What are Information Technology’s Key Qualifications?

Christina Dörge
Department of Computer Science
Universität Bremen
Bibliothekstr. 1
D-28359 Bremen
Germany
monika@informatik.uni-bremen.de

Carsten Schulte
Department of Computer Science
Freie Universität Berlin
Takustr. 9
D-14195 Berlin
Germany
schulte@inf.fu-berlin.de

ABSTRACT
What kind of IT key qualifications do we have to teach in order to lessen the digital divide between educated and uneducated IT users, and to enable students who live in an IT-rich world to gain insights into the underlying science?

In this paper we argue that key qualifications, IT skills, and IT or media literacy can not be reduced to the teaching of applicational aspects, that is, to the functional characteristics of the IT systems in question. Instead, IT key qualifications consist in knowledge about the *concepts* of computer science.

Applying the concept of educational lenses (described in previous papers) to a course on IT key qualifications, we are able to obtain a more focused approach towards a definition of IT skills.

Categories and Subject Descriptors
K3.2 [**Computers & Education**]: Computer and Information Science Education – *computer science education, information systems education*.

General Terms
Experimentation, Human Factors.

Keywords
Wider Access, Gender, Computers and Society, CS Ed Research, Pedagogy, Key Qualifications, IT Key Qualifications, Educational Lenses, Curriculum Issues

1. INTRODUCTION
Educational institutions worldwide have begun to react to the new demand in technological skills, and new curricula have been created. Many technology communities and associations, such as the ACM or governmental institutions, have started to work on proposals how to create a general technical education: among others, there are the Computer Science Teacher Association (CSTA) in the US, the UNESCO and the Bundesregierung Deutschland.

ITiCSE'08, June 30–July 2, 2008, Madrid, Spain.

Creating sustainable curricula is a daunting task, as information technology is rapidly changing.

In recent years, as a multitude of papers on ICT testify, the technology surrounding us has undergone many changes. One key aspect of this is technology’s continuing advance, another is the ongoing trend of miniaturization. Technology can be found virtually everywhere in our lives and our world. It is sometimes easily visible, as in PC-equipped workplaces, and sometimes invisible, as in the processing unit of one’s toaster. HUBWIESER specifies four categories to classify the situation, “information flood”, “the job market”, “commercial relevance” and “ubiquity”. Using this classification, it is nearly impossible to say about any human being that computer technology isn’t something he or she has had to deal with in one way or the other (see [7], p. 58f).

The changes in the technological landscape have not just happened recently. In 1985, writing a book about general education, the educationalist KLAFKI was convinced that we would need a stronger general education in technology in order to enable people to adapt to the ongoing changes ([8], p. 60).

From a scientific perspective there are complaints that general education matters concerning information technology are not as easy to develop as they have been for other subjects (see [9], p. 1f). MAGENHEIM states that this is because of a lack of adequate models for information technology, since IT is a rather young discipline compared, for example, to mathematics or chemistry. He also says that with IT, additional factors must be taken into account. Computers have their own special set of conditions of operation and demand their own skills ([9], p. 2).

In the Bologna-Process it was decided that it is important for all European Universities to create a system which will enable students to obtain a certain degree of quality in their education and which will make it possible to compare and to switch between Universities within the EU, one criterion for this being the credit point system “ECTS” and the Bachelor / Master Degree, another being key qualifications. These key qualifications will have to have a measurable standard of quality and will therefore have to be testable. Key qualifications in the Bologna-Process are described on an informal basis, and not in their special function as IT key qualifications.

The aim of this paper is to show a concept of teaching information technology – which has already been done in courses at the University of Bremen – not only to convey key qualifications, but IT key qualifications. The concept of „Educational Lenses“ will be used to show how a new perspective on course concepts can be gathered for a new basis for pedagogical discussions.

2. Key Qualifications vs. IT Key Qualifications

In order to explain the concept of IT Key Qualifications, we first must state what key qualifications are.

The term „key qualifications“ was first used in 1974 by MERTENS, the former head of the „Nürnberger Institut für Arbeitsmarkt- und Berufsforschung” (= institute for job market and occupational research). He used it to postulate that what has to be taught to meet the demands of the workplace (e.g. capacity for teamwork) must be more than the qualifications normally taught for a certain profession (e.g. being skilled with certain tools).

Over time, many different key qualifications were identified. In DIDI et al. from 1993 were identified more than 600 terms describing key qualifications in education science.

