

# Computer Science in Context – Pathways to Computer Science

**Maria Knobelsdorf**

Institute of Computer Science  
Freie Universität Berlin  
Takustr. 9, D-14195 Berlin, Germany  
knobelsd@mi.fu-berlin.de

**Carsten Schulte**

Institute of Computer Science  
Freie Universität Berlin  
Takustr. 9, D-14195 Berlin, Germany  
schulte@mi.fu-berlin.de

## Abstract

In order to increase participation in Computer Science (CS), contextual approaches are often suggested for teaching. Although these approaches are quite promising, we do not know what exactly *context* means and how CS teaching should implement these approaches. In the broadest sense *CS in context* means that CS is linked to subject areas outside CS, helping students to perceive CS as a meaningful, useful, and helpful subject that is established in outside arenas.

The study we present in this paper explores the characteristics of *CS in context* that form possible pathways into the field. For this purpose, we analyse the computing experiences of students majoring in CS-related fields. The study is part of our research project about computing processes. In this project, we investigate students' computing experiences in order to understand how students' interests, motivation, and requirements for computing develop and how computing influences their understanding of CS.

In the current study, we examine general qualitative aspects of CS in context, especially activities and habits that sharpen and stabilize students' self-image and world-image. Because we find surprisingly few examples of specific contexts (such as subject areas) that are related to students' subject choice, we finish this paper with a discussion about possible reasons and conclusions for further studies.

**Keywords:** CS, Context, Pathway, Wider Access, Gender, Computers and Society, CS Ed Research, Pedagogy, Computer Biographies.

## 1 Introduction

Although considerable efforts have already been made to improve the situation, we are still encountering the same substantial problems in Computer Science (CS): decreasing numbers of beginners, constant high dropout rates, and a very low number of female students. In order to understand the reasons for this situation, a recent study of ours investi-

gated students' computing<sup>1</sup> experiences (Schulte and Knobelsdorf 2007). In this study, biographical computing processes of CS majors were compared with those of students not affiliated with CS. We found that CS-affiliated students align their computing experiences with CS, whereas unaffiliated students exclude CS from their computing experiences. This produces a world-image of computing and CS wherein affiliated students perceive themselves as insiders, whereas unaffiliated students perceive themselves as outsiders. Consequently, computing is a starting point for CS-affiliated students, but also a barrier for CS-non-affiliated students to take up CS studies.

Our last study revealed an interesting aspect: students frequently experience computing unhampered by any kinds of regulations, spontaneously, and outside formal schooling. This free leisure time environment represents a context where students experience computers and CS implicitly. For some students this context becomes a starting point to CS, while for other students it is a barrier. We were interested in this aspect of context and examined it in further detail.

Contextual approaches are often suggested in order to increase participation in CS. Fisher and Margolis conclude from their work that "the context of computing is often very important for women students. Among our sample, more women than men link their interest in computer science to other arenas such as medicine, the arts, space exploration, etc." (Fisher and Margolis 2002, p. 80). We can even exaggerate the argument in the opposite direction: CS majors often seem to have a narrower view of CS, they explicitly neglect the role of context, and they set CS apart from other subjects. However, such a narrow view of CS is often the most important reason for low participation in CS. Rosser argues that "female students will be more attracted to science and its methods when they perceive its usefulness in other disciplines" (Rosser 1990, p. 64). Beck, Buckner and Nikolova suggest that "[s]tudents taking CS courses do not wish to study computers as an end in themselves, but rather to become proficient in their use to the extent that they can use the computer as a tool to accomplish some other, non-computer-related goal" (Beck, Buckner and Nikolova 2007, p. 358).

Contextual-based learning is an established approach in science education. In his article *On the Nature of "Context"*

<sup>1</sup> With the term computing we refer to all kinds of computer usage and interaction with the computer.

Copyright © 2008, Australian Computer Society, Inc. This paper appeared at the *Seventh Baltic Sea Conference on Computing Education Research (Koli Calling 2007)*, Koli National Park, Finland, November 15-18, 2007. Conferences in Research and Practice in Information Technology, Vol. 88. Raymond Lister and Simon, Eds. Reproduction for academic, not-for-profit purposes permitted provided this text is included.

in *Chemical Education*, Gilbert (2006) identifies a number of inter-related problems that chemical education has faced: content *overload*, learning of *isolated facts*, lack of *transfer and relevance*, and *inadequate emphasis* of the curriculum. “[T]he educational model that embodies the meaning of ‘context’ must be such that it provides an effective answer to the associated curricula and social problems” (Gilbert 2006, p. 958). Due to the fast accumulation of new scientific knowledge, we have to teach many concepts. This content load leads to teaching of isolated facts without supporting students to understand the correlation and meaning between the facts. As a consequence, we are faced with a lack of transfer and relevance. Students often do not know why they should learn the subject matter. Contextual-based approaches address these problems.

Although contextual-based approaches are quite promising, we do not know exactly what context means and how CS teaching should implement this approach. In the broadest sense, *CS in context* means that CS is linked to subject areas (application domains) outside CS and that it helps students to perceive CS as a meaningful, useful, and helpful subject that is established in outside arenas. But students’ motivation and interests depend on students’ prior experiences. In our research, we investigate students’ computing experiences in order to understand how their interests and motivation for CS were developed.

In the study presented in this paper, we are analysing computing experiences of students majoring in CS-related fields. We expect to find out more about *CS in context* as well as the role of context in providing pathways to CS.

Regarding the content, the paper is organised in three parts:

1. In section 2, we examine related work and discuss what *CS in context* means to CS Education.
2. Based on this discussion, we describe in sections 3 and 4 the research framework on which our study is based. This includes the theoretical background, our research instrument, the analysis procedure, and the participants of our study.
3. Finally, we present the results of the study in section 5.

The paper concludes with section 6 where we discuss the results and open problems.

