training components, but always holds for full programmes extending over a number of years. As a consequence another ingredient is needed for specification of this noninformatics component.

This has been found in the well-known general library classification system of Dewey/Mitchell *et al.*, 1996 (for an impression see Figure 2). This system was designed and introduced in 1876 by Melvil Dewey (size: 44 pages) and was in 1996 in its 21st edition (various printed volumes, but also online available). Dewey's system is multi-level and covers all disciplines in detail; it includes informatics, but in our opinion not in an adequate way. Therefore UCSI is used for informatics and Dewey's system for the other disciplines, but only in a global and very selective way.

3.3 A taxonomy of views

Any classification scheme should take account of the nature, depth or level of the components of the curriculum. For instance, it obviously matters what students do with a subject as 'operating systems' (UCSI domain 1.5): do they merely receive hands-on Windows user experience, or skills to design new operating systems, or theoretical insight in the concepts behind operating systems? The additional ingredient therefore has to be some taxonomy.

000 Generalities
- 010 Bibliography
- 020 Library & information sciences
- 030 General encyclopedic works
- .
- 090 Manuscripts & rare books

100 Philosophy & psychology
- 110 Metaphysics
- 120 Epistemology, causation, humankind
- 130 Paranormal phenomena
- 140 Specific philosophical schools
- 150 Psychology
- .
- .
- .
- .
- .
- 190 Modern western philosophy

200 Religion
- 210 Philosophy & theory of religion
- .
- 290 Comparative religion & other religions

500 Natural sciences & mathematics
- 510 Mathematics
- 520 Astronomy & allied sciences
- 530 Physics
- .
- 590 Animals

600 Technology (Applied sciences)
- 610 Medical sciences Medicine
- 620 Engineering & allied operations
- 630 Agriculture & related technologies
- 640 Home economics & family living
- 650 Management & auxiliary services
- .
- 690 Buildings

700 The arts Fine and decorative arts
- 710 Civic & landscape art
- .
- 790 Recreational & performing arts

Figure 2 Fragments from the Dewey Decimal Classification (part 2).

300 Social sciences

310	Collections of general statistics
320	Political science
330	Economics
340	Law
350	Public administration & military science
360	Social problems & services; associations
370	Education
380	Commerce, communications, transportation
.	
.	
390	Customs, etiquette, folklore

400 Language

410	Linguistics
420	English & Old English
.	
.	
490	Other languages

800 Literature & rhetoric

810	American literature in English
820	English & Old English literatures
830	Literatures of Germanic languages
840	Literatures of Romance languages
850	Italian, Romanian, Rhaeto-Romanic
860	Spanish & Portugese literatures
870	Italic literatures Latin
880	Hellenic literatures Classical Greek
890	Literatures of other languages

900 Geography & history

910	Geography & travel
920	Biography, genealogy, insignia
.	
.	
990	General history of other areas

Figure 2 Fragments from the Dewey Decimal Classification (part 2).

The experience from the pilot projects is that an educational goal taxonomy like Bloom's (Bloom *et al.*, 1956) is not appropriate. In such a taxonomy one is forced to descend to micro level to be able to assign educational goals to small units. Apart from the fact that the programme information does in general not offer enough detail, it also seems pointless to go into so much detail.

Here a global taxonomy is used in which we distinguish six so-called 'views', the view of:

1. **concept,**
   which emphasizes the mastering of theoretical concepts and abstract problems;
2. **model,**
   in which models are developed for concrete problems and situations;
3. **application,**
   which accentuates the application of theories and models to obtain practical results;
4. **design,**
   in which systems are designed from scratch to realization;
5. **use,**
   which concentrates on working with ready for use systems (both generic and specific);

6. **knowledge,**
   which deals with bare knowledge and insight in a specific subject.

Originally a set of five views was proposed (Mulder, 1992); here one of these views has been split into two new ones: 'application' (3) and 'use' (5). Views 1, 2 and 4 are more or less similar to the three so-called 'basic processes' or 'paradigms' in Denning *et al.* (1989), along which the discipline 'computing' can be considered, namely:

- theory, rooted in mathematics;
- abstraction , rooted in science (the experimental method);
- design, rooted in engineering.

4 EXAMPLES OF UCSI-BASED CURRICULUM COMPARISON

With UCSI we have compared the curricula of all informatics engineer's programmes that exist in the Netherlands; they run at the three Technical Universities (TU's) of Delft, Eindhoven and Twente and at the Open University (OU). The institutes' information on the programmes for students served as input in order to specify the various curriculum components on the basis of UCSI and Dewey. At the TU's we took all the courses of the first two years, whereas at the OU we took the compulsory part. Accidentally these have the same size in terms of study load: 84 so-called study points (each point the equivalent of 40 hours of study) or 3360 hours of study load in total which equals 50% of the full programmes.