Although it seem that key skills are something very broad, by using the term “key qualifications”, all authors associate the same idea: An ability to adapt to the rapid changes on the job market – knowledge extending and exceeding the skills needed to “just do the job”.

Therefore key skills are usually subsumed under some moiré abstract categories. MERTENS e.g. divided his key qualifications into four categories: Basic qualifications, horizontal qualifications, ubiquitous elements, and vintage-factors ([11], p. 36). ARCHAN and TUSCHEK, from the same scientific field as KLAFKI, have divided key qualifications into three categories: Professional competence, social competence and self-competence ([1], p. 5).

The authors of DIDI et al. used an empirical approach. They surveyed the over 600 terms and made a hit list with those most often used. Thus they found the following categories: the ability to communicate, the ability to cooperate, flexibility, creativity, associative thinking, autonomy, the capacity to solve problems, transferability, willingness to learn and the ability to assert oneself ([12], p. 2,, citing [2]).

This ‘hit list’ implicitly demonstrates one important aspect of key qualifications:

Talking about key qualifications means to prepare the learner / the learning generation for a continuously changing world: The days when it was possible to learn within a couple of years everything there can be known about a profession are long gone. “Life long learning” has become the buzzword of our current time ([1], p.3; [6], p.6; [10], p. 36 and p. 41f.) But key qualification also means to enable somebody to react to changes, to make it possible to decide whether a technology is dangerous or what to do with new technical developments: Nearly every day we are confronted with new technology and, accordingly, new decisions must be made. In addition there are certain implications connected with the usage of technology: Cell phones are traceable, RFID-chips in our clothes provide information on the price and the shop from which we have bought them, home banking via internet can be eavesdropped on by others, emails may contain dangerous attachments.

There may be few other fields where so many rumours are alive as in the field of computer technology. Each day we come across new information on TV or in the newspapers, telling stories of viruses, digital attacks, how technology is interfering with our private lives, new law acts and so on. Is greater clarity achievable by teaching key qualifications or do we need more than that – such as special key qualifications for IT?

Obviously, teaching *key qualifications* with the aid of computer technology seems possible. At the University of Bremen, several courses with the aim of teaching key qualifications were created and taught. In these courses the students had to work out a criteria catalogue to describe the functionality of a software they wanted to design for a certain purpose. They also had to check and evaluate several existing software programs against the criteria catalogue. They learned a number of basic facts about computer technology and used UML diagrams to explain what different programs are doing. The course concept mainly rests on the idea not to teach students to adapt their behaviour to particular pieces of software, but instead to get them to think about the conditions the software must satisfy in order to successfully help them with *their* tasks.

The key qualifications taught can be described using the above mentioned hit list:

- **The ability to communicate:** While creating criteria catalogues, participants learned to communicate their expectations about software and software design problems.
- **Ability to cooperate:** Some tasks could only be solved by working in teams.
- **Flexibility:** Many different programs and tools had to be used.
- **Creativity:** Designing the criteria catalogue and using the UML tool to create a readable diagram.
- **Associative thinking:** Structures of programs had to be understood by association
- **Autonomy:** Some problems had to be solved alone.
- **Capacity to solve problems:** The instructors didn’t help with every question. The students had to manage some problems by themselves.
- **Transferability:** Former experiences with computers had to be transferred to the new programs.
- **Willingness to learn:** Willingness to use an unknown system and programs (because LINUX was used as an operating system).
- **Ability to assert oneself:** During teamwork every team member had to state their opinion and discuss it with the others.

Further details can be found in these articles: [4] and [5].

But are the above key qualifications all that is to be understood when talking about IT key qualifications? Or are IT key qualifications more than this?

3. Teaching of IT Key Qualifications

In the above mentioned example - courses given at the University of Bremen - the aim was to teach IT key qualifications.