## 2 CS in Context

Research on broadening participation and interest in CS is often done from a gender perspective (Camp 2002, Cohoon and Aspray 2006, Margolis and Fisher 2002). This work points out the idea of pathways and the importance of context. In the following paragraphs, we examine this aspect in further detail. For further reading about participation and interest in CS, see for example Carter (2006), Peckham, Harlow, Stuart, Silver, Mederer, and Stephenson (2007), Turner and Turner (2005), and Vegso (2005).

Usefulness seems to be a reason why CS should be linked to other disciplines. For many students, particularly female students, the usefulness of CS is not self-evident. Studies in

this field conclude that CS becomes useful when it is linked to other arenas, disciplines, or fields to accomplish non-computer-related goals. Rosser suggests “[using] methods from a variety of fields or interdisciplinary approaches to problem-solving” (Rosser 1990, p. 64). Fisher and Margolis argue too that “[s]ome of the elements of a more contextual approach include early experiences that situate the technology in realistic settings; curricula that exploit the connections between computer science and other disciplines; [...]” (Fisher and Margolis 2002, p. 81). The recommendations for CS teaching correspond to students’ observed requirements: teach CS with interdisciplinary contextual approaches in realistic settings. From a more theoretical perspective of CS Education we ask: what exactly does a contextual approach mean? Lave and Wenger (1991) addressed this question and developed a theory about situated learning, which we consider in the next paragraph.

Situated learning, as suggested by Lave and Wenger (1991), means that learning takes place within the community where the knowledge is used, as opposed to learning in conventional schools that “is predicated on claims that knowledge can be Decontextualized [...]” (Lave and Wenger 1991, p. 40). “In summary, rather than learning by replicating the performances of others or by acquiring knowledge transmitted in instruction, we suggest that learning occurs through centripetal participation in the learning curriculum of the ambient community. Because the place of knowledge is within a community of practice, questions of learning must be addressed within the development cycles of that community [...]” (Lave and Wenger 1991, p. 100). Learning is a process that starts with *legitimate peripheral participation* (LPP) and becomes central in the *community of practice* (CoP). Lave and Wenger developed this theory studying and observing traditional apprenticeship. They claim that LPP can be generally applied to learning. But they do not apply their theory to conventional teaching in schools; they just mention that it is decontextualized.

Guzdial and Tew (2006) analysed Lave and Wenger’s theory about situated learning in order to apply it to CS Education and teaching. They claim that the lack of legitimacy is probably the biggest problem of traditional school teaching. They argue that “[t]he best that we in traditional school can do is to *align* our instruction with students’ perceived community of practice [...]” Lave and Wenger’s theory suggests that students must perceive *some* alignment for learning to occur”. They continue that “[u]sing LPP as our theoretical perspective, we might ask what communities of practice do our majors perceive [...]” (Guzdial and Tew 2006, p. 52). Guzdial and Tew relate two substantial points of alignment: learning activities must be aligned with an external CoP and with students’ purpose and expectations. They argue that to incorporate the notion of a CoP into teaching is not sufficient; the incorporation must be meaningful and realistic for the students. From this argument we can deduce that learning *CS in context* means an alignment between learning and an external CoP, as well as students’ expectations, and this alignment must provide students with a sense-making perspective on the subject matter.

An alignment between learning and an external CoP can be understood as learning in realistic situations with realistic problems. However, beginners' notions of realistic CoP might be wrong or even opposed to CS. Moreover, in a fast-changing field such as CS it is difficult to provide students with realistic situations or problems. Ben-Ari (2004) argues that "what is a real situation will depend decisively on the students' background and future plans, amplifying the tendency towards premature determination of an occupation. It is also clear that it is impossible to present young students with situations that are really 'real' [...]. Especially in a rapidly changing field like CS, the specific content of secondary and even undergraduate education can become rapidly outdated" (Ben-Ari 2004, p. 88).

Ben-Ari points out that the content is not essential. A contextual approach is more than a compelling example, more than demonstrating the use of learning material in an application domain. It is more important to consider the roles people take on in a CoP: "The most important lesson that I draw from analysing situated learning in the context of CSE [Computer Science Education] is the importance of domain knowledge in most of the CoPs that students are likely to join [...]. Curriculum design should be more cognizant of what Shaw calls roles as opposed to content [...]. Situated learning supports her claim that students should choose a specialization that is oriented either to an application area or to CS technical expertise or looking to future managerial responsibility" (Ben-Ari 2004, p. 95).

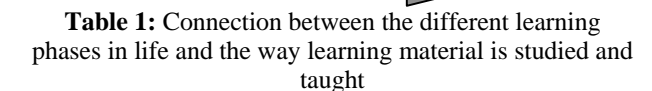
But an effective or useful specialization depends on the students' perspectives: their backgrounds, their future plans, and their prospective roles in a CoP. Contextual approaches claim to provide effective pathways to CS because learning is aligned with students' expectations, their background, their development, and their biography. Therefore, likely more than one pathway exists. Fisher and Margolis (2002) argue that "[w]e need to establish the sense that there are multiple valid ways to 'be in' computer science" (Fisher and Margolis 2002, p. 81). Furthermore, while one pathway is meaningful for some students, it might be meaningless for others.

## 2.1 Summary

We summarize the current discussion about contextual approaches in CS Education. Table 1 shows the problem that contextual approaches refer to. The transition between the different phases is very difficult. The initial phase, where interest and motivation for a subject or area grow, is highly contextualized. During high school, college, and university, students learn subject matter decontextualized. Finally, when they finish their studies, they are faced with a highly contextualized employment or job. CS learning material should be linked to a context in order to provide a better transition from one learning phase to another.

