Computer Science/Informatics in Secondary Education

Peter Hubwieser
Technische Universität München
Fakultät für Informatik
Boltzmannstr. 3, D-85748 Garching
+49 (89) 289 17350
peter.hubwieser@tum.de

Michal Armoni
Department of Science Teaching
Weizmann Institute of Science
76100 Rehovot, Israel
+972-8-934-4037
michal.armoni@weizmann.ac.il

Torsten Brinda
Didactics of Informatics
University of Erlangen-Nuremberg
Martensstr. 3, D-91058 Erlangen
+49 9131 85-2762
brinda@cs.fau.de

Valentina Dagiene
Vilnius University
Fac. of Mathematics and Informatics
Naugarduko 24, LT-03223 Vilnius
Lithuania, +370 698 05448
valentina.dagiene@mii.vu.lt

Ira Diethelm
Carl von Ossietzky Universität
Fakultät II - Department für Informatik
26111 Oldenburg
+49(0)441-798-2990
ira.diethelm@uni-oldenburg.de

Michail N. Giannakos
Ionian University
Department of Informatics
Platia Tsirigoti 7, GR-49100 Corfu
+30 6986309509
mgiannak@ionio.gr

Maria Knobelsdorf
University of Potsdam
August-Bebel-Str. 89
14482 Potsdam, Germany
+49-331-977-3104
knobelsdorf@cs.uni-potsdam.de

Johannes Magenheim
University of Paderborn, Faculty EIM
Institute of Computer Science
Fürstenallee 11, D-33102 Paderborn
+495251606341
jsm@uni-paderborn.de

Roland Mittermeir
Alpen-Adria Universität Klagenfurt
Universitätsstr. 65-67
A-9020 Klagenfurt, Austria
+43 463 2700-3513
roland@isys.uni-klu.ac.at

Sigrid Schubert
Universität Siegen
Institut f. Didaktik der Informatik
D-57068 Siegen, Germany
+49 (271) 740 3457
sigrid.schubert@uni-siegen.del

ABSTRACT
Computer Science (CS) Education research, specifically when focusing on secondary education, faces the difficulty of regionally differing political, legal, or curricular constraints. To date, many different studies exist that document the specific regional situations of teaching CS in secondary schools. This ITiCSE working group report documents the process of collecting, evaluating, and integrating research findings about CS in secondary schools from different countries. As an outcome, it presents a category system (Darmstadt Model), as a first step towards a framework that supports future research activities in this field and that supports the transfer of results between researchers and teachers in CS education (CSE) across regional or national boundaries. Exemplary application of the Darmstadt model shows in several important categories how different the situation of CSE in secondary education in various countries can be. The Darmstadt Model (DM) is now ready for discussion and suggestions for improvement by the CSE-community.

Categories and Subject Descriptors
K.3.2 [**Computer and Information Science Education**]: Computer science education, Curriculum.

General Terms
Human Factors, Legal Aspects.

Keywords
Secondary Education, Informatics, CS as s Subject.

1. INTRODUCTION
The benefit of research in the field of CS Education highly depends on its potential of being applicable to teaching. As concrete teaching is always very specific due to various circumstances like the educational goals of a country, the personal preconditions of the learners or the technical equipment, these circumstances also determine the applicability of any research result. In tertiary education (at least in its first years), the learning context might still be comparable enough. For example CS1 courses in a Bachelor of Science program, respecting the differences between courses for "majors" or "non-majors", very often have quite comparable learning conditions. In the field of secondary education however, the situation is very different: there might substantial differences between countries, between the states of a country, between different school types or directions of study. These differences may affect e.g. the personal properties of the students, the educational background of the teachers, curricula, standards or the technical equipment of the schools. Therefore, research results concerning education generally remain quite specific. Apparently, compared

to higher education, education in school is much more dependent on contextual circumstances such as political aspirations, the level of socio-economic development, the overall goals or the design principles of the educational system in the respective country or state.

As for any other subject, the only place to offer computer science education to all students of an age group is in school. Additionally, school education forms the basis university programs can build upon. Therefore school education in computer science is crucial as well for the basic IT-competencies people need in their everyday life as for the prerequisite knowledge of university students. Assuming that the emphasis of CS or Informatics education in school takes place after the primary/elementary stage, our working group at *ITiCSE 2011* aimed to develop an initial conceptual framework for CS/Informatics in secondary education. This framework should support the applicability and the transfer of research results as well as the transparency and objectiveness of the discussion about the teaching of computer science in school.

In order to achieve this purpose we decided to collect, evaluate and integrate case studies and research findings about Informatics in secondary school. After the evaluation of several extensive case studies we have developed a category system (called *Darmstadt Model,* shortly DM) that might be regarded as a first component of a future research framework. Additionally, the most important categories were applied to the initial case studies in order to compare the situation of CS in secondary education in several countries.

Additionally, the working group decided to establish a long term cooperation that should aim to a collection and stimulation of research activities for CS in secondary education as well as to a closer investigation of the situation in this field all over the world.

In this paper, we report the process of developing the category system as well as some examples of its application. It commences by clarifying the terminology used. In section 3 the problem of diversity of educational systems is discussed. Then, in section 4 the goals of the working group are presented. In section 5 the theoretical background that we used is introduced. Section 6 discusses related work before section 7 explains the methodology used for developing the category system presented in section 8. Several important categories are illustrated in section 9. In section 10 the proposed category system is discussed. Section 11 gives an outline of future work and possible applications.

We put the category system for discussion in the community, hoping that these initial results have an impact on future research activities in the field and that we can contribute to a better transfer between research and teaching in secondary CS education in different countries.

2. TERMINOLOGY

2.1 Informatics, Computer Science, ICT

While the term *Computer Science* (CS) is used in a very similar way internationally, the Term *Informatics* (respectively the German *Informatik* or the French *Informatique*) is understood differently depending on the country or the social or cultural background. In order to discover its specific meaning, we have to go back to the 60ies. Apparently some European countries had problems accepting the term *Computer Science* for the new scientific discipline that was arising around electronic data processing and its theoretical foundations [14]:

– “While computer engineering and information science are nearly self-explanatory, the term computer science is somewhat puzzling. Is hardware, the computer, in the focus of that science? What then is the difference to computer engineering? And if information processing is in its focus: What is the difference to information science? [..] It seems that Philippe Dreyfus introduced the French name informatique from the elements information and automatique or électronique in 1962. In Germany, Standard Elektrik Lorenz used the word Informatik already in the late fifties naming a manufacturing site (Informatik-Werk), but this name and trademark was dropped later. In France the notion informatique was used throughout the French press as a generic name related to computers and automation. The Académie Française defined informatique officially in 1967: ‘Science du traitment rationnel, notamment par machines automatiques, de l'information considéré comme le support des connaissances humaines et des communications dans les domaines technique, économique et social.’"

The German universities use *Informatik* and CS usually quite synonymous, which is proven by many faculty names (*Fakultät für Informatik* vs. *Faculty of Computer Science*, e.g. at Universität Bonn, Universität Magdeburg, TU Dresden) and by the translations of some websites of universities (e.g. ETH Zürich). On the other hand, some other universities in the German speaking part of Europe translate *Informatik* to *Informatics* (e.g. TU Vienna, Karlsruhe Institute of Technology).

Concerning the differentiation between CS/*Informatics* and *Information* and *Communication Technology* (ICT), the UNESCO/IFIP Curriculum 2000 defines [70]:

– ”*Informatics* (Computing Science): The science dealing with the design, realisation, evaluation, use and maintenance of information processing systems; including hardware, software, organizational and human aspects, and the industrial, commercial, governmental and political implications (UNESCO / IBI).
– *Informatics Technology*: The technological applications (artifacts) of informatics in society.
– *Information and Communication Technology* (ICT): The combination of informatics technology with other, related technologies, specifically communication technology.”

Nevertheless, in the same publication, “these definitions have been collapsed into one, all encompassing, definition of Information and Communication Technology (ICT). This implies that ICT will be used, applied and integrated in activities of working and learning on the basis of conceptual understanding and methods of informatics” [70]. Consequently, the term ICT has to be considered very carefully in the context of education. Its meaning may vary from “teaching basic concepts” to the pure application of systems.

As this is not the proper platform to discuss the subtle and quite varying differences between *Computer Science, Informatics* and *ICT*, it seems prudent to use the terms synonymously in this paper, as far as we restrict the discussion to the context of education.

2.2 Secondary Education

Suggested by the variety of educational systems, it seems necessary to define the educational field this working group focuses on. In this paper *Secondary Education* (SE) means the part of the educational process that follows the first stage of elementary or primary education. The goal of SE is general education, in contrary to vocational education, which aims at the acquiring of knowledge and competencies for a specific profession. A certain part of the SE system usually qualifies for the enrollment at universities. The SE process ends with the beginning of tertiary, education. The age of students in the SE stage might be different, depending on country or state. Nevertheless, the maximum range is from age 10 up to 19 years. The spectrum of grades and age groups pertaining to SE are detailed in the next section.

Several countries additionally differentiate between *lower* and *higher secondary education* in the sense that the first represents a compulsory part of school following immediately after primary school, while the second leads to some exam that grants access to higher education, e.g., to university enrollment. This second part might be optional.

3. THE PROBLEM OF DIVERSITY

There is a variety of very different approaches towards teaching Informatics in secondary schools. They differ heavily concerning organizational circumstances (e.g. within a mandatory vs. an elective subject), learning goals, topics (e.g. applied programming paradigms, languages), or teaching methods. Concerning the organizational circumstances, a short glance at the educational systems of the five countries (respectively states) that were represented in the working group reveals enormous differences (see Fig. 1).

Primary education lasts from 4 years in Bavaria, Lithuania and Austria, over 6 years in Greece to 6-8 years in Israel. Lower SE takes place in a single school type in Greece or is split into 4 types in Austria.

Within countries with a federal system there might be huge differences even between the single states. In the US level 1 of the recent standards of the CSTA were implemented from 3% by D.C. to up to 100% e.g. by Iowa or Massachusetts [73]. In Germany some states have a compulsory subject of Informatics at least for the majority of lower secondary schools (e.g. Bavaria or Saxony), while some others don't offer any regular Informatics courses (e.g. Bremen or Lower-Saxony) [61].

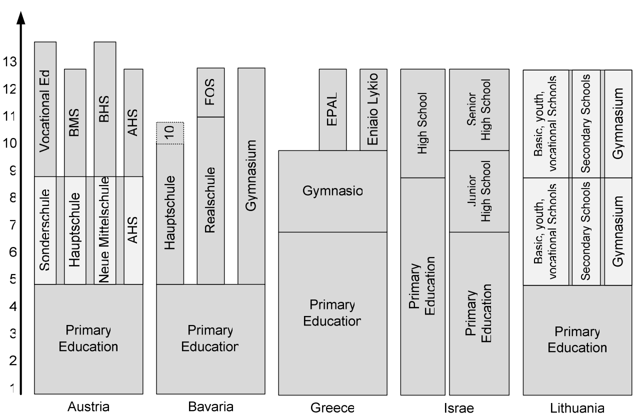

Figure 1. Educational Systems of the represented countries

Figure 1 clearly shows that even in the moderate sample of five countries "secondary school" encompasses very different grades (in the extremes it's 5 to 13 versus 9 to 12). Thus, apparently, any research on SE has to specify at least the grade level respectively the age group of the cohort studies in order to be properly categorized in a specific national school system.

4. GOALS OF THE WORKING GROUP

In the face of the diversity described above it seems at least doubtful, if a research result that was produced e.g., in the 6th grade of a *Gymnasium* in Bavaria, attended by the "best" 30% of the students of an age group, could be reproduced in the same grade in a Greek *Gymnasio*, where all students of the age group are united.

Therefore the working group aimed to produce a conceptual framework that should describe at least the most relevant aspects of CS in SE. Hereby it should help to

– compare and systemize research results,
– assess the possibility of transfer of the results to another country, and also
– stimulate new research projects by detecting new relevant research questions.