The idea of using all the above modules together was to convey the following concepts:

1. Self-reliant searching of ways to solve problems by using digital media
2. Self-training in new versions of a specific software or other functionally similar software
3. Ability to use PCs independently of their operating systems

4. Ability to autonomously keep up with the latest technological developments
5. Ability to recognize potential uses of IT for one's own needs
6. Ability to recognize dangers and risks for computer systems
7. Ability to use new trends and developments
8. Ability to assess socio-technical aspects of IT (potentials and threats)

(See [3], p.5)

The concept focuses on the kind of images (about IT) we must develop and teach to provide mental structures to which further knowledge can be successfully added later, *independently* of the ways in which that knowledge may be gained: Students must be enabled to add new information offered by other courses just as well as knowledge obtained in self-study. The course concept for IT key qualifications should include theoretical and practical parts. In addition, all courses should cover three major subjects (software, hardware, and additional aspects like algorithms and programming languages) in a way that will make it possible for the learner to see the connections between concepts. For example, software can be divided into groups, such as office software, tools, and operating systems. These groups differ in their characteristics, in the size of their source-code, the programming language used, or the operating system, which has to be used with a program. On the side of the hardware we have varying dimensions and different kinds of use and handling. Additionally, when we want to compare or discuss these matters we must go into greater detail: What is an algorithm, what is programming language? So it may be said that IT key qualifications must be more then mere key qualifications – „IT key qualifications are key qualifications expanded by specific IT-related aspects, defined by computer technology".

Therefore the main contents of the courses are:

- Basics of software
    - File systems and file formats (data compression, image formats)
    - Operating systems (differences and architecture)
- Basics of hardware
- Hardware: theoretical introduction to the components of a PC
    - Do-it-yourself: an old PC was used to let the students examine its components and apply the screwdriver themselves
- Internet applications and technologies
    - Architecture of the Internet (Client-Server-Principles, ideas of protocols and standards)
    - Security and protection (e.g. firewalls, virus scanners, malware)
    - Research in the Internet (catalogues, search engines, databases and additional resources)
    - Social Software (current trends of communities and the included technology)
- Additional aspects
    - Databases (basics about what a database is and what can be done with it, principles of relational databases)
    - From HTML to word processing (Principles of formatting texts and mark-up languages)
    - Open-Source, free software, license models, and patents: Digital-Rights-Management

(See also [3], p.7)

The results of a questionnaire after one of the courses was encouraging: 95% of the participants stated that they would now less likely shy away from trying to use an unfamiliar computer – even if it ran an operating system different from the one they were used to. This sheds some light on a possible way to find a good course concept for teaching IT key qualifications (see [5]).

But is the additional IT content taught enough to transform key qualifications into IT key qualifications? How do we know that the course successfully prepares students to cope with change in IT content (e.g. new concepts in file systems and operating systems)? Although students now are more confident to be able to do so (which in itself is important), there is no categorization system, which can be used to check whether the content taught was sufficient or if gaps still exist. In addition, future courses will obviously have to change the contents' specific details due to changes in hard- and software systems. How do we ensure that these changes do not diminish the positive effects of the course concept?

4. The Concept of Educational Lenses

The previous section showed that IT key skills have to be seen as key skills plus IT content. Therefore it is very difficult to define sustainable IT key skills, as the IT content is changing so rapidly.

In [10] the concept of educational lenses was presented. It is meant to provide 'didactic filters' to ensure that by teaching IT systems students learn important basics of CS, and about societal/ethical issues connected with CS. In [13] this concept is refined for use in literacy-oriented courses, and with regard to the idea of the dual nature of digital artifacts. The dual nature of IT helps to analyze the problem of defining IT key concepts as consisting of key concepts plus IT content: Usually, key concepts focus on function, whereas content focuses on structure. In other words, in order to understand IT, one needs to be educated in structural as well as in functional aspects, and additionally, one needs to develop an understanding of how these two dimensions relate to each other.

Therefore, educational lenses can be used to develop a course model of how to teach the concepts. The main point here is to explain the concepts at work (=in context) instead of the use of a certain program (see [4] and [5]).

The lenses are shown in the following table:

Educational Lens	Structure (the 'technical' aspect as it would be called by a traditional Engineer)	Function (the 'social' aspect as it would be called and (typically) considered to be unimportant by an engineer)
Automation	Which data structures allow for which kinds of operations? What is the structure (algorithms) of the explicit and implicit operation(s)?	What real-world process has been automated, and for which purpose?

Interaction	What is the structure of the explicit operations?	What is the effect/purpose of the interaction (GUI/use) metaphor? Which strategies are used?
Information processing	What is the structure of the implicit operations and data visualization?	What is the effect/purpose of these implicit operations and data visualization?
Networking	Are there any networking/cooperation facilities? If so, how do they operate?	How can people cooperate using the artifact?
Norm, regulation and law	Which structural aspects (operations, data structure) are due to regulation, and/or lead to regulation?	What is the purpose or rationale of certain regulations? What is the effect of these regulations?
Societal and ethical aspects	Analyze the structural development path of the artifacts: Which structures are fundamental/important?	Analyze the functional development path: Which functions are fundamental/important?