A contextual approach links learning material to activities in realistic settings and CoPs. This approach enhances the quality of teaching and learning because it demonstrates the usefulness of the learning material, and this motivates students. However, until now only vague guidelines have been

Initial phase	High school/ University	Employment/ Job world
Highly contextualized	Decontextualized	Highly contextualized



**Table 1:** Connection between the different learning phases in life and the way learning material is studied and taught

developed for implementing contextual approaches. Therefore the question remains open what kind of context should be chosen in order to effectively implement a contextual approach. Given the current discussion in CS Education, several perspectives are possible:

- **Hypothesis 1.** *A context is effective if it aligns learning material with students' current interests.* The alignment between context and students' interests means that a context must be found that shows students how to use CS in areas the students are currently engaged in. For example, if students are engaged in biology, CS applications and activities in biology will be very effective. If they are not interested in biology, a context from biology will be useless.
- **Hypothesis 2.** *A context is effective if it aligns learning material with examples students can accept as realistic.* Students need to understand how CS is used in practice. Therefore, a context is effective if it provides examples which students accept as realistic and meaningful. For example, if students are able to accept that CS is used in biology, biology will be an effective context, regardless of whether the students are interested in biology.
- **Hypothesis 3.** *A context is effective if it aligns learning material with students' current interest in a CoP.* Depending on the roles members play in their CoP different ways of using CS in an application domain are possible. A context is effective if it supports students in exploring roles they want to play. For example, if students are engaged in biology, a demonstration of the different roles and activities a biologist plays and takes on will be very effective.
- **Hypothesis 4.** *A context is effective if it aligns learning material with realistic examples of possible CoP.* Possible roles are demonstrated to students. Students are introduced to a possible CoP where they can become familiar with different roles and activities in authentic situations. For example, if students can perceive possible future roles in biology, biology will be an effective context, regardless of whether students are currently interested in becoming biologists.

In summary, the question is whether teachers should adjust subject matter to students' interests in other disciplines, or only to realistic problems or settings. To put it differently, learning to solve their prevailing problems by using CS can motivate students, as can learning to solve the problems of 'others' in this way. The other question is whether students

are more interested in contextualized learning materials or in the possible roles, responsibilities and activities such materials provide.

In the following section, we introduce very briefly the research framework that forms the context for the study of this paper. Thereafter, section 4 contains a description of our study's participants and research questions. Finally, in section 5, we present the results of the study with regard to the four hypotheses.

### 3 Research Framework

This study is part of a larger research project that assumes that today's students – whether in K-12 or at university level – enter the CS classroom with preconceptions of CS. The research project investigates students' everyday contexts that provide or influence students' conceptualizations of CS.

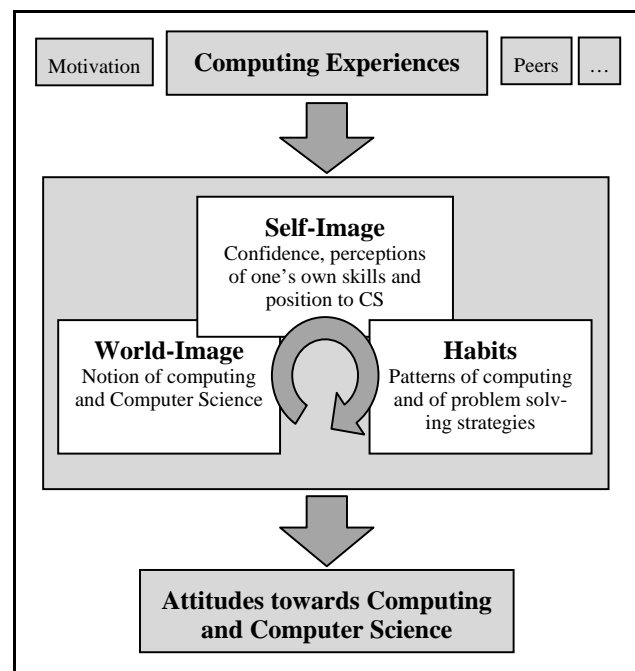
Novice conceptualizations often differ from the scientific concepts. In science teaching, theories of novice conceptualization processes and conceptual change play a major role in understanding and supporting learning. Conceptual change is generally defined as a form of learning that changes an existing understanding. Students already have an understanding (conception, belief, idea, or way of thinking) about the subject, which in educational research is called pre- or misconception. Teaching under terms of conceptual change primarily implies uncovering student preconceptions in order to help students to change their conceptual framework. Conceptual change theories seem to be useful in demonstrating the importance of taking into account students' prior knowledge and incorporating students' everyday contexts into teaching.

Under the terms of conceptual change, learning is not only a cognitive acquisition of knowledge. It also includes and affects all aspects of a student's personality: the student's personal story, self-perception, and view of the world, as well as habits, and learning styles. We assume that conceptions of and beliefs in CS are developed through a learning process which, among other factors, is influenced through computing in formal and informal settings. In our research project at the *Institute of Computer Science, Freie Universität Berlin*, we aim to understand the conceptualization process of CS in order to develop didactical interventions for CS teaching. We want to understand the whole process of computing and its role and impact on students' learning processes in CS. To this end, we have developed a biographical research approach where we survey students' personal computing stories. Our data gathering method provides autobiographical computing narrations in written form, which we call *computer biographies* (Computer-Biographies, Webpage). In the following section, we describe this biographical perspective and the way it is connected to the *contextual approach*.

#### 3.1 Biography as a Method

A computer biography is a story a person tells about his or her computing experiences. Typically, a story is told from a personal, subjective point of view and contains only those

aspects that the author considers to be valuable and important for the story. In particular, when we ask CS majors to write down their own computer biography, they are implicitly triggered to write about those experiences that explain why and how they became CS majors. Such texts usually follow a typical narrative pattern: starting with a beginning such as the first contact with a computer and ending with the current situation, for example, with a happy ending. In between we find important experiences that fostered or constrained their development. We also find information about computing experiences and activities. As computing and CS are closely related (especially for novices), computer biographies reveal information about conceptualisations of CS.



**Figure 1:** The analytical categories self-image, world-image, and habits as specifications of the biographical computing process (Schulte and Knobelsdorf 2007, p. 32)

From a more scientific point of view, computer biographies form a qualitative biographical research design to explore computing experiences and their influence on people's belief systems related to CS. Such a belief system is made up of a person's self-image, world-image, and habits (see Figure 1). The *self-image* includes self-conception and judgement, as well as attitudes regarding the subject's own computer skills and orientation in the computer world. The *world-image* comprehends personal theories and preconceptions about computing and CS. *Habits* comprehend learning strategies, typical performances with the computer and reactions to problems (Schulte and Knobelsdorf 2007, pp. 31).