In their frequently cited definition, Miles and Huberman [46] stated that a *conceptual framework* "explains, either graphically or in narrative form, the main things to be studied—the key factors, concepts, or variables—and the presumed relationships among them" (p. 18).

Maxwell specified more precisely [42]: "The second major source of modules for your conceptual framework is prior theory and research [..] As LeCompte and Preissle (1993, p. 239) stated, 'theorizing is simply the cognitive process of discovering or manipulating abstract categories and the relationships among these categories.' My only modification of this is to include not simply abstract categories, but concrete and specific concepts as well" (p 42, the cited reference is [37]).

In order to reach its goal, the group therefore decided to develop a suitable *category system* by coding and categorizing appropriate publications as a first step. In detail, the research results could be compared by using the values of the categories as metadata, describing the differences as well as the similarities of the research context. For example one research project might have evaluated the effects of a certain teaching method in a classroom of the 10th grade at a US municipal high school with 30 predominantly male students, 64% of which had immigration background, while a second study with the same research goal might have been conducted at all Bavarian *Gymnasiums* with more than 45.000 students of the 6th grade, 51% female and 10% with immigration background. The proposed category system could serve as a multidimensional road map that helps to assess the "distance" of research results according to certain variables that might be relevant. In our example there are substantial differences regarding the school type, the number and age of the students that participated, the gender distribution and the proportion of students with immigration background. Thus the two studies might be regarded as very "distant" and therefore, the results of both studies will have to be interpreted quite differently.

On the other hand the metadata given by the category values might be used to suggest trends or correlations between the categories, e.g. between the age of the students and the suitability of a certain programming language or teaching concept. This might stimulate new research projects that could try to provide evidence for such trends or correlations. Additionally, researchers might have a look at the distribution of existing research over the categories. There might be some categories that were not covered yet by research at all, which might encourage them to pose new research questions. These effects could contribute to the evolution of a new *research framework* for CS in SE that might start with our category system.

In a further step this category system might be used to produce a questionnaire for a survey that investigates the implementation of CS in SE and the experiences of different countries

The long-term goal of the working group is to establish international cooperation in the field of SE, collecting and comparing research findings from as many different countries as possible about the effects and outcomes of different teaching approaches. This could be performed in the following steps:

- collect case studies and other research results from different countries,
- categorize and compare the content using our category system,
- conduct comparative studies about CSE in secondary schools,
- produce guidelines for installing a subject of CSE or improving CSE education in other ways,
- propose and conduct new collaborative research projects.

5. THEORETICAL BACKGROUND

As the working group focuses on the applicability of research results in the classroom respectively on the possibility of transfer to similar teaching situations, we decided to start with a theoretical framework that describes the context of classroom teaching as complete and as detailed as possible. The Berlin model seemed to meet these requirements sufficiently. Some members of the group had used this model extensively for teacher education, as discussed in [12].

5.1 The Berlin Model (BM)

In 1962 Paul Heimann proposed the *Berlin model*, described in English by Uljens [67]. It represents a theoretical framework for the preparation and planning of school lessons. This well-structured model has been successfully applied in several educational studies. The BM distinguishes between the *preconditions* of learning, several *decision areas,* and finally the *consequences* of learning measures. The preconditions as well as the consequences are sub-categorized as *anthropogenic* or *socio-cultural*. The four decision areas considered are *intentions* (e.g. learning objectives), *content (topics, knowledge)*, teaching and learning *methods,* and *media*. In detail the different categories may address the following aspects (as far as they are relevant in our context):

- *anthropogenic preconditions*: age and social level of students, gender aspects, prerequisite knowledge;
- *socio-cultural preconditions*: school system, legal preconditions, outcome definition by curricula or standards, ethnic and traditional aspects, technical and financial resources;
- *decision area of intentions*: learning goals, objectives, outcomes, competencies, standards;
- *decision area of content*: central topics, areas of subject domain knowledge;
- *decision area of teaching methods*: teaching approaches, typical learning and teaching methods;
- *decision area of media*: computers, internet, textbooks, unplugged media etc.;
- *socio-cultural consequences*: research findings about the large-scale changes that are caused by the subject, e.g. a changed attitude towards data protection or increased levels of user skills in the society, change in the choice of directions of study (e.g. more women in Informatics);
- *anthropogenic consequences*: learning outcomes of the students: which competencies or skills have been acquired?

5.2 Refinement of categories

In this section two very important categories (*Intentions* and *Content*) of these will be discussed and refined based on existing work.

5.2.1 *Intentions: Learning Objectives*

Concerning the structure of learning objectives (in the category *intentions*), we follow Anderson and Krathwohl [3] who regard learning objectives as a combination of a certain type of *knowledge* and an observable *behavior* (called cognitive process) concerning this type of knowledge, forming the two dimensions of their revision of Bloom's taxonomy:

- *knowledge dimension*, partitioned into a. factual, b. conceptual, c. procedural, and d. metacognitive knowledge,
- *cognitive process dimension*, partitioned into 1. remember, 2. understand, 3. apply, 4. analyze, 5. evaluate, and 6. create.

To describe the specificity of learning objectives, Anderson and Krathwohl [3] proposes three levels:

- *global* objectives: "complex, multifaceted learning outcomes that require substantial time and instruction to accomplish";
- *educational* objectives: "derived from global objectives by breaking them down into more focused, delimited form";
- *instructional* objectives, "focus teaching and testing on narrow, day-today slices of learning in fairly specific content areas".

5.2.2 *Intentions: Competencies*

Based on the well-known definition of a competence by Weinert [72], Klieme et al. [36] proposed a definition of competencies that covers cognitive abilities and skills enabling students to solve particular problems successfully and responsibly in various situations as well as the motivational, volitional, and social readiness and capacity of them. Weinert [72] suggests that an individual degree of competency is determined by various facets like: ability, knowledge, understanding, skills, action, experience, and motivation.

Currently, there is no consistent differentiation between the terms *competence* and *competency* [57]. The plural *competencies* is often used to stress that the constructs in question are complex and multifaceted. We will use the term *competence* in the sense of a more global ability where certain competencies represent components of it.

Dörge [18] compared the different background and use of the terms *Competency*, *Skills* and *Qualification* in the German and the English language area and found considerable differences.

With regards to interdisciplinary competencies, the European Union presented the "European qualification framework" (EQF) [1] that distinguishes 8 levels of competence, e.g. Level 3: "take responsibility for completion of tasks in work or study, adapt own behavior to circumstances in solving problems".

As a suitable framework for the development of subject specific competency models, the OECD has presented "The Definition and Selection of Key Competencies (DeSeCo)" [57].

Recently Magenheim et al. [39], Schubert and Stechert [58] published the first outcomes of their MoKoM-project that aims to develop an empirically-based competency models in the context of Informatics in school. They started with a theory-driven model that was enriched through empirical data. In addition to the objective of developing competencies, the MoKoM-project aims to develop "test instruments that are appropriate for competence measurement and design, and the evaluation of learning environments that have been proven to be of high quality through competence measurement" [58]. For a much more reduced subject area [8] presented an empirically founded competence model for object-interaction in introductory programming.

5.2.3 Intentions: Standards

Concerning standardization the subject of Informatics runs far behind the traditional subjects like Mathematics. The *Principles and Standards of the National Council of Teachers of Mathematics* [53], are the best-known and most influential example internationally. They describe framework conditions for instruction on all grade levels, from the beginning of primary education to the end of secondary schooling and provide guidelines for improving mathematics teaching by moving towards comprehension- and problem-based instruction. In particular the NTCM presents a definition of *problem solving* that might be transferred to CSE as well: "Problem solving means engaging in a task for which the solution method is not known in advance. In order to find a solution, students must draw on their knowledge, and through this process, they will often develop new mathematical understandings. Solving problems is not only a goal of learning mathematics, but also a major means of doing so" ([53], p. 52).

A very comprehensive discussion of educational standards was presented by Klieme et al. [36].

Some proposals for educational standards in Informatics came from Austria [19] and from the German *Gesellschaft für Informatik* (GI) [23].

Very recently the CSTA Standards Task Force presented its K-12 Computer Science Standards (Revised 2011) in a draft version [66]. These standards may be comprised by the subcategory *Standards* (of the category *Intentions* in the DM). It defines three levels for the learning outcomes, where the highest level is divided into three discrete "courses":

- level 1 (recommended for grades K–6) Computer Science and Me,
- level 2 (recommended for grades 6–9) Computer Science and Community,
- level 3 (recommended for grades 9–12) Applying concepts and creating real-world solutions,
  - level 3A: (recommended for grades 9 or 10) Computer Science in the Modern World,
  - level 3B: (recommended for grades 10 or 11) Computer Science Principles,
  - level 3C: (recommended for grades 11 or 12) Topics in Computer Science.

In order to avoid the perception that CSE should focus exclusively on programming, five complementary and essential strands throughout all three levels are distinguished:

- computational thinking;
- collaboration;
- computing practice;
- computers and communication devices;
- community, global and ethical impacts.

These strands are further illustrated by lists of competencies that represent the proposed standards. Additionally the draft paper also offers a variety of activities, assigned to the levels and strands, respectively that show in detail how classroom teaching might look like.

5.2.4 Content: Subject Domain Knowledge

Concerning the categorization of subject domain knowledge (category content), we propose to apply the well-known *ACM Computing Classification Scheme* from 1998 (www.acm.org/about/class/1998).

Concerning the educational value of knowledge elements in CSE, several taxonomies were presented. Schwill adapted the fundamental ideas identified by Bruner [13] to Informatics [59]:

"A fundamental idea with respect to some domain (e.g. a science or a branch) is a schema for thinking, acting, describing or explaining which

(1) is applicable or observable in multiple ways in different areas (of the domain) (horizontal criterion),
(2) may be demonstrated and taught on every intellectual level (vertical criterion),
(3) can be clearly observed in the historical development (of the domain),
(4) will be relevant in the longer term (criterion of time), and
(5) is related to everyday language and thinking (criterion of sense)."

6. RELATED WORK IN CSE

Recently Malmi et al. [40] characterized the current research activities in Computing Education. They developed a classification scheme, investigated the theoretical quality of the publications (Theory/Model/Framework/Instrument) as well as validity and reliability issues and gave the following definition: "A *research framework* is an overall orientation or approach that guides or describes the research, as opposed to a specific method or technique. A research framework may have associated theoretical, epistemological, and/or ontological assumptions (e.g. phenomenography), may prescribe or suggest the use of particular methods (e.g. grounded theory), or may simply be a descriptive term for a kind of research activity that has certain characteristics (e.g. action research, case study)". One of the long-term goals of our group is to develop such a framework for CS in SE. Additionally, we could plug-in the framework of Malmi to expand the category *Research* of the DM (see section 8.2).

Randolph [55] examined the current research practices in the field of computer science education, based on papers from the ITiCSE, the SIGCSE and the ICER conferences. He found that one third of articles did not report research on human participants" and "nearly 40% only provided anecdotal evidence for their claims". Additionally, he detected some typical research methods that depend upon the continent the first author came from. These methods might form subcategories of the category *Research* of the DM (see section 8.2).

Joy et al [34] investigated the interdisciplinary nature of Computer Science Education and reported a survey of 42 publications. They reviewed existing taxonomies for the general fields of Computer Science and Education, and a novel taxonomy. This was "specifically aimed to help new researchers in the field understand what types of papers are published and where they appear." They elaborated the following final categorization scheme: *System, Technology, Resources, Other technical, Theoretical pedagogy, Practical pedagogy, Curriculum, Social factors, Psychology factors, Other educational, Other.* Section 10 (table 13) shows a comparison of these categories with our model.

Hazzan et al. [27] have produced a very instructive research-based "Guide to Teaching Computing Science" that offers an extensive overall view over the aspects that have to be considered for teaching CS. Hazzan was also one of the authors of an overview of the application of qualitative research in CSE [26]. This book will be one of the next texts that will be coded using the DM in order to expand or detail its categories.