For each course of the individual curricula the following data were assembled in a database:

- name, code, position in the curriculum, educational setting (class, lab, etc.) and study load (institute's data);
- the UCSI or Dewey specification, dividing the programme components - when necessary - over various domains (we decided on a global specification, not deeper than UCSI level 2);
- the corresponding views, choosing from the six types as described in subsection 3.3.

On the basis of these data all kinds of reports may be produced answering questions with respect to the curricula, both for a particular institute and comparatively across the institutes. Let us give two examples.

4.1 Different streams at one institute

In Table 1 we analyse the OU curriculum with respect to the four specializations or 'streams' that can be chosen by students. Course data have been totalled over the various UCSI (main) domains. For the noninformatics subjects the basis is the

Dewey classification, but collected in four main disciplines, as shown in Table 2. The remaining part of the curriculum, elective courses (60 study points) and a substantial final project (24 study points), is more difficult to specify, being student dependent. No unique specification can be made for this component.

Other than is the case with just a list of subjects or courses, the data from Table 1 'speak for themselves'. The variance in emphasis on the four different streams is clearly visible in the study load data of the compulsory component. It would be both possible and useful to draw conclusions from the table's data, but in this paper we confine ourselves to the mere presentation of 'numeric prints' of the educational programme of one institute in its various appearances.

Table 1
UCSI specification of the informatics engineer's programme of the Dutch Open University: version 1994/1995, compulsory part (50% of the full programme), four streams: CS = computer systems, SS = software systems, IS = information systems, I&M = informatics and management

UCSI main domain and domain		*Study load (in hours) per stream*			
Code	*Descriptor*	*CS*	*SS*	*IS*	*I&M*
1	*Computer systems*	*1044*	*564*	*288*	*564*
1.1	Hardware structures and digital systems	204	84	84	84
1.2	Computer architecture	216	180	36	180
1.3	Interfacing and peripherals	120	36	24	36
1.4	Communication and networks	144	144	144	144
1.5	Operating systems and system software	120	120	–	120
1.9	Miscellaneous	240	–	–	–
2	*Software systems*	*786*	*1266*	*462*	*666*
2.1	Programming languages and environments	276	468	192	276
2.2	Software architecture	240	276	120	240
2.3	Software engineering	120	120	120	120
2.4	Artificial intelligence	30	42	30	30
2.5	Theory of computing	120	120	–	–
2.9	Miscellaneous	–	240	–	–
3	*Information systems*	*522*	*750*	*990*	*522*
3.1	Information bases	168	228	228	168
3.2	Information systems architecture	120	180	180	120
3.3	Information systems engineering	114	222	222	114
3.4	Interaction and presentation	120	120	120	120
3.5	Theory of information systems	–	–	–	–
3.9	Miscellaneous	–	–	240	–

Table 1 (continued)

4	*Context of informatics*	*48*	*60*	*252*	*288*
4.1	Management and informatics	–	12	204	–
4.2	Domain specific and dedicated systems	–	–	–	–
4.3	Informatics operational environment	24	24	24	24
4.4	Informatics in society	24	24	24	24
4.9	Miscellaneous	–	–	–	240
	Informatics (total)	**2400**	**2640**	**1992**	**2040**
–	*Systems*	120	–	12	–
–	*Mathematics*	600	720	840	630
–	*Engineering*	240	–	–	–
–	*Management*	–	–	516	690
	Other disciplines (total)	**960**	**720**	**1368**	**1320**
	Total	**3360**	**3360**	**3360**	**3360**

4.2 Curricula at different institutes

Another example is presented in Table 2 where we compare the different informatics engineer's programmes in The Netherlands. Again it is not within the scope of this paper to draw conclusions, still the large differences between the institutional profiles are noteworthy.

Table 2
UCSI specification of the informatics engineer's programmes of the three Dutch Technical Universities (TU1, TU2 and TU3) and the Open University (OU): versions 1994/1995, 50% of the full programme (compulsory); for the OU two of the four streams are shown (see also Table 1)

UCSI main domain		*Component (%) per educational programme*				
Code	*Descriptor*	*TU1*	*TU2*	*TU3*	*OU/SS*	*OU/I&M*
1	Computer systems	21.2	9.0	13.5	16.8	16.8
2	Software systems	21.2	40.1	28.8	37.7	19.8
3	Information systems	10.6	2.1	9.8	22.3	15.5
4	Context of informatics	5.9	4.8	13.1	1.8	8.6
–	Informatics, not assigned (elective)	7.0	–	2.3	–	–
	Informatics (total)	**65.9**	**56.0**	**67.5**	**78.6**	**60.7**
–	Mathematics	23.5	43.0	27.1	21.4	18.8
–	Other disciplines (mainly Management)	10.6	1.0	5.4	–	20.5
	Other disciplines (total)	**34.1**	**44.0**	**32.5**	**21.4**	**39.3**

Table 2 is presented in terms of the UCSI main domains, but could just as easily be presented at any UCSI level of detail since the data are available in the database. From the data in Table 2 not only a comparison of different programmes can be made, but other questions can be answered as well, for example with respect to the relative use of the six views within the programmes and to the development throughout the years of a programme within one institute.