In the next section, we present our empirical study, which connects the theoretical framework about *context* with the biographical research framework.

## 4 Current study

In the current study presented in this paper, we try to apply the analytical categories world-image, self-image and habits to explore the relevance of context in computer biographies. Referring to the discussion above, a context might be effective if learning material is related to useful applications that students can experience by using the computer. Consequently, analysing students' computing activities should reveal the relevance of contexts. It should also reveal the relevance of roles in a CoP, as roles are determined by the typical activities of a member of a CoP.

We chose a group of students who had just recently enrolled in a CS-related subject in order to focus on students who are dependent on context more than others. We expect bioinformatics students to talk about computing experiences related to biology, bio-technology, or biochemistry; we expect them to want to locate computing in the context of biology. Likewise, we expect mathematics majors to talk about theoretical, logical, and abstract issues concerning CS. Finally, we expect CS education majors to talk about the reasons why they chose to become future CS teachers.

### 4.1 Participants

In order to investigate contextual approaches to CS, in our study we examine computer biographies of students majoring in CS-related fields: bioinformatics majors entering university (25 men, 22 women), mathematics majors with CS as a minor subject (5 men, 5 women) and second-year CS Education majors (19 men, 3 women). The fact that these students have such varied interests makes them a highly interesting study object for our research purpose.

"Bioinformatics is the application of computational tools and computer technologies to model, analyse, store, retrieve, manage, present, and visualize biological data" (Zhang, Lin, Olsen, and Beck 2007, p. 186). Bioinformatics covers biology, bio-chemistry, and CS, and refers to hypotheses 1 and 3 concerning an effective context. Therefore bioinformatics is ideal for students who are interested in all three subjects.

German mathematics majors are obliged to take a minor subject, which very often is CS or physics. CS and mathematics are strongly related to each other, and there is anecdotal evidence that many mathematicians perceive CS as applied mathematics.

In Germany, the subject *CS Education* focuses on teaching CS in high school. Every teacher is obliged to study CS, education, and another major subject such as mathematics, history, sports, or sociology. We hypothesize that these student groups are even more likely to enjoy contextualized teaching approaches than other students because their subject choice shows that they wish to align different areas of their interest. Therefore they should be highly interested in learning how to apply CS in other areas (contexts).

First step	Extraction of basic information needed to build a coding system
Process of data access	Gaining overview about roles and contexts in biographies Finding important topics and structures for first potential codes
First coding	Material coding based on the considerations above
Second step	Material coding
Setting up code system	Development of explicit codes and coding rules based on first coding (activities and periods)
Second coding	Two independent coders coded all material, applying the developed coding rules and codes
Third step	Interpretation of coded results
Relating codes	Relating activities and periods and their interpretation concerning roles and context
In-depth interpretation	Focusing on important activities in decision period

**Table 2:** Analysis and interpretation procedure, the coding steps and their illustration

### 4.2 Analysis Procedure

As described in section 3.1, we connect the theoretical framework for context with our biographical research framework. In order to operationalize context, we investigate activities. Computing activities means uses of the computer in a certain context. When we investigate computing activities we should find out something about students' perspectives on context. Students not only describe their computing activities but also give feedback on how they enjoy them. These personal opinions about students' experiences can express a certain position that refers to a CoP (Community of Practice). A context is also effective if learning material is aligned to a CoP. This alignment is specified in the activities carried out by members of a CoP and in the roles these activities are related to. Furthermore, activities and habits determine roles. Therefore, in the analysis process we focus on activities and related roles.

The analysis was divided into three consecutive steps. In the first step, we read the material in order to get a general

idea of relevant information and to identify potential codes for the coding system. Here, we relied on concepts of coding systems from prior biographical studies. During this process, we observed that the biographies partly resemble each other in their structures.

In the second step, we set up and refined the coding system, and testing it until we had a final result: we coded the material according to the activities described and for three different parts or periods of a computer biography.

In the third step, we examined all activities in their respective periods for frequency, particularities, or length. Thereafter, analysis was narrowed down to focus on more specific aspects of the biographies: the relevance and characteristics of the most important activities with regard to role and context. In Table 2 we summarize these steps.

The results of this analysis and interpretation are presented in the next section.

## 5 Results

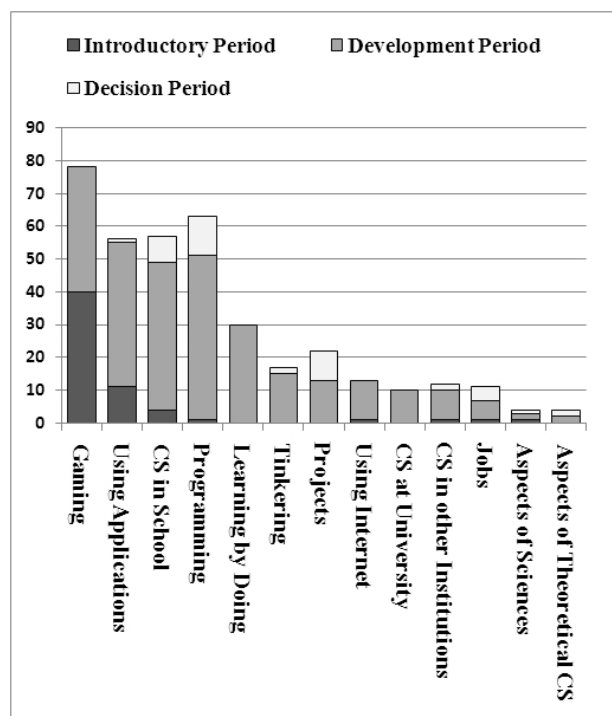
In this section, the results are presented in a way that roughly reflects the analysis procedure. First, we present an overview of activities analysed in the first coding step (section 5.1). Then we describe in section 5.2 how we divide the biographies into different periods. In this second analysis step, we reveal only some hints referring to context and roles. Thereafter in section 5.3, the analysis is narrowed to examine the most important period in more detail.