In 2007 Weeger [71] compared in his national synopsis the implementation of Informatics in the secondary schools of the 16 German states. German states have very different educational systems. In the same year Blumrich [10] collected some information about the international situation. It focused on organizational aspects. The only countries outside Europe that he investigated were Japan, New Zealand and the Philippines. Starruß [61] updated and completed the synopsis about the German situation. These synopses provide an instructional overview over the situation of CS in SE in Germany and other European countries. They are also to be coded using the DM.

The issue of the assignment of knowledge to the subsequent steps of the learning process was addressed by several curriculum proposals. In 1993, ACM produced [54]. In 1994 UNESCO offered its curriculum for Informatics in secondary schools [69]. It was renamed to "Information and Communication Technology in Secondary Education" in 2000 [70] and extended to "Information and Communication Technology in Education" in 2002 [68]. In 2003 the ACM K-12 Task Force Curriculum Committee published its "Model Curriculum for K-12 Computer Science" [65]. These curricula provide possible values for the category *Knowledge* of the DM.

In 2005 Dagiene and Mittermeir organized a new series of specific conferences that was called "Informatics in Secondary Schools – Evolution and Perspectives (ISSEP)". Until today, ISSEP took place 2005 in Klagenfurt (Austria) [51], 2006 in Vilnius (Lithuania) [17], 2008 in Torun (Poland) [49] and 2010 in Zurich (Switzerland) [29]. Although there are some contributions from other countries (like Israel, Japan or the US), the main emphasis of these conferences lies in central and eastern Europe. ISSEP 2011 took place in Bratislava (Slovakia) [35]. There were several publications describing national initiatives, e.g. by A. Tucker, who presented and explained the situation in the US and the ACM K12 [64] or the UK [24]. Frequently these initiatives come from eastern European states, e.g. as described by V. Dagiene in her papers about Informatics education in Lithuania [15, 16]. In the last years several more or less detailed discussions of computer science education in secondary schools have been written, e.g. from Austria [45], Poland [25], [63], New Zealand [7], Israel [5]. Regarding the US, the Computer Science Teachers Association of the ACM (CSTA) had produced a close description of the situation [62], which discussed also the comparable aspects in several other countries, e.g. Austria, Israel, and Poland. This was followed by the publication of Wilson et al. [73], which investigated the implementation of the ACM Curriculum in each of the US-states and demonstrated how differently the US states have implemented the ACM K12 curriculum. All these publications contain many very informative and extensive texts about CE in SE. These might be coded in the near future in order to evaluate the DM.

7. METHODOLOGY

7.1 Text corpus

In order to structure the work of the group, we used the Berlin model [67] as a theoretical framework. As many members of the group are acknowledged experts in the field, we decided to produce as many case studies as possible that could be used as a starting point. The case studies should follow the structure of the BM and should be completed well before the ITiCSE conference. This resulted in five case studies, all together covering 57 pages of text: *Austria* (by Roland Mittermeir), *Bavaria* (a federal state of Germany, by Peter Hubwieser), *Greece* (by Michail N. Giannakos), *Israel* (by Michal Armoni) and *Lithuania* (by Valentina Dagiene). Table 1 presents the size and the number of inhabitants of the five countries (respectively states) considered.

Table 1. The countries/states described by the case studies

Name	Type	Inhabitants	Size (km^2)
Lithuania[1]	Country	3.3 Mio	65.000
Israel[2]	Country	7.6 Mio	22,145
Austria[1]	Country	8.3 Mio	83.870
Greece[1]	Country	11.2 Mio	131.957
Bavaria[3]	State	12.5 Mio	70.551

[1] following europa.eu; [2] www.mfa.gov.il; [3] www.bayern.de

7.2 Development of the category system

The work had to start with a qualitative text analysis of the case studies. As there are several approaches for qualitative text analysis, the working group had to decide which methodology might be the most suitable for the intended purpose. After a detailed discussion the methodology of Philipp Mayring was chosen, who had combined several techniques for systematic text analysis [43] to a very systematic process. Following Mayring, the category system might be either derived from a suitable existing theory (deductive category application) or developed during the analysis from the text corpus (inductive category development). The first strategy incorporates also the revision of the existing category system. Both methods may also be combined.

As the case studies were written according the BM, it seemed obvious to follow the strategy of *deductive category application* [43] and start with a category system that was taken directly from the BM:

- *Preconditions*
    - *anthropogenic*
    - *socio-cultural*
- *Decision areas*
    - *Intentions*
    - *Content*
    - *Teaching and learning methods*
    - *Media*
- *Consequences*
    - *anthropogenic*
    - *socio-cultural*

The software *MaxQDA* (www.maxqda.com) should be used for all coding work and for the numerical evaluation of the coding results.

For the first coding step, the Bavarian case study was coded in a plenary session by all members of the working group in order to expand the category system and to standardize the personal coding habits. Hereby we discovered that several important categories were missing or not suitably located in the hierarchy of the BM in order to properly capture the content of the case studies. Thus several new categories were included, e.g. the categories *Preconditions/Teacher education* and *Curriculum development*.

The remaining four case studies were coded in groups by three members each, including the author of the case study, who should explain his text occasionally. After finishing this coding step, each group gave a report of its coding experiences and proposed changes for the category system. It turned out that every group found (partly) different new (sub-) categories and that there were serious problems with the hierarchy of the BM, particularly with the distinction of *Preconditions* and *Decision area*, which frequently depended on the professional position of the persons in the addressed target groups.

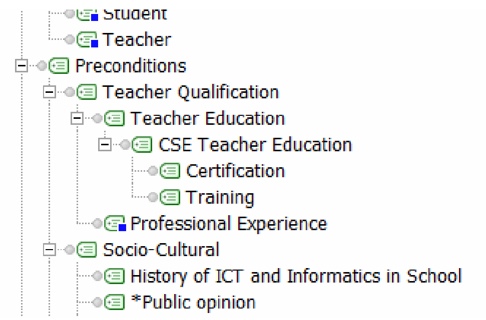

Figure 2. Part of the moderately revised BM category system.

At the end some more categories (e.g. *Policies*) had to be introduced, some others had to be moved to another place in the hierarchy, e.g. *Motivation* from a subcategory of *Methods* to the top-level, because it might also be a precondition in some cases. Most of the categories were refined by adding new subcategories. This resulted in a model that comprised 70 categories in a five-level-hierarchy system.

After this first round two more coding rounds were performed. These were done by pairs (not including the respective author), involving different persons regarding each document and round, which resulted in 1154 coded text fragments altogether.

7.3 Problems with the Berlin model

As mentioned above, the most serious problem with the BM was the distinction of *Preconditions* and *Decision area.* In the case of comparing studies from different countries, decision areas turned out to be quite specific to various professional levels or stakeholders. On the highest level, one might see national policymakers and ministries of Education. In some countries, regional policy makers and regional school administration might have to be considered as deciding authorities. Usually, schools (and their location in particular areas of the region, e.g., rural or urban) have a spectrum of decisions to make, and finally, teachers will, either formally granted or informally usurped, decide on how they conduct their way of teaching. In the papers considered, a kind of onion-like structure emerged where for each stakeholder the outer layers were "preconditions" while the layer controlled by this stakeholder would be the respective "decision area". Thus, depending on the author's text and/or the coder's perspective, different coding results emerged.

A further critical result was the low average percentage of the intercoder agreements of the coding iterations (see table 2). As our category system should be used by the community at the end of the development process, intercoder agreement is a crucial measure for the objectivity of coding results and hereby for the usability of the category system.

The agreement percentages were calculated automatically by MaxQDA with a threshold of 10% overlapping, which means that two codings are counted as equal if the coded text fragments overlap at least 10% of one of them.

Table 2. Intercoder agreement percentages

Comparison	1 -- 2	1 -- 3	2 -- 3	Average
Austria	12,00*	13,00*	48,00	24,33
Bavaria	42,00**	51,00	52,00	48,33
Greece	44,00	33,00	40,00	39,00
Israel	40,00	44,00	55,00	46,33
Lithuania	22,00	27,00	46,00	31,67
Average	32,00	33,60	48,20	37,93

* The document was changed slightly after coding round 1

** This coding was done in a plenary session of the WG

Caused by the multiple coding strategy, the calculation of intercoder reliability coefficients (e.g. Cohens kappa, see [38]) would have been quite complicated. Thus the group regarded the percentages as upper thresholds for the (usually more valid) coefficients, as the former are low enough to make clear that the coding quality was not very good. Following [38], the coefficients should be at least 0.7.

The comparisons of round 1 with round 2 respectively round 3 (columns 1—2 and 1—3 in table 2) were appropriate only in a very limited sense, because we had changed the category system after round 1. Still, the improvement of the agreement between rounds 2 and 3 was moderate. In the case of Greece, it was even worse than between round 1 and 2. It has further to be taken into account that the Austrian document was changed slightly, which caused a shift in the numbering of the paragraphs and hereby confused MaxQDA. Additionally, the Bavarian study was coded in the plenum, which might explain the relatively high agreement of the following rounds.

In a very close discussion it was supposed that there might be several reasons for the percentages indicating bad agreement:

(1) As already mentioned, the problematic distinction of *Preconditions* and *Decision* area caused many differences in their subcategories.

(2) There was not much time to define the categories exactly. Therefore, some of them were interpreted quite differently.

(3) The coding teams coded with a very different granularity. Some teams coded words, other sentences, other paragraphs. Despite our agreement to code sentences or paragraphs, this remained a factor because the documents contained many tables, bulleted lists or figures.

(4) The software tool MaxQDA was very strict in accepting agreements. A closer look at the agreement percentage of coding rounds 2 and 3 of the Lithuanian document, (category *Decision Areas\Intentions\Standards)* showed that MaxQDA had calculated an agreement percentage of 54%. On the other hand, manually counting the agreements resulted in 84%. The apparent reason for this was that in cases where one coder had coded a whole paragraph and the other several sentences in the same paragraph, only one of these sentences was counted as agreement. Assuming that this had happened in other categories too, some of the "real" agreement percentages might be substantially higher than those calculated by MaxQDA.

Nevertheless, Table 3 shows that some categories had even according MaxQDA agreement percentages better than 66%, despite its unfavourable calculation method.

Table 3. Categories with high agreement percentages

Category	Codings	Agreement
Consequences\Dropout	2	100,0%
Decision Areas\Media\Textbooks	6	100,0%
Preconditions\Anthropogenic\Age	7	88,9%
Preconditions\Socio-Cultural\History of ICT and Informatics in School	36	77,5%
Preconditions\Anthropogenic\Gender	19	76,4%
Decision Areas\Teaching Methods\General Education	3	66,7%

In order to offer a solution to these problems for future coding activities, we propose the following:

(1) These problems convinced the working group to change the category system considerably, which led to a new model that will be described in the following section.
(2) One of the next steps of the group will be a close description of the categories, as far as possible based on definitions from literature.
(3) Future activities should define the granularity of the codings very carefully.
(4) Intercoder Agreement should be calculated according to well accepted measures like Cohens kappa (see [38]) by suing appropriate software tools.

8. THE DARMSTADT MODEL (DM)

As mentioned above (see reason (1) in 7.3), the distinction between *Preconditions* and *Decision Areas* caused many coding problems in the respective subcategories. On the other hand this distinction carried very important information in many cases, because it described the borderline between what has to be accepted and what might be changed by the regarded target group. Therefore it was decided to split these problematic aspects from the original model, forming a new dimension (draft label: *Berlin Model Top Dimension*), which reflects the differentiation between *Preconditions*, *Decision area* and *Consequences*. Nevertheless, it turned out that this new dimension would be appropriate only if a second new dimension would be introduced for describing the range of influence respectively the level of responsibility of the persons focused upon.