Table 1: Eduactional Lenses (see [13] for more info)

We use the example of a word processing system to concretize these ideas:

The function of a word processor seems trivial: It automates the mechanics of writing. Instead of laboriously producing marks (characters) on a sheet of paper with a feather or a pencil, the word processor – like a typewriter – produces these markings; and in addition to a typewriter with its fixed fonts we can easily change the font of our text from e.g. *Arial* to *Times New Roman*. Of course, word processors have many useful additional features like spell checking, support for the inclusion of tables, etc. – but it is the improvement of computer-based word processing over what we used to call 'mechanics of writing', which is our central focus here.

We must now analyze the structural aspects of this improvement: How is it that it is now so easy to change fonts and the layout of the text? Word processors have changed the nature of the written language. For centuries, written language was defined as (more or less) permanent markings on a material. Materials like stones, parchment, papyrus, paper, or the sand on a beach can be used to be marked with written characters. If the marking is changed, the written text is changed, too. Therefore a character was a unity of its layout (font) and its meaning as a member of the alphabet of its language. With the introduction of computers, characters lost this unity – they are now separated in fonts and codes. When the user types in a text into a modern word processor, patterns of code are stored and immediately visually presented in the currently chosen font. This produces the illusion that typing a text into a word processor can be compared to typing with a typewriter.

It is important to note that this *structural* feature (the representation of characters using codes and fonts) is the foundation for many features of word processors. It is the reason why a word processor can assign a different font to a given text without destroying it – the codes of the characters stay the same.

Word processors emulate the mechanics of producing markings on a material. While this function by itself may not seem to be a very interesting aspect, the structure behind it immediately leads to several additional functions: Regardless of the chosen font, a computer can search for patterns of codes. When a finished text is finally printed on paper, we can, without additional effort, save a 'copy' of that printout to our hard drive (that is, we save a pattern of codes and some additional information from which the word processor will be able to reproduce more copies the next time we use it). It is easy to change the text, to e.g. swap, add, and delete sentences, because any changes we make will be changes of code patterns. After the editing process, the new patterns of code may be arranged using the chosen font.

The seemingly unremarkable feature – the automation of the mechanics of writing – has important societal and ethical implications.

Here are some of the fields affected:

1. Production and reproduction of texts (copying and plagiarism)
2. Dissemination of texts (email, www)
3. Use and usefulness of texts (searching for and in documents; dissemination of texts as electronic books, and their different value compared to paper-based books)
4. General understanding of the concept of 'text'. There is a noticeable change in the previously fundamental defining feature of written language as some fixed (permanent) visual pattern on a material. This feature used to have the consequence that written language was, as the writing process itself, sequential. Now it is possible (or at least easier) to write iteratively. Consequently, texts in the www are often much less sequential than we are used to from book pages.

In summary, teaching IT key skills for word processors has to include basic or general technical (=structural) aspects. Doing this can help learners to better understand the structural complexities of software products emulating typewriters, enabling them to anticipate and search for yet undiscovered functionalities (functions). On a larger scale, they may learn to exploit the structure of IT *systems* using available functions, and to grasp at least some of the structural aspects to be able to adapt the systems to their own needs. Even in word processors the more advanced functions rely on their users writing texts with structural aspects of the software in mind. E.g. in order to use automatically generated content tables, headings have to be typeset using predefined style-sheets (heading1, heading2…).

In this example, we analyzed the subject of automation and which societal impacts are connected with it. We briefly hinted at aspects which come into focus when using different lenses of analysis: Interaction is based on WYSIWIG; some aspects of content – layout and document structure – are types of information processing; comments and change tracking provide networking facilities; norms are the codes used internally by the software (like Unicode), as well as the program's data structures and its patterns of use.

Educational lenses highlight certain basic structural (=technological) aspects in context with their functions (their impact on software use, on provided functionalities, and/or on societal consequences). From the perspective of IT skills (in the sense of non-conceptual skills), educational lenses show how to add IT content to the learning process.