### 5.1 Results of the First Analysis Step

In 47 biographies of bioinformatics students, we found only four students mentioning aspects of science. Two of them relate computing to biology; the other two combine two different fields of interest (biology and computing). In the other 43 biographies biology or science are not mentioned.

Similar observations can be made in the group of math students. One student mentions his interest in theoretical computer science. We interpret this as an indication of a mathematical context. Two students regard CS as important in order to cope with the course *computational mathematics*. We can also interpret this as an allusion to CS (or computing) in context; probably the interpretation ‘math in context’ would be more accurate here. Overall, three of ten biographies contain indications of a context; however, these indications are rather vague.

Some CS Education majors mention that they found CS at school interesting, but only three students link their computing experiences to their subject choice. One student considers his CS teaching in school as didactically valuable and therefore wants to study CS Education himself. Another biography claims the opposite: because of negative impressions of the teaching quality in school, the student wants to become a CS teacher and do a better job. The third biography claims that apparently CS teachers are not obliged to program so much. In addition, several students mention that they enjoyed their CS courses at school without explicitly linking these experiences to their subject



**Figure 2:** Activities and Periods (the y-axis gives the total nomination number of each activity in all biographies surveyed)

choice. We find some explicit references to subject choice, but not as many as expected.

### 5.2 Activities and Periods

The biographies are structured like a narration with a beginning, a climax, and a (happy) ending. Our analysis of these patterns, together with related activities, reveals three periods in the biography. We call the first part the *introductory period*. This period starts with the first computer contact. It contains experiences and situations that are initiated by accident or by others. After the introductory period, a *period of development* begins. This period is characterized by purposeful experiences where students develop their interests. During this development a *decision period* might take place. A decision period contains important experiences that are decisive for the future. Such experiences are described in detail and are much longer than other experiences and events in the biography.

For all periods we examine the activities and experiences the students have had with the computer. We also examine how the students perceive their own activities and experiences. This helps us to identify which type of activity was experienced in which period and had most influence on the students. Figure 2 shows the types of activity according to the different periods.

#### 5.2.1 Introductory Period

The introductory period starts with the first computer contact and ends when a certain development, which does not

happen accidentally, can be observed. The biographies typically start with a description of the first encounter with computer, the surrounding events, and the experiences of the student. The introductory period contains experiences and situations that are initiated by accident or by others. The transition to the next period, the period of development, is triggered mostly by the fascination for computers and the challenge they offer to create something by oneself. This transition is mostly smooth. Sometimes the change is indicated by a new paragraph or keywords that mark the beginning of something new in the story.

In 77 biographies we found 63 introductory periods. The 14 biographies without this period were either very short (1-2 sentences) or started directly with the period of development. The students typically first used a computer between the ages of eight and eleven. Almost always the first computing activity is gaming. Most students mention gaming (40 times) as their major activity in this period, considering it as interesting (9) and fun (7). The activity most frequently mentioned after gaming is the use of applications (11). Some students experience the first computer activities at school (4), such as the use of applications. But the majority encounter the computer outside school, mostly at home. Parents, brothers and sisters, other family members, or peers help them and explain the computer usage. Usually this *supportive* person is male. Very often students mention family members' professions when they belong to computer science or engineering, probably in order to explain why they themselves are interested in CS, too.

Our overall impression of the introductory period is that all of them are very similar and also very short in comparison to the time period they describe. The central activity in this period is gaming. After the first coding process we observed that the introduction to computing happens mainly by chance. Therefore, we define the introductory period as being accidental. Indeed, in nearly all biographies the first contact was accidental. Only in some biographies can a kind of purposeful introduction to computing be detected; these biographies were coded as beginning directly with the development period. With regard to roles and context in this introductory period we summarise:

<b>Dominant context</b>	leisure time
<b>Typical activity</b>	using applications (games)
<b>Typical role</b>	beginner, learning, passive, introductory help by others, typically male person

### 5.2.2 Period of Development

After the introductory period, we typically find a period that is marked by a certain development. The period of development is characterized by purposeful experiences. Students guide their development instead of being guided by others or by accident as in the introductory period. In 77 biographies we found 81 periods of development, because in four biographies this period was interrupted by the deci-

sion period and therefore contained two periods of development.

In the period of development many different activities are described. Sometimes they are just enumerated. Almost all school experiences and activities take place in this period. The programming activity occurs in this period. These are the activities that were mentioned the most: programming (50 times), CS in school (44), using applications (43), gaming (37), and tinkering – especially with hardware (15).

Many students describe a great fascination for computing and a great interest in exploring and learning more about computers. Interests, satisfaction, or fascinations were mentioned 67 times in this period. For a better illustration we quote text examples from three biographies:

*"[...] my interest grew again because of the seemingly never ending possibilities with this tool."* [101B1986m]<sup>2</sup>

*"The possibilities the computer offers are uncountable."* [92B1988m]

*"At this time I was inspired by the creative possibilities of computers."* [87B1985m]

But this period also contains ups and downs in which students lose their interest in CS for a while or are bored by it. We found text samples in which students described disappointing (16), boring (8), and uninteresting situations (6) without fun (2). Problems, however, are perceived as challenges to learn and to explore more. *Learning by doing* activities were mentioned in 30 biographies, but 27 times this was in the period of development. Learning is perceived as widening the possibilities and skills and therefore rewarding. New experiences and activities trigger students to learn more and to gain more new experiences. Furthermore, the students describe self-confident and independent habits regarding solving computer problems and learning programming. They also appear to be confident (especially women) of coping with a major in the subject.

These learning habits are common to most CS majors we have investigated so far in previous studies. But unlike many CS majors, the students majoring in CS-related fields do not divide computing into *using* and *designing*, in which the latter is superior and reserved for computer scientists; the computing experiences seem to have less impact on their self-image. In comparison, it seems that computing experiences and self-perceived computing skills are very closely connected to the self-image of CS majors; their identity seems to rely quite heavily on the self-image in computing, whereas it does not for the students majoring in CS-related fields.