It was clear that these changes of the original model would affect the meaning of all its subcategories also. Thus it was decided to propose a new model founded on the basis of the BM, which we called the *Darmstadt Model*, honoring the location of the conference.

8.1 The new category system

The new *Darmstadt Model* has three dimensions (see figure 3):

Dimension 1 (*Berlin Model Top Dimension)* comprises the categories of the first level of the original Berlin Model: *Preconditions*, *Decision Areas* and *Consequences*.

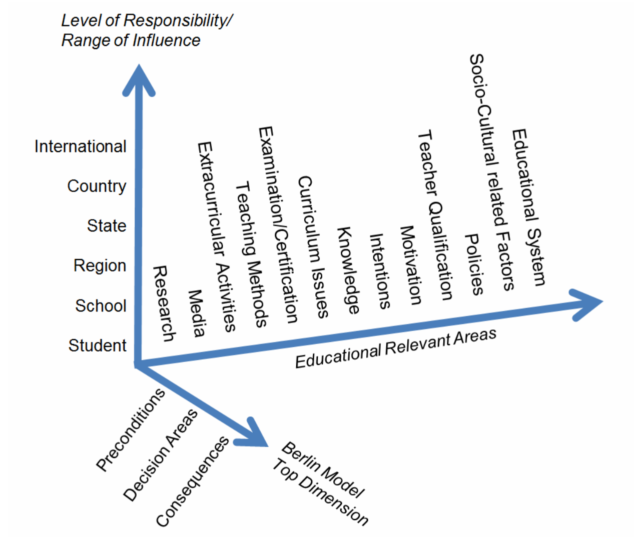

Figure 3. The 3-dimensional Darmstadt Model

Dimension 2 (*Level of Responsibility/Range of Influence*) determines the decision level of the regarded stakeholders. According to the position of the respective person in the school-system, the following subcategories are suggested: 1-*Student/Pupil, 2-Class-room, 3-School, 4-Region, 5-State, 6-Country, and 7-International*.

Dimension 3 (*Educational Relevant Areas*) describes issues that are directly relevant for educational activities. It comprises the remaining subcategories of the original BM that have turned out to be relevant in our context (e.g. *Intentions*) and additionally several other categories that had emerged during coding (e.g. *Educational System*). This dimension has the following categories at level 1and 2:

– *Educational system*: Organizational aspects of subject, Enrollment, School type,
– *Socio-Cultural related Factors*: History of ICT and Informatics in School, Age, Gender, Social and Immigration Background, Family Socialization, Public opinion, Techno-economic development,
– *Policies*: Research and Funding Policies, Education Policies, Quality Management,
– *Teacher Qualification*: Teacher Education, Professional Experience,
– *Motivation*: Student, Teacher
– *Intentions*: Learning Objectives, Competencies, Standards
– *Knowledge*: Computer Science, ICT
– *Curriculum Issues*
– *Examination/Certification*
– *Teaching Methods*, CSE, General Education
– *Extracurricular Activities*: Contest
– *Media*: Technical infrastructure, Textbooks, Tools, Didactical software, Visualization software, Unplugged Media, Haptic media
– *Research*

8.2 Working with the model

Consequently, according to the structure of the DM, coding has to be performed in three dimensions. This means that every coded text fragment would have to be connected simultaneously at least to one category on each dimension. Nevertheless, in some cases only the categories on the "original" dimension 3 (*Educational relevant* *areas)* might be essential. *In such cases coding in the two* new dimensions 1 and 2 might be omitted, which would lead to the default-meaning of "undecided".

Additionally, the DM might be used in a very flexible way. This could be realized, e.g. by folding respectively unfolding the sub-categories below a certain level, depending from the relevance of the regarded text that will be coded. For example, it might be sufficient to apply the category hierarchy in some cases down to *Teacher Qualification*, while in other cases it might be suitable to apply the categories of the two lower levels (see figure 4).

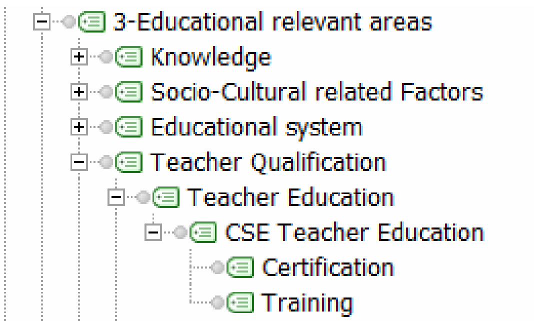

Figure 4. Unfolded subcategories

Unfortunately, the software tool MaxQDA doesn't support multi-dimensional coding. Therefore it has to be simulated by representing the three dimensions by "artificial" top-level categories (see figure 5) and performing *multiple coding* (at least one coding per top-level-category for a point in our 3-dimensional "coding space").

Depending on the specific focus of its application, the DM might be expanded at certain categories by plugging-in other specific category systems or taxonomies: For example the ACM *Computing Classification Scheme* into the category *Educational relevant areas\Knowledge\Computer Science,* the new *CSTA Standards* (e.g. from draft 2011) into the category *Competencies* or the taxonomy for CSE research that was developed by [40] into the category *Research.*

In this paper we apply this technique e.g. by plugging in the taxonomy of [3] in the category *3-Educational relevant areas\Intentions\Learning Objectives* (see section 9.6).

8.3 Recoding of the case studies

After introducing the DM, the codings of the case studies were adapted to the new category system. Thus, codings of joined categories were also joined into the new category; codings of all subcategories of the former top-level categories *Preconditions, Decision areas* and *Consequences* were doubled by firstly copying them to the new subcategories of the *Berlin Model top dimension* (see figure 5) and then moved to the new subcategories of the *Educational relevant areas* dimension. For example a coding of the old category *Decision areas\Intentions\Standards* was copied to *Berlin Model top dimension\ Decision areas* and then moved to *Educational relevant areas\ Intentions\Standards* afterwards. Thus, all codings of the old categories were removed. At the end the old subcategories were deleted.

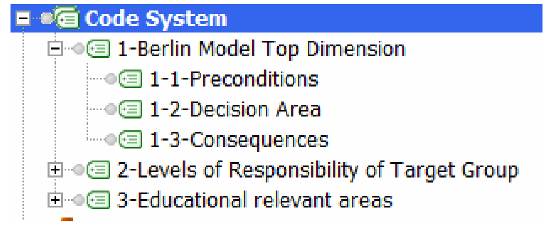

Figure 5. The Darmstadt Model in MaxQDA

Afterwards the intercoder agreements percentages of the new MaxQDA project (that reflected the Darmstadt model) were calculated. The results showed a clear increase, which indicates that at least some of the worst coding problems were solved by the new model.

Table 4. Intercoder agreements according to the new model

Comparison	**1 -- 2**	**1 -- 3**	**2 -- 3**	**Average**
Austria	18,00*	20,00*	54,00	30,67
Bavaria	41,00**	49,00	56,00	48,67
Greece	48,00	34,00	49,00	43,67
Israel	45,00	58,00	63,00	55,33
Lithuania	26,00	30,00	53,00	36,33
Average	35,60	38,20	55,00	42,93

* The document was changed slightly after coding round 1

** This coding was done in a plenary session of the WG

These results might suggest that the DM is more suitable to the coded texts, but nevertheless a reasonable evaluation will have to be performed as one of the next steps.

9. IMPORTANT CATEGORIES

In this section some categories will be illustrated by reporting a summary of the coded text segments from the case studies. As there were about 70 categories (and more than 1100 coded text fragments) after the three coding iterations, it is not possible to cover all categories here. Thus the categories that are suggested to be the most important ones will be selected. The choice was determined by ranking categories according to the number of coded documents (out of 5 case studies x 3 coding rounds = 15 possibilities) in which the respective category had been applied.

Despite the adaptation of the coding results to the new Darmstadt model (DM, see section 8.3), nevertheless the case studies were originally coded using the Berlin Model (BM). Therefore, these "most important" categories should be selected according to the coding results following the BM.

In order to compare the coding frequency of these categories, it did not seem suitable to compare the categories of all levels. In this case, the categories with many subcategories would be put at a disadvantage, despite the fact that these might be the most discussed and, therefore, most interesting ones. Thus the counting was restricted to a certain level of the category system, adding up the codings of the subcategories of the ignored deeper levels. A short look at the hierarchy showed that this level could be 3 or 4, because otherwise the categories would be too abstract or too detailed. The highest ranked categories according to this measure were the following:

Table 5. Categories coded in 10 or more documents

Category path	Coded documents
Research	15
Preconditions\Teacher Qualification\Teacher Education	15
Preconditions\Socio-Cultural\History of ICT and Informatics in School	15
Preconditions\Socio-Cultural\Organizational aspects of subject	15
Preconditions\Socio-Cultural\Education system	15
Decision Areas\Content\Computer Science	15
Decision Areas\Curriculum Issues	14
Preconditions\Anthropogenic\Gender	12
Decision Areas\Content\ICT	12
Decision Areas\Intentions\Learning Objectives	11
Decision Areas\Intentions\Competencies	10

Because of the poor information the case studies contained with regard to the category *Research* and due to our long-term plan to investigate the whole research field using the methodology of [40], we cancelled the illustration of this category for this paper.

As the case studies covered only five countries (respectively states) the summaries cannot be regarded representative for the international situation in any way. Due to the methodology, which cuts text segments out of its context, the resulting texts might not even describe the situation in the single countries properly any more. In order to document that we are well aware of this, the names of the regarded countries were removed. To keep this anonymity, Bavaria is also referred to as a "country" in the following sections. As Bavaria runs its own, independent education system and has even more inhabitants than the four regarded "real" countries, this should be appropriate.

Nevertheless, the text fragments show the enormous diversity of circumstances and implementation details within the regarded categories. Following the heading, the paths of the categories in the old (BM) respectively new (DM) model are displayed.

9.1 Educational systems

BM: Preconditions\Socio-Cultural\Education system

DM: 3-Educational relevant areas\Educational system

Some of the information that has emerged from the codings of this category was already used in the section 3, e.g. in figure 1.

The overall organizational concept of education varies from federal to central, but is within the responsibility of the government in all regarded countries.

Primary Education starts at the age of 6 in all regarded countries. It takes four years in some countries, 6 years in others and somewhere partly even 8 years. In one country, ICT plays an important role in this stage already, while in others only a very small amount of schools offers information technology courses.

SE is generally split in two stages in all countries, e.g. lower/higher SE or junior/senior high schools. In some cases there is only one common type for lower secondary schools, in other cases up to four. Somewhere it is very easy to switch between these different types, in other countries very difficult. One country has a strict vertical structure in SE, where the students are separated at the age of ten into three school types of different level. In other countries the school system structure is more horizontally structured, easing the switch from one school type to another.

Three case studies report about 3-4 different directions of study within the same school type:

1. *theoretic, scientific, technology,*
2. *science & technology, foreign languages, economy and music & arts,*
3. *humanities, mathematics and science, technology, art.*

Most of the countries considered allow elective courses during SE. In the most liberal countries, 60 % of the lessons in upper SE are the same for everybody while 40% are to be chosen by each student. However, the breadth of choice is usually constrained by budgetary limits and, therefore, dependent on the size of groups opting for particular subjects.

In one country full-time education is compulsory from the age of 6 or 7 to 16. In two others, students have from 5 to 10 lessons per day.

There are more than thousand schools in each of the countries considered with apparent very different average school size. In one country there is an average of 174 students per secondary school, in another 970 students per school. In some countries there are more parallel education paths that lead to university by the maturity exam, while in another only the central examination at one type entitles to enroll at any university directly.

Generally, at the end of secondary school students take matriculation exams. Some exams are necessary for obtaining matriculation diploma, while others are not mandatory.

9.2 Organizational aspects of subject

BM: Preconditions\Socio-Cultural\Organizational aspects of subject

DM: 3-Educational relevant areas\Educational system\Organizational aspects of subject

Embedded into the overall educational system, the organization of the subject of Computer Science into the overall curriculum plays an important contextual role.