5. Conclusion

Putting the course concepts from the first part of the article together with educational lenses, the following categories within the course structure can be identified:

a) **Automation:** Historical development of computer technology – to provide a first insight into the history of computers. Mechanization of mental labour – for what reasons and purposes were computers invented?

b) **Interaction:** Concepts and structures of application software. In which ways can computers interact with human beings? What kind of behaviour can users expect from computers?

c) **Information processing:** What are computers capable of doing? (E.g. sorting information, fast processing, easy ways to distribute digital products. Use of databases to store huge amounts of data.)

d) **Networking:** New methods of working and social platforms through Wikis and Blogs, client-server architectures, potentials and dangers of the Internet: Surfing, home banking and emails.

e) **Norms, regulations and law:** What kind of rules does a machine have to follow? What can it do, what is impossible? Learning about these aspects by using HTML.

f) **Social and ethical aspects:** Vanishing or alteration of professions, information flood, faster data connections and information gathering, questions about copyright and law concerning the distribution of documents and music (plagiarism).
What do teachers have to know about computer technology? What must be done to produce responsible citizens?

Educational lenses offer a way to group the potential contents of computer technology courses. By providing analytical categories, they allow a different look at these contents and therefore a new way to discuss IT key qualification course structures. It is important that IT key qualifications will be the qualifications students take home from a computer technology course.

6.REFERENCES

[1] Archan, S. und Tuschek, E.: „Schlüsselqualikationen – Wie vermittle ich sie Lehrlingen?“, Österreichisches Institut für Bildungsforschung der Wirtschaft, Wien, 2002 http://www.ausbildungsjournal.at/art/arc_048_02_schlusselqualikationen.pdf [02. Jan 2007].

[2] Didi, H.-J., Fray, E., Kloft, C. und Vogt, H.: „Einschätzungen von Schlüsselqualifikationen aus psychologischer Perspektive“, Zusammenfassung eines Gutachtens im Auftrag des Bundesinstituts für Berufsbildung, Bundesinstitut für Bildungsforschung, Bonn, 1993

[3] Dörge, C. und Büschenfeldt, M.: „Vermittlung von IT-Schlüsselkompetenzen für den nachhaltigen Umgang mit Digitalen Medien“, in ‘GdW-Praxishilfen’, Luchterhand Verlag, Köln, S. 1_14. Ergänzungslieferung 66, Dezember 2006, Gruppe 9:9.20.20.32.

[4] Dörge, Christina: Einsatz von Open-Source-Software zur Vermittlung von ITSchlüsselkompetenzen, Open-Source-Jahrbuch 2007 - “Zwischen Softwareentwicklung und Gesellschaftsmodel”, Berlin, 2007

[5] Dörge, Christina: "IT Key Qualifications For Students In Education". SITE Full Paper Acceptance, San Antonio, Texas, USA, 2007 (Conference Proceedings)

[6] Gonon, P.: „Und am Schluss auch noch Latein?“, Panorama (6), S. 6 ff, 2002 http://www.infopartner.ch/periodika/2002/Panorama/Heft_6_2002/pan2606.pdf [02. Jan 2007].

[7] Hubwieser, P.:“Didaktik der Informatik“, Springer Verlag, 2000

[8] Klafki, W.: „Neue Studien zur Bildungstheorie und Didaktik“, Beltz Verlag, Weinheim, Basel, 1985

[9] Magenheim, J: „Towards a Competence Model for Educational Standards of Informatics“, In: WCCE 2005 - Proceedings of the 8th IFIP World Conference on Computers in Education, University of Stellenbosch, Cape Town (SA), 4-7. Juli 2005.

[10] Magenheim, J. und Schulte, C.: Social, ethical and technical issues in informatics—An integrated approach In: Education and Information Technologies, Springer Netherlands, Vol 11, Numbers 3-4 / Oktober 2006, S. 319 - 339

[11] Mertens, D.: „Schlüsselqualifikationen. Thesen zur Schulung für eine moderne Gesellschaft“, Mitteilungen aus der Arbeitsmarkt- und Berufsforschung 7, S. 36_43., 1974 http://doku.iab.de/mittab/1974/1974_1_MittAB_Mertens.pdf [02. Jan 2007].

[12] Orth, H.:„Schlüsselqualifikationen an deutschen Hochschulen, Luchterhand, Neuwied, 1999

[13] Schulte, Carsten: „Duality Reconstruction - Teaching Digital Artifacts from a Socio-Technical Perspective“ In: 3rd International Conference ISSEP Informatics in Secondary Schools Evolution and Perspectives. Informatics Education – Contributing Across the Curriculum, 1-4 July, 2008; Toruń, Poland