Our overall impression of the period of development is that it starts with activities students are interested in. They re-

<sup>2</sup> This code identifies a biography. The first number is the biography number; the upper-case letters refer to students' major subject and are B for bioinformatics, CSE for CS Education, or M for maths; the second number is the student's year of birth; the lower-case letter is f for female or m for male.

gard these activities as an opportunity to explore the computer's possibilities and to learn more about it, for example, programming as a starting point. Students show an addiction to exploring the computer, learning by doing. This improves their self-image and provides a positive world-image (computing is fun, creative, self-exploring). The students start as users and become designers without seeing a dichotomy between these two aspects of computing. Becoming a designer is not a change from one group to another but a development from simple to complex activities. With regard to roles and context in the period of development we summarise:

<b>Other contexts</b>	school, learning, homework
<b>Typical activity</b>	more applications, also programming as a new computing activity
<b>Typical role</b>	roles are changing: advanced learner, more active, problem-solving and exploring

### 5.2.3 Decision Period

The decision period is typically characterized by a special experience that is decisive for the future. This experience is described in more detail than other experiences and events in the biography. We determined that only one decision period can take place. This period can be detected by looking for formulations containing superlatives, differentiations, and keywords such as 'everything changes', 'special', 'other', 'new', 'important'... A new paragraph can also indicate a break or change in life. Experiences that are described in detail or differ from the chronological description of the rest of the story are also evidence for the decision period. The decision period is characterized by a high degree of students' self-determination. Ups and downs are no more mentioned. Instead, the students restrict themselves to aspects that are important or 'decisive' for their biography. There was no decision period that was not preceded by a period of development. The decision period is like a consolidating climax of the period of development. Therefore we should rather have named this period the *period of consolidation or summarization*.

In the period of development many different activities were described. In the decision period the students concentrate on a few important activities such as programming or project activities. Internet, tinkering, and games are not mentioned any more, or are no longer seen as important. In the 20 decision periods we analysed, the following activities are mentioned most often: programming (12), CS in school (8), doing projects (8), and doing a student job (4). These activities are rated as being interesting (9), fun (6), and satisfactory (4).

Altogether, we found 20 decision periods in 77 biographies. We observed that the biographies without a decision period very often end with a single paragraph or sentence in which the students describe a decision. In this final part the students explain why they decided to major in their subject:

*"Because computer science was too dry for me and I was also interested in biology and chemistry, I chose to major in bioinformatics."* [109B1986m]

*"For a long time I wasn't interested in computer science, but during my PhD in medicine I realized that in our research group projects were more successful when they were done with computers."* [107B1971m]

*"I became curious and decided to major in this subject."* [140M1986f]

But these text samples are not (!) decision periods. The decision period is a biographically established decision, not an explanation of subject choice. The decision period is characterized by a consolidation of interest in CS, and we assume that this decision is permanent or consistent for a while. With regard to roles and context in the decision period we summarise:

<b>Other contexts</b>	school, projects, jobs
<b>Typical activity</b>	focusing on a single computing activity
<b>Typical role</b>	role changes toward expert: competent usage, perceived as satisfying

## 5.3 Interpretation of Activities in the Decision Period

As we have seen in section 5.2.3, in the decision period one type of activity often becomes central, and is a major factor for the decision to study CS. In this section, we explore these important activities in more detail, and interpret them with respect to context and related roles. The activities to be analysed are: programming, learning CS at school, and jobs and projects.

In a decision period the most prominent activity (the one most often mentioned and described) is programming. Programming is closely related to CS at school and to projects. CS courses at school can be considered as positive, although at the same time students claim to have had to learn programming on their own, or that the teacher was often not able to explain the subject matter. Therefore, the role of CS in high school is critical: while teachers are regarded as incompetent, they introduce many students to programming and are able to increase their interest, motivation, and programming skills; in many cases, programming influences the subject choice and is explicitly mentioned:

*"I did a lot of programming in my leisure time; it's the main reason for my decision to study [CS Education]"* [16CSE1981m]

At first sight, no links between programming and context or roles as well as statements referring to specific contexts can be found. Only one student is exceptional. He is a student of Bioinformatics, argues for the importance of free software, uses only such (like FreeBSD), is a member of the free software foundation Europe, and programs in C++ for



FreeBSD. These activities are an example of participation in a CoP. However, the link to Bioinformatics seems only marginal. Apparently, the aspect of fun is an important reason for the popularity of programming. Students enjoy to solve a task or problem, thereby creating a product or at least an artefact. The next text sample illustrates this:

*“Programming [in CS class at school] was always fun because in most cases a running piece of software was produced.” [14CSE1983m]*

So, programming positively influences students’ self-image in two ways. First, it is fun and students want to engage in computing and CS. Second, this experience demonstrates students that CS is something they are good at:

*“As my CS course at school needed a month to cover only if/else-constructs in Pascal, I knew it is my task to study it on my own. During fall holidays, I took a book of programming and studied learning material of the whole school year [in two weeks]. However, the CS course at school wasn’t useless, because the teacher helped me to improve my programming skills, and gave me more difficult programming tasks.” [135M1987m]*

Other biographies refer to programming contests, which gave students the opportunity to overcome self-consciousness, thus positively influencing their self-image. In several biographies students mention that it was rewarding to compete with other students during programming. Another example may illustrate this aspect:

*“My interest for programming rose when my CS course at school started. The whole course was about programming [...] At home I added additional functionality to the programs, which was beyond the required tasks.” [09CSE1985m]*

We analysed project activities because projects are probably closely connected to roles and contexts – at least projects are embedded in an application area. The application areas mentioned are: school homepages, building and configuring a network in the school building, robotics, and other application areas described only vaguely, such as web-related projects. The last cited biography above, for example, continues with a description of two interesting projects. One project dealt with the development of software for Kephra robots in cooperation with a local university. The other project was set up in cooperation with a local company where the students participated in an R&D project involving adaptable lights for cars. Such projects often start in CS courses at school. But we did not find projects that were confined to a CS course. There were examples of cooperation between school and other institutions, or between the CS course and another subject.