Basically there are four possible structural implementations of Informatics courses in school curricula [31]:

1. Compulsory subjects: all students of a certain grade or at least all students of a certain direction of study have to attend the course that is explicitly visible in the timetable. Examples are German Language, Mathematics or Geography.
2. Optional subjects or courses: the students are offered a variety of subjects additionally to the compulsory ones. They may choose none, one or more of them, e.g. modern dance, chorus or rock climbing.
3. As a compromise between these two extremes there are subjects to be chosen out of a list of choices from which students have to take (at least) one, e.g., one of several foreign languages.
4. Integration: some topics might be taught integrated into other subjects, e.g. traffic education into the subjects of physics and geography.

In primary schools none of the reported countries has implemented any form of compulsory CSE.

In lower SE, there is a compulsory subject for CSE only in two of the countries considered, in one of them, in the common type, in the other one in two (of three existing) types of schools. Another state has a compulsory subject for Information technology. In a fourth one, a CS curriculum for junior high schools is under development and experimentally run in some schools. In some school types, Informatics does not figure in the compulsory curriculum of the lower secondary level at all. But in most of these cases Informatics is a topic in the optional part of the curriculum.

In higher SE, generally students can opt for various specialization areas, one of them informatics. The specialization areas can be studied from 10th or 11th to 12th or 13th grade and the respective topic can serve as examination topic for the school-leaving examination (matriculation exam, Matura). Additionally there are some countries, where in certain school types CSE is mandatory for all students or for some directions of study only.

In some countries the school system is characterized by a rather large degree of curricular freedom at the school level. Thus schools may set a particular focus at specific topical areas. Such areas might be music or sports; it might be also foreign (or ancient) languages and last but not least it might be Informatics (with rather open semantics of this term). This implies that students reach the subsequent stage of education with a high variance of CS- or ICT-related knowledge.

9.3 History of ICT and Informatics in School

BM: Preconditions\Socio-Cultural\History of ICT and Informatics in School

DM: 3-Educational relevant areas\Socio-Cultural related Factors\History of ICT and Informatics in School

In order to understand the current situation and efforts for the future, it seems necessary to have a look at the history of ICT and Informatics, which has been part of SE in one or the other way for nearly 50 years. We found the following milestones of history of ICT and Informatics in the case studies.

Already in the 60ies some countries started to teach Informatics in schools [20], offering elective courses. In admiration of the impressive achievements of the first mainframe computers (e.g. the calculation of the space trip to the moon) the first approach was focused on hardware, it was proposed to teach assembler programming, Boolean algebra and formal languages.

With the creation of Informatics as a serious scientific discipline, a new didactical approach stressed the systematic development of algorithms, which promised valuable educational effects. Triggered by the educational reform of the 70s [56], it was supposed to pay attention particularly on the application background of the programs that the students should develop. Computer science as an elective subject was available in some high schools since the 1970's, though curricula have changed a few times since then.

Simultaneously with the increasing propagation and the dramatic drop in prices of software applications at the beginning of the 80s it was suggested to teach the usage of standard software instead of programming skills. As a consequence, the Computer Science education in some countries still focuses rather on applications such as text-processing, presentation-software, spreadsheet-software, and communication and information retrieval over the internet (shortly, ICT-topics covered by the ECDL). Some countries started "Informatics" courses in these days, which were merely dominated by ICT Education.

Triggered by the spread of the Internet, the *Information-centered Approach* was elaborated. It claimed that Informatics as the science of information should be taught as well as Physics as the science of energy and Chemistry as the science of matter. Following this approach, the students should learn mainly how to deal with extensive or complex information [11]. Hubwieser et al. [33] had introduced a special information-oriented curriculum that was based on modeling already in 1997: „The emphasis lies on the representation of information about complex systems, which we call modeling. In our opinion these techniques support the students in nearly all problem solving tasks, within Informatics as well as within all other subjects". This teaching approach integrates some goals, content and methods of the former approaches, as far as they have proven to be suitable.

At the beginning of the new millennium, one of the considered countries re-named its Informatics course to *information technologies* (IT).

9.4 Gender Aspects

BM: Preconditions\Anthropogenic\Gender

DM: 3-Educational relevant areas\ Socio-Cultural related Factors\Gender

The findings on gender aspects in the case studies cover different aspects of Informatics education from prior knowledge and beliefs to the decision to take an Informatics class in school or learners' performance in Informatics classes.

According to [9], the assessment of interventions in classes of various grades (actually from Kindergarten to grade 12) [50] has shown no gender differences in both interest and capability. Gender differences became noticeable at grades 5 to 6 and pronounced in higher grades, notably in upper secondary level. Apparently adolescence, role patterns, and role expectation introduce inhomogeneity.

In 2009 the JIM-survey [44] has shown that in a certain German state about 90% of the boys and girls used Internet more than once a week, the average daily usage time of Internet was about 134 min. 47% of this time was used for communication, 18% for games, 14% for searching information and 23% for entertainment (e.g. music). The only real big difference (more than 20% relatively) between boys and girls in regard to these percentages was gaming: boys spend 24% of their Internet time for games, girls only 8% [44].

In one of the considered countries young women at the Gymnasiums are more successful (2.5% of the boys have to repeat a grade, but only 1.5% of the girls). As far as the new subject Informatics in the same country is concerned, research has just begun. In the same country a survey among the Informatics teachers [52] was conducted in autumn 2009. About 500 answered the online questionnaire. One of the most interesting results was that the performance in CS of the girls is clearly better compared to the boys in grade 6, slightly better in grade 7, but clearly worse in grade 9 and dramatically worse in grade 10 according the perception of the teachers.

In another country, most students start their formal studies in CS in grade 10, that is, when they are about 15 years old. Girls are a minority among the advanced students of CS. In 2006, 1133 girls and 1285 boys took the basic 3-unit exams, with an average grade of 84.07 and 82.86, respectively. 1593 girls and 3914 boys took the advanced 2-unit exam, with an average grade of 89.23 and 88.89, respectively.

9.5 Teacher Education

BM: Preconditions\Teacher Qualification\Teacher Education

DM: 3-Educational relevant areas\Teacher Qualification\Teacher Education

Following Shulman [60] we can separate domain specific subject matter content knowledge, pedagogical content, knowledge and curricular knowledge from general pedagogical knowledge. Mishra and Koehler [47] have added knowledge about technology to this model.

Recent investigations in traditional subjects like Mathematics [41] as well as experiences in Informatics [22] have shown that teacher education is a very critical factor for the learning success of the students. The didactical knowledge of Mathematics teachers has turned out to be a very dominant factor e.g. for the cognitive activation of their students [6]. Therefore one cannot address the issue of Informatics education without addressing the issue of teacher education.

In many countries teacher education is a quite complicated procedure with many different regulations and a wide range of possible paths to become a teacher for different schools. Teacher education for Informatics relies on the regular teacher education system of each country. But it is even more different than the different systems, because of a missing tradition in Informatics teacher training in this still young subject.

In some countries the majority of active teachers got their formation in CS during short in-service courses despite the existence of a formal degree program for teacher education, because only younger teachers had a chance to study CS-education at the university level.

Teacher education is generally performed by two different organizations: universities and/or specific teacher colleges, sometimes depending from the target level of education. In some country specific school types, teachers are mostly practitioners with an academic degree who get an additional in-service pedagogic education. In one country the teaching license has to be acquired generally as a specific university degree in one or more subjects.

Currently from only one country is reported that there are nationwide standards for teacher education, also for CSE.

In one case, all primary and secondary school teachers are selected based on the results of a certain nationwide exam. Especially, the teaching of Computing and ICT in SE is conducted by teachers holding an undergraduate degree in Computer Science, Computer Engineering or Applied Informatics. In another case, in order to teach CS in secondary schools, teachers must have a teaching license, issued by the Ministry of Education. To obtain such a license, a teacher must have at least a baccalaureate degree in computer science, software engineering, computer engineering, information systems, information technologies, or electrical engineering. In addition, the teacher must have a teaching diploma in computer science. These diplomas are issued by colleges and universities, and require the teacher student to take professional CS courses, general educational and pedagogical courses, and specific CS-educational courses, as well as a practicum phase. Teachers who do not meet these requirements are required to take extensive in-service courses.

In several countries some CS teachers were recruited from the high tech industry, switching from industry to an educational career. These have a rich CS background but no formal educational preparation. However, this is a temporary situation, since these teachers are required to complete a teaching diploma in a limited period

For a compulsory subject, there is a need of at least 1-3 teachers per school, depending from the number of attending students. In one certain country the introduction of a new subject for CSE started with several in service training programs at five universities about 8 years before the subject was started. Simultaneously some universities installed a regular course of study for Informatics teachers. The in-service programs led to a regular university degree in CSE and were successfully attended by about 300 teachers. But still, 8 years after the actual start of the subject, about 50% of the practicing teachers don't have a university degree in CSE in this country.

Comparing these views from different countries shows the difficulties and different responsibilities many countries have to cope with while forming Informatics teachers. Only few countries seem to have an organized and widely established a structure for becoming an Informatics teacher. Many countries struggle with the introduction of Informatics as a compulsory subject and/or the shortage of Informatics teachers. They chose very different ways to raise their number with in-service programs. In most regarded countries the pioneers of Informatics at school are retiring now and a shift of paradigm from the technical and programming oriented view to ICT literacy and the social impacts of computer science might be noticed although it is still not agreed on, what an Informatics teacher should be able to teach (see content chapter). There is much research needed in this area.

9.6 Intentions: Learning Objectives

BM: Decision Areas\Intentions\Learning Objectives

DM: 3-Educational relevant areas\Intentions\Learning Objectives

As any other project, teaching processes should be guided by clear defined goals of the intended outcomes. In the case of teaching this means to define what students should be able to do after having attended the lessons.

We discovered 88 explicitly expressed learning objectives in the case studies, which turned out to be very different in several aspects. Concerning the three levels of specificity according [3], instructional objectives were mentioned only in one case study: "We have shown in [31] that a quite simple object oriented program that is context oriented, easily demands up to 40 or more instructional objectives in order to be understood by the students." Concerning the two upper levels of specificity, the distribution of the objectives over the case studies was quite different (see table 6). While the first two studies focused on global objectives, the fourth one described its goals more by educational objectives. One case study did not mention any learning objectives explicitly.

Table 6. Specificity of the described learning objectives

	Global	Educational	Total
Country 1	10	3	13
Country 2	11	2	13
Country 3	3	2	5
Country 4	25	32	57
total	49	39	88

Concerning the knowledge part of the objectives, we tried to assign them to the categories of the *ACM Classification scheme* and found that this was possible for 75 objectives, while 13 others were general, which means outside IT or computer science here. Table 7 shows the results for the *ACM Categories* with 5 or more assigned learning objectives.

Table 7. CS Knowledge of the described learning objectives

ACM Classification Category	Assigned objectives
H. Information Systems\H.0 General	23
H. Information Systems\H.4 Information Systems Applications\H.4.1 Office Automation	8
H. Information Systems\H.5 Information Interfaces and Presentation	6
I. Computing Methodologies\I.1 Symbolic and Algebraic Manipulation\I.1.2 Algorithms	8
K. Computing Milieux\K.4 Computers and Society	6
K. Computing Milieux\K.6 Management of Computing and Information Systems\K.6.1 Project and People Management	5

Apparently more than 30% of the objectives had a knowledge part from *Information Systems\General* according the ACM scheme.

Finally the learning objectives that were addressed in the case studies should be compared to the K-12 Computer Science Standards published by the CSTA in 2011 (see [66] and section 5.2.3). The comparison was restricted to 40 learning objectives that were mentioned in the case studies as the most prominent goals of the educational activities (i.e. presented by lists of explicitly intended competencies or objectives). Regarding the CSTA-levels the result is shown in table 8 (the highest percentage of each country is printed in bold).