5 CONCLUSION

UCSI is a useful instrument for 'measuring' informatics educational programmes, as shown in the examples in this paper. The results of this kind of study are important for beginning students as well as for industry and government who take in graduates. The various informatics educational programmes can be mapped on 'numeric prints', showing the similarities and the differences. Similar comparative studies can be done at an international level, for example comparing the two curricula on computing science (ACM/IEEE-CS, 1991) and information systems (IS'97, see Davis *et al.*, 1997).

6 ACKNOWLEDGEMENTS

The authors would like to thank Deny Smeets and Tinie Veenstra-Strijland for their valuable help in running a pilot project in which UCSI was developed.

7 REFERENCES

ACM/IEEE-CS (1991) *Computing curricula 1991*, Report of the ACM/ IEEE-CS Joint Curriculum Task Force. ACM, New York; IEEE-CS, Los Angeles.

Bloom, B.S. [ed.] (1956) *Taxonomy of educational objectives, Handbook 1: Cognitive domain.* David McKay, New York.

Buckingham, R.A., Hirschheim, R.A., Land, F.F.and Tully, C.J. (1987) Information systems curriculum: a basis for course design, in *Information systems education: recommendations and implementation* (eds. R.A. Buckingham, R.A. Hirschheim, F.F. Land and C.J. Tully), BCS Monograph in Informatics. Cambridge University Press, Cambridge, 14-133.

Computer Abstracts (1990) Classification scheme. *Computer Abstracts,* **34**, table of contents in each volume number.

Coulter, N.S. [ed.] (1991) Update to the Computing Reviews Classification System. *Computing Reviews,* **32** (1), 5-50.

Davis, G.B., Gorgone, J.T., Couger, J.D., Feinstein, D.L. and Longenecker Jr, H.E. (1997) IS'97: Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *Data Base*, **28** (1), 1-94.

Denning, P.J., Comer, D.E., Gries, D., Mulder, M.C., Tucker, A.B., Turner, A.J. and Young, P.R. (1989) Computing as a discipline. *Communications of the ACM,* **32** (1), 9-23.

Dewey, M. [devisor] / Mitchell, J.S. [ed.] (1996) *Summaries DDC 21: Dewey Decimal Classification,* edition 21. Forest Press, New York. Online version: http://www.oclc.org/fp/.

Hacquebard, A.E.N., Mulder, F., Smeets, D. and Veenstra-Strijland, T. (1992) Wat doet een opleiding aan informatica? Onderzoek het met UCSIE!. *TINFON,* **1** (3-4), 96-99 [in Dutch].

IEEE-CS (1983) *The 1983 IEEE Computer Society model program in computer science and engineerin.* Report of the Model Program Committee of the Educational Activities Board. IEEE, Los Angeles.

Longenecker Jr, H.E. and Feinstein, D.L. [eds.] (1991) *Information systems. The DPMA model curriculum for a four year undergraduate degree.* Report of the DPMA Curriculum Task Force 1990s. DPMA, Park Ridge IL.

Mulder, F. (1992) *Identiteit van informatica-onderwijs.* Inauguration. Open universiteit, Heerlen [in Dutch].

Mulder, F., Hacquebard, A.E.N., Smeets, D. and Veenstra-Strijland, T. (1992) Database technology for curriculum comparison as applied to informatics education, in *European Conference on Educational Research. Book of summaries, Vol. 1* (eds. Tj. Plomp, J.M. Pieters and A. Feteris), University of Twente, Enschede, 165-167.

8 BIOGRAPHY

Fred Mulder is working at the Dutch Open University from its start in 1983 and is full professor in informatics education since 1991. From 1993 till 1996 he was dean of the Faculty of Engineering. He holds degrees in chemical engineering (Bachelor), applied mathematics (Engineer) and theoretical chemistry (Ph.D.). After a postdoc research project in Canada, he went to teach informatics and mathematics in higher professional education, prior to his OU career. He has served on various national committees, such as the quality audit committees for informatics programmes at universities as well as higher professional institutes and committees for informatics at secondary schools. He is representing The Netherlands in the education committee TC3 of IFIP.

Anneke E.N. Hacquebard is a consultant for informatics and education since 1986. Her main fields of interest and professional activities are informatics curricula in higher professional education, the introduction of informatics curricula in secondary schools at a national level and the introduction of the European

Computer Driving Licence in the Netherlands. She chairs the education department of the Dutch Computer Society. From 1970 till 1986 she has taught mathematics, operational research and informatics and was a staff member at an institute for higher professional education. She holds a degree in mathematics from Leiden University and a qualification for teaching informatics at the higher professional level.