When biographies contain descriptions of projects, they refer to certain contexts, but these seem to be important only as triggers for programming activities such as robotics or an interactive homepage for the school. The functionality of CS at school, in projects, and in jobs seems to provide incentives to start programming.

Altogether we found only a few biographies mentioning projects in a CS course at school. This aspect is somewhat

confusing, as projects are of great importance in the German tradition of CS teaching at high school. For example, in Berlin, where many of the participants of the study supposedly attended high school, an entire school semester should be devoted to a project. Perhaps fewer projects are done in schools than are required by the curriculum, or these projects are too irrelevant for the students to include them in their biographies. Overall, projects and jobs are often described as motivating, fun and relevant, but the description often remains superficial, without details.

So far, we have discussed the significance of context as an application area. Another approach focuses more on roles than on context. Now we re-examine the above-mentioned activities from the perspective of roles.

We found only sparse allusions to roles in the biographies. For example:

*“I decided to study CS for teaching at high schools because I didn’t want to become a programmer.” [29CSE1983m]*

*“As both CS teachers were incredibly incompetent, I became motivated to show that it is possible to teach the subject matter in a reasonable way and to engage also those [pupils] who are marginally interested in [CS]” [10CSE1975m]*

The third allusion to roles was already mentioned above: programming as an activity to contribute to Free Software. All of these allusions are rather vague. Overall we did not find sufficient information about our hypotheses three and four. Instead, it seems as if none of the hypotheses could be confirmed; perhaps roles are not so important, after all. In the next section we discuss this aspect in further detail.

## 6 Discussion

In this study we wanted to find out more about the importance and characteristics of context and the student perspective on context. Based on related work, we supposed that contextual approaches were important and therefore somehow visible in computer biographies of students majoring in CS-related fields.

Referring to the four hypotheses from section 2.1, we have some interpretations based on the results of our study:

**Hypothesis 1.** *A context is effective if it aligns learning material with students’ current interests.*

**Hypothesis 2.** *A context is effective if it aligns learning material with examples the students can accept as realistic.*

**Hypothesis 3.** *A context is effective if it aligns learning material with students’ current interest in a CoP.*

**Hypothesis 4.** *A context is effective if it aligns learning material with realistic examples of possible CoP.*

Our results reveal that the students do not explicitly connect computing with usefulness. We have not found any substantial data showing that the participants link computing to a context – either with current interests or with realistic examples (hypotheses 1 and 2). Computing is not closely related to specific application areas; although the bioinfor-

matics students obviously aim at career prospects that place CS in the context of biology, they do not explicitly ask to learn CS in a biological context. They are simply interested in the general possibilities offered by computing. It seems that the activity itself is enough motivation, and no additional incentives are needed.

With regard to the importance of context to providing pathways into the CS field, we found that rewarding and motivating activities are important. We cannot say whether, for example, gaming is important as a first motivating computing experience, but it is typical and somehow it seems to spark interest in exploring more related experiences. One of these can eventually be programming – in most cases, triggered by external factors, such as the opportunity to enrol in a CS course.

## 6.1 Discussion of Methods

Nevertheless, we found surprisingly little information about context and activities related to roles. Some possible reasons are listed below.

1. We asked the wrong questions (data gathering method was wrong).
2. We asked the wrong students (the population should be different).
3. We asked too early (data gathering should be later in the students' studies).
4. We asked too late (data gathering should be earlier, in high school).
5. We analysed data wrongly (data analysis procedure was wrong).
6. We did everything right, but contextual approaches are simply not so relevant for students.

The use of computer biographies certainly has some implications. This method is based upon texts in which participants describe their experiences with computers. These are individual memories of past activities at the computer. This empirical method claims that such texts are written narrations and that they contain a story line. Not only isolated facts and descriptions are given, but also additional information such as feelings or opinions. For example, we found comments such as:

*"In high school I chose to focus on biology as my main subject and became enthralled for the genetic code; to me, it is the perfect programming language."* [108B1985m]

While this text sample is a reference to programming, it is also an example of additional information included in a biography that extends the focus of describing computing. Such additional information is quite common.

References to context and roles are typical examples of such *additional information*. Having found hardly any information on contexts, we have to ask whether this issue is caused by the instrument used (see reason 1). Do computer biographies contain the additional information as described? The answer is: yes they do. This can be seen, for

example, in section 5.2, where additional information concerning activities is summarized.

While biographies contain additional information in general, perhaps they do not contain information about context and roles. Context is somehow different from other additional pieces of information like feelings, peers, problem-solving strategies, characterizations of the computer and of CS. We do not see such a qualitative difference between references to context/roles and references to other additional information in computer biographies. Some information as to the context is even given directly (see section 5.1). In summary, we do not think that the results are caused by an incorrect empirical approach.

Another reason for the result might be the population studied (see reason 2). Perhaps the students we asked to write their computer biographies were somehow not able to include information about context. We deliberately focused on students who were not majoring in CS but studying CS in context with another subject (teaching, math, and biology). The argument was that students who chose to study not just CS but 'CS plus something else' were even more likely to acknowledge contexts. Given the results, this might be wrong, but the reason is unclear.

Similarly, we possibly asked them at the wrong time (see reasons 3 and 4); if we had asked them earlier or later in their lives, the results might have been different. A biographical narration is written from the specific perspective of a person at the actual point in the present. This perspective organizes how the past is conceptualized. By changing this point of view, we expect the story told to be different. In fact, this aspect is the reason why computer biographies are not objective descriptions of facts, but instead contain a richness of implicit information (what we called *additional information* above). We surveyed students who had just entered university (math and bioinformatics group) and students who had already studied for several semesters (CS teacher group). It may be that freshmen indeed focus more on the subject they are about to engage in. Therefore they focus in their computer biographies on the issue CS, because we asked them from a CS perspective, on their first day in university, and during a CS course, to write their computer-biography.