Table 8. CSTA-levels of the most prominent objectives

Level	1	2	3A	3B	3C
Country 1	13%	13%	**50%**	25%	0%
Country 2	25%	**63%**	13%	0%	0%
Country 3	25%	0%	25%	**50%**	0%
Country 4	30%	30%	**40%**	0%	0%
Total	23%	30%	**33%**	13%	0%

This shows that on the one hand that in total level 3 (applying concepts and creating real-world solutions) and particularly sublevel 3A (Computer Science in the Modern World) is mostly addressed by the case studies, while level 3C is totally ignored. On the other hand the main emphasis of the intentions is different: country 1 and country 4 have the focus on level 3A, while country 2 has its emphasis on level 2 and country 3 on level 3B.

Similar different is the representation of the five CSTA-strands in the case studies (see table 9), abbreviated as follows: CTH = computational thinking; COL = collaboration; CPR = computing practice; CCD = computers and communication devices; CGE = community, global and ethical impacts. Please note that one learning objective might address several strands.

Table 9. CSTA-strands of the most prominent objectives

Strand	CTH	COL	CPR	CCD	CGE
Country 1	88%	75%	75%	**100%**	50%
Country 2	38%	50%	**63%**	50%	13%
Country 3	**100%**	0%	25%	50%	25%
Country 4	**80%**	20%	70%	**80%**	30%
Total	**73%**	40%	63%	**73%**	30%

While in total computational thinking and computers and communication devices are the mostly addressed strands, the countries show again substantial diversity, e.g. country 1 emphasizes computers and communication devices, country 3 focuses more on computational thinking. Furthermore there are differences in the evenness: country 1 addresses the strands between 50% and 100% of its objectives, while country 2 is low in *community, global and ethical impacts* and country 3 ignores *collaboration* totally.

9.7 Intentions: Competencies

BM: Decision Areas\Intentions\Competencies

DM: 3-Educational relevant areas\Intentions\Competencies

During the coding process it turned out to be quite demanding in many cases to distinguish between the categories *Competencies* and *Learning Objectives*. This seems to depend on the specific point of view of the author(s) of the respective case study or even on the authors of the sources used by them. Within the working group, the following definition was agreed upon: learning objectives reflect the aims of the teaching persons, while competencies describe the needs of the “customers” of an educational process, concerning the desired outcomes. Secondly, according the definition of [72], competencies include also components outside cognition, like motivation and volition as well as the application in a “real world situation”. Thus, it is often not possible to decide whether a statement like “students should be able to ...” describes a learning objective or a competency. Therefore, many of the learning objectives that were reported in the section above might define competencies as well in another context. This might be true in particular for the global objectives according to [3], e.g.:

– The students should be able to act responsibly and efficiently in a world of work and profession that is ubiquitously penetrated by IT.

In one of the considered countries, the IT curriculum emphasizes value-based attitudes and general skills. The aims of separate IT courses are much more narrow and pragmatic there. In the last two grades of basic school (9th and 10th) students are taught to summarize ICT knowledge that was obtained in school and out of it, improve their ICT skills, and are prompted to get deeper awareness of Informatics as a science which might encourage them for further studies of the subject. The aims of general course of IT for the 11th and 12th grades are cognitive as well, while the advanced course is intended for the training of specific application skills in one of the three chosen areas of ICT (data base, programming or multimedia) [4]. The targets for general skills are divided into four groups:

– Learning and working
– Communication
– Problem-solving and research

– Critical thinking and evaluation

In another country, compulsory Informatics/ICT-instruction will be shifted from 9th grade forward to the lower secondary level in order to provide room for Informatics/CS education in the 9th grade. It should provide skills needed by young people to act prosperous in a "digital world". Hence, Informatics in secondary schools should teach those concepts that make Informatics distinct from other disciplines taught in school but serve directly to qualify students to become socio-technically oriented thinkers and socially responsible members in societies resting on technical progress (e.g.[28],[48]).

In some other countries, the curriculum is presented only in terms of content (knowledge units and the time that their teaching process should take), and structural organization (what should be the order of teaching in each of the units). Learning objectives are not described, and neither are competencies.

9.8 Curriculum Issues and Content

BM: Decision Areas

Curriculum Issues; Content\CS; Content\ICT

DM: 3-Educational relevant areas

Curriculum Issues; Knowledge\CS; Knowledge\ICT

The analysis of the case studies provided strong evidence of the categories *Curriculum issues* and *Content* according to both models (BM resp. DM), which were coded according to the following definitions:

– The category *Content* is related to a specific knowledge element, e.g. of a taxonomy that classifies CS subject areas like the *ACM classification scheme*. In a text corpus that should be analyzed, CS topic descriptions could be provided down to small size granularity when for instance topics of a single CS-lesson are described.

– *Curriculum issues* are often dealing with CS topics also, but additionally offer a broader and more complex view on the educational context. The target group, the sequencing of topics or methodical aspects of their introduction into the classroom work might be described there.

We identified indicators of both categories in every case study. But in some cases it was very difficult to distinguish exactly between *content* related aspects and *curriculum issues*. Therefore we will combine the codings of these two categories in this section. Nevertheless, we consider it useful for the purposes of further analysis to keep this differentiation in mind.

As already explained in section 8.2, the category system can be used at different levels of abstraction, depending on the application context. Concerning the category *Content,* this holds also for the differentiation of ICT-related content (*Content\ICT*) and CSE-content (Content\CSE). Apparently, it is necessary to decide between these two subject areas in some cases, e.g. when deciding about the predominant orientation of a certain teaching concept (ICT versus CS). On the other hand it might be impossible to decide between these two in some other cases. Anyway, both concepts are closely linked to curriculum issues in CS and are mutually influencing each other, e.g., if a software system is used as an example to illustrate a certain CS concept. The analysis of the five case studies regarding these aspects results in the conclusion that most of the examined curricula refer to ICT as well as to CSE. Therefore, we will present them in an integrated description in this paper. Nevertheless, predominantly ICT-related content is presented only in a short summary in some cases.

According to the case studies, the curricula for all schools and subjects are generally decided by the government. In some cases the contents are strictly prescribed for all subjects, in others there is some freedom of choice for the schools concerning subjects or learning content.

In one country, the European Computer Driving License (ECDL, see www.ecdl.com) plays an important role in the selection of knowledge elements for the officially prescribed curriculum. This led to a curriculum that has a focus on ICT skills in lower SE. In upper SE students can opt for Informatics as a specialization area and receive an education that is more oriented towards the science of Informatics, depending on the qualification of the respective teachers.

In another country, the selection process for the curriculum followed the *information-centered didactical approach* [11], arguing that the application fields representation, processing, and transportation of information, as well as interpretation of representations play an important role in the information society. [32] proposed to derive knowledge elements for the curriculum following this partition:

(1) *Representation of information*,
(2) *Processing and transport of representations*,
(3) *Interpretation of representations.*

Depending on their relevance fields, Hubwieser [30] assigned these knowledge elements to one of the following four categories (categories 1-3 might be included in the curriculum):

(1) relevant even beyond the limits of automatic information processing, e.g. modeling techniques, which can be applied to real world systems,
(2) relevant for all ICT systems, e.g. algorithms, principal limitations of computability,
(3) relevant for a certain class of ICT systems, e.g. concept of register machine, data structures of text processors or spreadsheets, principles of object oriented programming,
(4) relevant for a certain instance of ICT systems only, e.g. menu structure of MS Word 2010, how to fix a favorite URL in Firefox 3.0, syntax elements of Java 2.0.

This strategy resulted in the curriculum that is displayed in table 10.

Table 10. Curriculum of country 2

Grade/Theme	Basic concepts
6.1 Representation of information	Representation, interpretation
6.2 Object oriented modeling of documents	Object, attribute, class, method, aggregation
6.3 Hierarchical structures	Tree, root, leaf, node, edge
7.1 Network structures	Link, anchor, Internet, cyclic structures, reference, Hypertext
7.2 Exchange of information	E-Mail, attachment, mail server, principles of E-mail transfer
7.3 Basic concepts of algorithms	Representation of algorithms, control structures (sequence, conditional, repetition)
9.1 Functional modeling	Data flow diagrams, function, parameters, return value, concatenation, simple data types
9.2 Data modeling	Object (Data record), class (table), association (relation), query language (SQL), data protection, data security
10.1 Object oriented modeling and programming	Object and class diagrams, state and sequence charts, variable, assign statement, array, data encapsulation, generalization, polymorphism, specialization, state of objects, state machines
10.2 Generalization and specialization	Inheritance, polymorphism, class hierarchies
10.3 Software project	Combination of several modeling and implementation techniques (e.g. OOP and data base systems)
11.1 Recursive data structures	Lists, trees, graphs, recursive algorithms
11.2 Software engineering	Project planning, software life cycle, process model, coordination of parallel processes
12.1 Formal languages	Alphabet, BNF, grammars, productions, syntax, semantics. Syntax diagram, finite automaton
12.2 Synchronization of parallel processes	Communication protocols, layer models, topology of computer networks, Internet, semaphore and monitor concept
12.3 Basic functionality of a computer	Components (CPU, memory, storage systems), von-Neumann principle and architecture, register machine, assembler language, computer as a state machine
12.4 Limitations of computability	Run time complexity, principal and efficiency-caused limitations, data encryption, Halting problem

In a third country, ICT content dominates the curriculum throughout the first two years at SE. In the third year the learners are introduced into fundamental algorithms and programming using Logo (see table 11).

Table 11: Curriculum of country 3

Cognitive Domain Axe	**General Aim**	**Principle Indicative Concepts of Cross-curricula Perspective**
Grade 1		
I get to know the computer as a unified system	Basic Informatics notions; Historical presentation of the development / evolution of computers; The hardware of the computing system; The software of the computer system; Hardware, software and data protection; Ergonomy-Precautions	Technology System, Change Code, Communication, Time-Space, Hygiene Cooperation
I communicate with the computer	The communication graphical environment; The web browser environment	Communication, Technology Expression, Aesthetics Symbolism Time-Space
Use of expression, communication, discovery and creativity tools	Expression (writing-painting) with computer assistance Information and communication using the Internet Organization, cooperation, programming, contribution to the team goals Undertaking responsibilities	Communication, Technology; Expression, Symbolism Time-Space Change, Progress Cooperation Interaction
The computer in school and everyday life	Computer uses in everyday life (in school, home, banks etc)	Technology Communication, Expression Time-Space Change, Progress Cooperation, Exploitation Interaction
Grade 2		
I get to know the computer as a unified system	Computer units; Multimedia computers (their characteristics) and the multimedia applications; Representation of the information using computer; Computer networking-Networks and their operational exploitation	System Time-Space Communication Symbolism, Code Organization Part-Whole
I communicate with the computer	Discovery with the — assistance" the computer can provide Saving and managing files	Technology Linearity Interaction Organization, Change
Use of expression, communication, discovery and creativity tools	Use of tools: numerical processing and graphical presentation of data; Presentation tool; Information and communication using the Internet	Technology, Communication Time-Space Taxonomy, Exploitation Change, Problem Expression Validity, Cooperation
The computer in profession	Changes and consequences in the work environment due to the new technologies introduction and use; Emerging needs	Technology, Work Time-Space Exploitation Change, Adaptation Necessity
Grade 3		
I get to know the computer as a unified system	Programming languages; Basic steps of solving a problem with the computer use; Programme development and execution	Problem, Solution Evaluation, Organization Succession, Differentiation Change, Adaptation Communication Interaction
Use of expression, communication, discovery and creativity tools	Multimedia application development	Expression, Aesthetics Interaction Linearity Cooperation, Evaluation
The computer in society and civilization	The influence of Information and Communication Technologies on science, art, civilization, language, environment, quality of life etc	Technology, Civilization Digital world Environment, Communication Interaction Employment, Progress Exploitation

The overall goal of the curriculum in a fourth country is to understand the scientific concepts that are integrated in current technological developments, by exposing them to the scientific aspect of CS as well as to its more technologic aspects, introducing students to fundamental concepts of CS (see table 12).