We indeed have the impression that biographies from students who intend to become CS teachers contain more information about context. On the other hand, this might be due to the fact that the students have a more precise perspective of their future roles in their jobs as teachers. In conclusion, we admit that perhaps we would have found a different picture if we had asked this population at another point in their lives. Nevertheless, and this is the crucial point, this does not explain why, at this point, they wrote so little about context.

Another explanation could be that we did not use the right analysis procedure (see reason 5). We chose to analyse the described activities in the biographies in order to find out more about context. We concentrated on activities because biographies comprise mainly activities. Based on the related work in section 2, we concluded that activities would be the

most obvious aspect to look for in order to gain information about context. Perhaps another approach is possible and our operationalization of context must be revised.

## 6.2 Conclusion

In their biographies, the students write about their computing experiences. The stories focus on computing and not on context. This could be the reason why we did not find much information related to context. On the other hand, biographies are likely to include many additional thoughts and comments – whatever kind of information the author thinks of.

Our interpretation of the results is that students do not perceive roles and contexts as being important – otherwise we would have found more information about it (see reason 6 from section 6.1). We interpret the results of the study as a hint that even freshmen who enrol in CS-related fields do not link CS to contexts. In general, they simply do not know how CS is embedded in contexts; how, for example, CS is used to solve biological problems. We had the impression that students relate their computing activities with the context *computing*, and their computing activities were limited to the computing context. In addition, they were not able to relate their own computing experiences to contexts, and rather perceive computing as a kind of closed world, detached from its surroundings.

What is the consequence of this interpretation when it comes to teaching? Can we conclude that teaching in context is not important because context is presumably not a central factor in the awareness of students? On the contrary, if the interpretation of the results in the paragraph above is correct, then the students do not perceive CS *in context* although they study CS-related fields. A conclusion therefore is that students perhaps choose to study CS-related fields because their interest in computers, computing and CS was not enough to perceive a future Community of Practice that they could wish to belong to.

From our recent study we know that CS majors tend to have a fixation on computers. Perhaps the students majoring in CS-related fields tended to reproduce this belief system as well. Therefore they chose a CS-related subject because they were not identifying with this 'fixation on computers'. But then this conclusion supports the belief that students have a very narrow perspective of CS; it is then even more important to use contextual approaches in teaching. Altogether, we conclude that more discussions and studies are necessary to explore and understand students' perspectives on CS and the promising contextual approach in CS teaching.

## References

Beck, J., Buckner, B. and Nikolova, O. (2007): Using Interdisciplinary Bioinformatics Undergraduate Research to Recruit and Retain Computer Science Students. *Proceedings of the 38th SIGCSE technical symposium on Computer science education: SIGSCE '07*, New York, USA: 358-361, ACM Press.

Ben-Ari, M. (2004): Situated Learning in Computer Science Education. *Computer Science Education* **14**(4): 85-100.

Camp, T. (2002): The Incredible Shrinking Pipeline. *ACM SIGCSE Bulletin* **34**(2): 129-134.

Carter, L. (2006): Why Students with an Apparent Aptitude for Computer Science Don't Choose to Major in Computer Science. *Proceedings of the 37th SIGCSE technical symposium on Computer science education: SIGCSE '06*, New York, USA: 27-31, ACM Press.

Cohoon, J.M. and Aspray, W. (2006): *Women and Information Technology: Research on Underrepresentation*. Cambridge, Mass., MIT Press.

Computer-Biographies: Webpage of the research project. <https://www.inf.fu-berlin.de/w/DDI/ComputerBiographies>

Fisher, A. and Margolis, J. (2002): Unlocking the Clubhouse: The Carnegie Mellon Experience. *ACM SIGCSE Bulletin* **34**(2): 79-83.

Gilbert, J.K. (2006): On the Nature of "Context" in Chemical Education. *International Journal of Science Education* **28**(9): 957-976.

Guzdial, M. and Tew, A.E. (2006): Imagineering Inauthentic Legitimate Peripheral Participation: An Instructional Design Approach for Motivating Computing Education. *Proceedings of the 2d International Computing Education Research Workshop: ICER '06*, New York, USA: 51-58, ACM Press.

Knobelsdorf, M. and Schulte, C. (2005): Computer Biographies – A Biographical Research Perspective on Computer Usage and Attitudes towards Informatics. *Proceedings of the 5th Koli Calling Conference on Computer Science Education: Koli Calling '05*, Turku, Finland: 139-158, TUCS General Publication.

Lave, J. and Wenger, E. (1991): *Situated learning: Legitimate peripheral participation*. Cambridge, Cambridge Univ. Press.

Margolis, J. and Fisher, A. (2002): *Unlocking the Clubhouse: Women in Computing*. Cambridge, Mass., MIT Press.

Peckham, J., Harlow, L.L., Stuart, D.A., Silver, B., Mederer, H. and Stephenson, P.D. (2007): Broadening Participation in Computing: Issues and Challenges. *Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education: ITiCSE '07*, New York, USA: 9-13, ACM Press.

Rosser, S. (1990): *Female Friendly Science: Applying Women's Studies Methods and Theories to Attract Students*. New York, USA, Pergamon Press.

Schulte, C. and Knobelsdorf, M. (2007): Attitudes towards Computer Science-Computing Experiences as a Starting Point and Barrier to Computer Science. *Proceedings of the 3rd International Computing Education Research Workshop: ICER '07*, New York, USA: 27-38, ACM Press.

Turner, E.H. and Turner, R.M. (2005): Teaching entering students to think like computer scientists. *Proceedings of the 36th SIGCSE technical symposium on Computer science education: SIGCSE '05*, New York, USA: 307-311, ACM Press.

Vegso, J. (2005): Interest in CS as a Major Drops Among Incoming Freshmen. *Computing Research News* **17**(3): 126-140.

Zhang, M., Lin, C.-C., Olsen, G. and Beck, B. (2007): A Bioinformatics Track with Outreach Components. *Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education: ITiCSE '07*, New York, USA: 186-190, ACM Press.