The overall principle is a zipper-approach, integrating theoretical and practical aspects and modularity with elective components [21], [5]. After the two foundational units, which zip conceptual notions with algorithmic (computer programming) structures, the program offers 6 alternatives for the 3rd unit, decided by the teacher. Students who wish to continue their CS studies in high school take another two units, complementing to the highest 5-unit level. These two units are taken in grade 11 or 12.

Table 12. Curriculum of country 4

Unit	Content
1,2	Algorithmic design, conditional execution, repetitive execution, correctness and efficiency of algorithms, sub-tasks (methods, functions), basic data structures (arrays), characters and strings. Implementation of algorithmic solutions in Java or C# (depending on teachers' choice).
3	Alternatives (chosen by the teacher): – information systems, – logic programming, – computer graphics, – computer organization and assembly language, – functional programming, – introduction to web programming. In each of these options the student is expected to develop a final project
4	Software design: recursion, data types, the data structures stack, queue, linear list and binary tree, efficiency.
5	Advanced unit; Alternatives (chosen by the teacher): – Introduction to computer programs, – Principles of computer use, – Drawing with a computer, – Text and keyboard, – Internet and electronic mail, – Projects with Logo.

The fifth case study reports from an IT course in 5th and 6th grade that is integrated into different subjects. It contains five parts:

- Introduction to computer programs,
- Principles of computer use,
- Drawing with a computer,
- Text and keyboard,
- Internet and electronic mail,
- Projects with Logo.

In the 9th and 10th grade the IT course should summarize and systematize students' knowledge, guide them towards a purposeful usage of their skills, and draw attention to the correct application of the technologies and their legitimacy.

For those who wish to grasp fluency in programming principles, a 34 hours optional module on "Elements of algorithms and programming" offered:

- Conception of algorithms, ways of writing
- Programming languages, compilers
- Preparation of algorithms, coding and running the program
- Dialog between program and user
- Entering and output of data, printing formats
- Main actions of algorithms: assignment, loop
- Simple data types
- Stages of program development
- Control data and correctness of program
- Programming style and culture
- Simplest algorithms and their programming

The IT course for upper secondary grades (11 to 12) offers several optional modules, including developing algorithms and programming as well as databases. The teaching of programming embraces four main fields:

1) basic constructions of Pascal,

2) data structures,

3) algorithms,

4) a version of the Pascal language in a Free Pascal environment.

10. DISCUSSION

The Berlin Model was criticized for not considering normative aspects of curriculum development, excluding the rationale of topics and objectives of learning processes and postulating the independency of its fundamental categories. An excellent overview of the criticism on the BM from several authors since the1960th is shown in [2], p. 47: "The category of learning and the *Lerntheoretische Didaktik* are neutral - they do not constitute norms, do not prescribe an ultimate intention, and do not determine an underlying rationale. They neither restrict behavior, nor determine decisions (e.g. regarding intention and content). Instead, the *Berliner Modell* facilitates teachers by only making explicit the range of possible decisions".

As the case studies were produced based on the BM, they did not cover these aspects at all. As a consequence, currently the *Darmstadt Model* is neglecting the normative foundation of CSE also. Therefore it is to be expected that the Darmstadt Model will have to be expanded with additional categories that will consider the rationale of objectives and topics of learning processes in CSE.

A serious flaw of the methodology was the poor intercoder agreement (see section 7.3). This should be improved by proper definitions of the categories, common granularity and the application of more suitable software tools before the next coding activities will take place.

As described in section 6, [34] developed a taxonomy for CSE research by comparing a set of publications. The comparison of the Darmstadt Model with this taxonomy could serve as a first check concerning its completeness. Table 13 compares the categories of [34] with apparently corresponding categories of the DM.

Table 13. Darmstadt Model compared to [34]

Taxonomy of [34]	Darmstadt Model
System	*Teaching methods, Media*
Technology	*Teaching methods, Media*
Resources	*Media*
Other technical	*NONE*
Theoretical pedagogy	*Research, Learning Methods, Intentions, Motivation etc.*
Practical pedagogy	*Learning Methods, Intentions, Motivation etc.*
Curriculum	*Curriculum Issues, Content*
Social factors	*Socio-Cultural related Factors*
Psychology factors	*Research*
Other educational	*Research, Learning Methods, Intentions, Motivation etc.*
Other	*NONE*

As this comparison shows, there might be some deficit concerning Educational theory (pedagogy) in the DM. This has to be investigated more closely by coding some publications of this research field in the future.

One might be tempted to derive the common grounds or the most apparent differences between the situations of CS in SE in the five countries covered by the case studies. But, unfortunately, the descriptions of the relevant aspects were very different regarding abstraction level, the detail or context and the viewpoint of the authors. Additionally, much information about context of the statements was lost by the coding and extraction process. Therefore a fair comparison of the situations in the covered countries should make use of a questionnaire that would be designed specifically for that purpose (see section 11).

11. FUTURE WORK AND APPLICATIONS

As already stated in section 4, the working group agreed to establish a long-term collaboration, aiming to stimulate, evaluate and disseminate research in CS in SE. Therefore the DM in its current state is regarded as an interim result that should be evaluated, expanded and maybe corrected in some regards. As the coding of the five case studies produced by members of the working group has shown, the original BM is not yet complete.

For the moment it might be applied as a checklist for the production of case studies about the situation of CS in SE in further countries or states. Additionally it could be applied for a systematic comparison of publications about the situation of CSE in different countries, e.g. from Poland, [63], USA [62], or as a guideline for semi-structured interviews of experts.

The next step in the development of the model will be a careful description of the categories. As far as possible this will be done following existing literature relevant to the respective category. As some categories have a specific meaning within the model, those descriptions will have to be defined in a discussion process within the working group. The definitions will be validated by coding more documents and measuring the intercoder reliability coefficients.

After validating the definitions, the Darmstadt Model could be applied to produce a questionnaire for a quantitative survey about the situation of CSE in different countries. The definitions could be used as explanations of the questions. This could contribute to identify common ground as well as differences in the teaching approaches of the participating countries.

Further, the categories of the DM (in a further development stage) could serve as a source of metadata for categorizing research papers about CSE concerning their specific circumstances. Finally, based on this metadata, a categorization of research in CS in SE could be conducted, using the taxonomy of [40] for the category *Research*.

This would be the point of time the working group would have reached its original goal: to develop a research framework for CSE in Secondary schools.

12. CONCLUSION

The original goal of the working group was the development of a research framework for CSE in SE. As very often in scientific research, this goal turned out to be very ambitious and much harder to reach than originally expected, particularly if there is only one week for (physical) collaboration.

Nevertheless, this report delivers a model - still to be discussed in details within the CSE-community - that will foster and strengthen future work by using it as a theoretical framework for research in the area of CSE. Furthermore, this paper does not pretend that it covers the state of CSE/ICT-Education in specific countries. Rather it is focused on the rationale of the Darmstadt Model, based on the qualitative empirical analysis of several case studies.

Hereby, the working group managed to overcome the main difficulties and completed a substantial part of the long way to the original goal. But there is still much to do in this field.

13. ACKNOWLEDGEMENTS

We have to thank Barbara Owens for the idea, for the support of the working group and for her hard efforts to find members from the US, Liz Adams and Jürgen Börstler for managing all the working groups, providing us with information, confidence, food and drinks and all the patience with the “non-natives”, Guido Rößling for the organization of a great conference and Margot Phillipps for the linguistic assistance.

14. REFERENCES

[1] Recommendation of the European Parliament and of the Council of 23 April 2008 on the establishment of the European Qualifications Framework for lifelong learning. Official Journal of the European Union, C 111 (May. 2008), 1–7.

[2] Allert, H. 2005. Modeling coherent social systems for learning, Univ.

[3] Anderson, L. W. and Krathwohl, D. R. 2001. A taxonomy for learning, teaching, and assessing. A revision of Bloom's taxonomy of educational objectives. Longman, New York.

[4] Armoni, M. and Gal-Ezer, J. 2004. On the achievements of high school students studying computational models. SIGCSE Bull 36, 17□21.

[5] Armoni, M., Benaya, T., Ginat, D., and Zur, E. 2010. Didactics of Introduction to Computer Science in High School. In Teaching fundamental concepts of informatics. 4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010, Zurich, Switzerland, January 13-15, 2010 ; proceedings, J. Hromkovic, R. Královic and J. Vahrenhold, Eds. Lecture notes in computer science 5941. Springer, Berlin, 36–48.

[6] Baumert, J., Blum, W., Brunner, M., Dubberke, T., Jordan, A., Klusmann, U., and et al. 2009. Professionswissen von Lehrkräften, kognitiv aktivierender Mathematikunterricht und die Entwicklung von mathematischer Kompetenz (COACTIV): Dokumentation der Erhebungsinstrumente. Berlin: Max-Planck-Institut für Bildungsforschung. Materialien aus der Bildungsforschung 83.

[7] Bell, T., Andreae, P., and Lambert, L. 2010. Computer Science in New Zealand high schools. In Proceedings of the Twelfth Australasian Conference on Computing Education - Volume 103. Australian Computer Society, Inc, Brisbane, Australia, 15–22.

[8] Bennedsen, J. and Schulte, C. 2006. A Competence Model for Object Interaction in Introductory Programming. In Proceeding of 18th Workshop of the Psychology of Programming Interest Group (PPIG 18), 215–229.

[9] Bischof, E. 2010. Interventionen im (Informatik-)Unterricht. Ein Versuch bei SchülerInnen das Bild der Informatik zu erweitern. Klagenfurt, Alpen-Adria-Univ., Diss., 2010.

[10] Blumrich, S. 2007. Schulsysteme und informatische Bildung im internationalen Vergleich. Diplomarbeit, Technische Universität.

[11] Breier, N. and Hubwieser, P. 2002. An information-oriented approach to informatical education. Informatics in Education 1, 1, 31□42.

[12] Brinda, T. and Hubwieser, P. 2010. How to teach didactics of informatics to informatics student teachers. In New developments in ICT and Informatics education, To appear.

[13] Bruner, J. S. 1960. The process of education. [a searching discussion of school education opening new paths to learning and teaching]. A Vintage book V-234. Vintage Books, New York NY.

[14] Coy, W. 1997. Defining discipline. In Foundations of Computer Science, C. Freksa, M. Jantzen and R. Valk, Eds. Lecture notes in computer science. Springer Berlin / Heidelberg, 21–35.

[15] Dagiene, V. 2005. Teaching Information Technology in General Education: Challenges and Perspectives. In From Computer Literacy to Informatics Fundamentals, 53–64.

[16] Dagiene, V. 2008. Teaching Information Technology and Elements of Informatics in Lower Secondary Schools: Curricula, Didactic Provision and Implementation. In Informatics Education - Supporting Computational Thinking, Third International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2008, Torun, Poland, July 1-4, 2008. Lecture notes in computer science. Springer, 293–304.

[17] Dagiene, V. and Mittermeir, R. T., Eds. 2006. Information Technologies at School. Proceedings of the 2nd International Conference ISSEP 2006. Vilnius, Lithuania. Publishing House TEV, Vilnius.

[18] Dörge, C. 2010. Competencies and skills. Filling old skins with new wine. In Key competencies in the knowledge society. Springer, Berlin [u.a.], 78–89.

[19] Dorninger, C. 2005. Educational Standards in School Informatics in Austria. In From Computer Literacy to Informatics Fundamentals, R. Mittermeir, Ed. Lecture notes in computer science. Springer Berlin / Heidelberg, 65–69.

[20] Forneck, H. J. 1990. Entwicklungstendenzen und Problemlinien der Didaktik der Informatik. In Beiträge zur Didaktik der Informatik, G. Cyranek, H. J. Forneck and H. Goorhuis, Eds. Diesterweg, Frankfurt am Main, 18–53.

[21] Gal-Ezer, J. and Harel, D. 1998. What (else) should CS educators know? Communications of the ACM 41, 9, 77–84.

[22] Gal-Ezer, J. and Stephenson, C. 2010. Computer science teacher preparation is critical. ACM Inroads 1, 61□66.

[23] Gesellschaft für Informatik (GI) e. V., Ed. 2008. Grundsätze und Standards für die Informatik in der Schule. Bildungsstandards Informatik für die Sekundarstufe I. Empfehlungen der Gesellschaft für Informatik e. V., erarbeitet vom Arbeitskreis »Bildungsstandards«, Bonn.

[24] Glendinning, I. and Low, M. 2010. Collaborative Initiatives for Promoting Computer Science in Secondary Schools. In Teaching fundamental concepts of informatics. 4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010, Zurich, Switzerland, January 13-15, 2010 ; proceedings, J. Hromkovic, R. Královic and J. Vahrenhold, Eds. Lecture notes in computer science 5941. Springer, Berlin, 100–111.

[25] Gurbiel, E., Hardt-Olejniczak, G., Kolczyk, E., Krupicka, H., and Syslo, M. M., Eds. 2005. Informatics and ICT in Polish Education System.

[26] Hazzan, O., Dubinsky, Y., Eidelman, L., Sakhnini, V., and Teif, M. 2006. Qualitative research in computer science education. SIGCSE Bull 38, 408□412.

[27] Hazzan, O., Lapidot, T., and Ragonis, N. 2011. Guide to teaching computer science. An activity-based approach. Springer, Berlin.

[28] Hromkovič, J. 2006. Contributing to General Education by Teaching Informatics. In Informatics Education – The Bridge between Using and Understanding Computers, R. Mittermeir, Ed. Lecture notes in computer science. Springer Berlin / Heidelberg, 25–37.

[29] Hromkovic, J., Královic, R., and Vahrenhold, J., Eds. 2010. Teaching fundamental concepts of informatics. 4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010, Zurich, Switzerland, January 13-15, 2010 ; proceedings. Lecture notes in computer science 5941. Springer, Berlin.

[30] Hubwieser, P. 2000. Didaktik der Informatik: Grundlagen, Konzepte, Beispiele. Springer-Lehrbuch. Springer, Berlin.

[31] Hubwieser, P. 2007. A smooth way towards object oriented programming in secondary schools. In Informatics, Mathematics and ICT: A golden triangle: Proceedings of the Working Joint IFIP Conference: WG3.1 Secondary Education, WG3.5 Primary Education; College of Computer and Information Science, Northeastern University Boston, Massachusetts, USA 27th □ 29th June 2007, D. Benzie and M. Iding, Eds., Boston,MA.

[32] Hubwieser, P. and Friedrich, S. 1998. Teaching informatics as a subject. In Capacity Building for IT in Education in Developing Countries, IFIP TC3 WG3.1, 3.4 & 2.5 Working Conference on Capacity Building for IT in Education in Developing Countries, 19-25 August 1997, Harare, Zimbabwe. IFIP Conference Proceedings. Chapman & Hall, 291–295.

[33] Hubwieser, P., Broy, M., and Brauer, W. 1997. A New Approach in Teaching Information Technologies: Shifting Emphasis form Technology to Information. In Information technology. Chapman & Hall, London.

[34] Joy, M., Sinclair, J., Sun, S., Sitthiworachart, J., and López-González, J. 2009. Categorising computer science education research. Education and Information Technologies 14, 105□126.

[35] Kalaš, I. and Mittermeir, R. T., Eds. 2011. Informatics in Schools: Contributing to 21st Century Education. 5th International Conference on International Conference on Informatics in Schools: Situation, Evolution and Perspectives, ISSEP 2011, Bratislava, Slovakia, October 2011. Springer, Berlin.

[36] Klieme, E., Avenarius, H., Blum, W., Döbrich, P., Gruber, H., Prenzel, M., Reiss, K., Riquarts, K., Rost, J., Tenorth, H.-E., and Vollmer, H. J. 2004. The Development of National Educational Standards. An Expertise. Bundesministerium für BildungundForschung, Berlin.

[37] LeCompte, M. D., Preissle, J., and Tesch, R. 1993. Ethography and qualitative design in educational research. Academic, San Diego, Calif. ; Toronto.

[38] Lombard, M., Snyder-Duch, J., and Campanella Bracken, C. 2010. Practical Resources for Assessing and Reporting Inter-

coder Reliability in Content Analysis Research Projects. http://astro.temple.edu/~lombard/reliability/.

[39] Magenheim, J., Nelles, W., Rhode, T., Schaper, N., Schubert, S., and Stechert, P. 2010. Competencies for informatics systems and modeling: Results of qualitative content analysis of expert interviews. In Education Engineering (EDUCON), 2010 IEEE, 513–521.

[40] Malmi, L., Sheard, J., Simon, Bednarik, R., Helminen, J., Korhonen, A., Myller, N., Sorva, J., and Taherkhani, A. 2010. Characterizing research in computing education: a preliminary analysis of the literature. In Proceedings of the Sixth international workshop on Computing education research. ICER '10. ACM, New York, NY, USA, 3□12.

[41] Max-Planck-Institut für Bildungsforschung. 2007. COACTIV. Ein Forschungsprojekt zur Untersuchung von Lehrerkompetenz, Berlin.

[42] Maxwell, J. A. 2005. Qualitative research design. An interactive approach. Sage Publications, Thousand Oaks, Calif.

[43] Mayring, P. 2000. Qualitative Content Analysis. http://qualitative-research.net/fqs/fqs-e/2-00inhalt-e.htm.

[44] Medienpädagogischer Forschungsverbund Südwest. 2009. JIM 2009 Jugend, Information,(Multi-) Media. Basisstudie zum Medienumgang 12- bis 19-Jähriger in Deutschland, Stuttgart.

[45] Micheuz, P. 2005. 20 Years of Computers and Informatics in Austria's Secondary Academic Schools. In From Computer Literacy to Informatics Fundamentals, R. Mittermeir, Ed. Lecture notes in computer science. Springer Berlin / Heidelberg, 20–31.

[46] Miles, M. B. and Huberman, A. M. 1994. Qualitative data analysis. An expanded sourcebook. Sage, Thousand Oaks, Calif. [u.a.].

[47] Mishra, P. and Koehler, M. J. 2006. Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge. Teachers College Record 108, 6, 1017–1054.

[48] Mittermeir, R. Informatikunterricht zur Vermittlung allgemeiner Bildungswerte. OCG Schriftenreihe 2010, 271, 54–73.

[49] Mittermeir, R. T. and Syslo, M. M., Eds. 2008. Informatics Education - Supporting Computational Thinking, Third International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2008, Torun, Poland, July 1-4, 2008. Lecture notes in computer science. Springer.

[50] Mittermeir, R., Bischof, E., and Hodnigg, K. 2010. Showing Core-Concepts of Informatics to Kids and Their Teachers. In Teaching fundamental concepts of informatics. 4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010, Zurich, Switzerland, January 13-15, 2010 ; proceedings, J. Hromkovic, R. Královic and J. Vahrenhold, Eds. Lecture notes in computer science 5941. Springer, Berlin, 143–154.

[51] Mittermeir, R., Ed. 2005. From Computer Literacy to Informatics Fundamentals. Lecture notes in computer science. Springer Berlin / Heidelberg.

[52] Mühling, A., Hubwieser, P., and Brinda, T. 2010. Exploring Teachers' Attitudes Towards Object Oriented Modelling and Programming in Secondary Schools. In ICER '10: Proceedings of the Sixth international workshop on Computing education research. ACM, New York, NY, USA, 59–68.

[53] National Council of Teachers of Mathematics (NCTM). 2000. Principles and standards for school mathematics, Reston, VA.

[54] Pre-College Task Force Comm. of the Educ. Board of the ACM, C. 1993. ACM model high school computer science curriculum. Commun. ACM 36, 87□90.

[55] Randolph, J. 2008. Computer science education research at the crossroads. A methodological review of computer science education research, 2000-2005. VDM Verlag Dr. Müller, Saarbrücken, Germany.

[56] Robinsohn, S. B. 1967. Bildungsreform als Revision des Curriculum. Aktuelle Pädagogik. Luchterhand, Neuwied a.Rh.

[57] Rychen, D. S. 2003. Key competencies. For a successful life and a well-functioning society. Hogrefe & Huber, Cambridge Mass. u.a.

[58] Schubert, S. and Stechert, P. 2010. Competence model research on informatics system application. In New developments in ICT and Education. Workshop of WG 3.1, June 28-30, 2010 Amiens, IFIP, Ed.

[59] Schwill, A. 1993. Fundamental Ideas of Computer Science. Zentralblatt für Didaktik der Mathematik, 1.

[60] Shulman, L. S. 1987. Knowledge and teaching: Foundations of the new reform. Harvard Educational Review, Feb., 1–22.

[61] Starruß, I. 2010. Synopse zum Informatikunterricht in Deutschland. Analyse der informatischen Bildung an allgemein bildenden Schulen auf der Basis der im Jahr 2010 gültigen Lehrpläne und Richtlinien. Bakkalaureatsarbeit, Technischen Universität Dresden.

[62] Stephenson, C., Gal-Ezer, J., Haberman, B., and Verno Anita. 2005. The New Educational Imperative: Improving High School Computer Science Education. Using worldwide research and professional experience to improve U. S. Schools. Final Report of the CSTA Curriculum Improvement Task Force. ACM, CSTA, New York.

[63] Syslo, M. M. and Kwiatkowska, A. B. 2008. The Challenging Face of Informatics Education in Poland. In Informatics Education - Supporting Computational Thinking, Third International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2008, Torun, Poland, July 1-4, 2008. Lecture notes in computer science. Springer.

[64] Tucker, A. 2010. K-12 Computer Science: Aspirations, Realities, and Challenges. Teaching Fundamentals Concepts of Informatics, 22\textendash34.

[65] Tucker, A., Ed. 2003. A . Final Report of the ACM K–12 Task Force Curriculum Committee October 2003, New York.

[66] Tucker, A., Seehorn, D., Carey, S., Moix, D., Fuschetto, B., Lee, I., O'Grady-Cuniff, D., Stephenson, C., and Verno, A. 2011. CSTA K-12 Computer Science Standards. Revised 2011. CSTA Standards Task Force. file:///C|/Users/Hubwieser/AppData/Roaming/Mozilla/Firefox/Profiles/eass6k5e.default/zotero/storage/P96QBAVK/CSTAStandardsReview2011.pdf. Accessed 21 June 2011.

[67] Uljens, M. 1997. School didactics and learning. A school didactic model framing an analysis of pedagogical implications of learning theory. Psychology Press, Hove.

[68] van Weert, T. J. and Anderson, J. 2002. Information and Communication Technology in Education. A Curriculum for Schools and Programme of Teacher Development.

[69] van Weert, T. J. and Tinsley J. D. 1994. Informatics for secondary education. A curriculum for schools. UNESCO, Paris.

[70] van Weert, T. J., Ed. 2000. Information and Communication Technology in Secondary Education. A Curriculum for Schools, Genf.

[71] Weeger, M. 2007. Synopse zum Informatikunterricht in Deutschland. Analyse der informatischen Bildung der allgemein bildenden Schulen – durchgeführt auf der Basis existierender Lehrpläne und Richtlinien. Bakkalaureatsarbeit. Accessed 2 February 2010.

[72] Weinert, F. E. 2001. Concept of Competence: A conceptual clarification. In Defining and Selecting Key Competencies, D. Rychen and L. Salganik, Eds. Hogrefe and Huber, Seattle.

[73] Wilson, C., Sudol, L. A., Stephenson, C., and Stehlik, M. 2010. Running on Empty. http://csta.acm.org/runningonempty/fullreport.pdf. Accessed 21 June 2011.