Understanding IT and gender

Tone Bratteteig, Department of Informatics, University of Oslo

Information Technology (IT) is male-dominated, both as a profession and an area of work and as a discipline and field of study (Adam 1995; Klawe & Levenson 1995). This is nothing new: IT has been male-dominated for a number of years (Bratteteig & Verne 1997; Kvande & Rasmussen 1989). From time to time new initiatives aimed at increasing the number of women in computing are made (e.g. Camp 1995), but they do not seem to have the intended effects. The explanations for this can be many and may have less to do with the technology or the women, cf. the high percentage of women in technical areas in Malaysia (Mellström 2003). It is, however, interesting to take a closer look at IT and gender for adding pieces to the explanations of why women seem to not choose IT. In this paper I concentrate on the IT side, aiming to look at gender from the inside of informatics.

Looking for gender aspects in informatics

Informatics is a broad discipline concerned with improving computer performances in hardware and software, as well as in applications tailored for particular functional areas. Informatics also includes human-computer interaction and information systems analysis and design; both are areas that include socio-cultural knowledges and skills. The focus is also here on the computer – the IT – and this makes it difficult to both see and discuss gender aspects (Bratteteig & Verne 1997). We will end up with questions like: Is software gendered? Will a female programmer write a piece of code differently – how – and why? What difference would it make? Are computer systems gendered? How can you tell? How could they be made differently, and what difference would that make? Do we want that?

In order to explore gender aspects of computer science, Bratteteig and Verne (1997) suggest analyzing the field based on Harding (1986)'s 5 research programs for gender studies in the sciences:

1) Equity studies: the fact that women in informatics are outnumbered by men in Norway and many other Western countries,

2) Studies of the uses and abuses of science and technology: for IT and informatics such studies are normally carried out by social scientists, and studies of technology (ab)uses in female-dominated areas are taken to represent a feminist position (Cockburn 1983; 1985; Waldén 1994),

3) Critiques of the existence of pure science: critique of the neutrality and objectivity of technology has a long tradition in Scandinavia (Nygaard 1986; Bjerknes & Bratteteig 1988; Bratteteig 2004). Studies of gendered informatics practices are also relevant here (Mörtberg 1997; Fletcher 1994),

4) Studies to reveal social, symbolic and structural meanings: in informatics, the culture of hackers and engineering (Turkle 1984; Hacker 1989; Wajkman 1994; Håpnes & Rasmussen 1991; Bratteteig 2002)

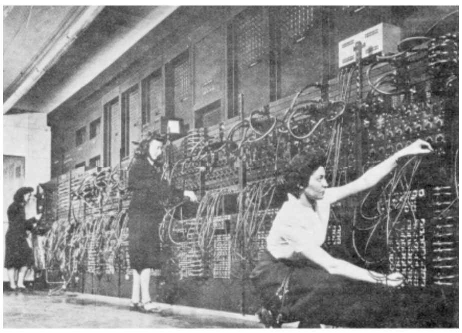
5) Epistemological inquiries to establish alternative understandings of knowledge: discussing whether understanding of IT in informatics can be different from a different perspective (Bratteteig and Verne 1997).

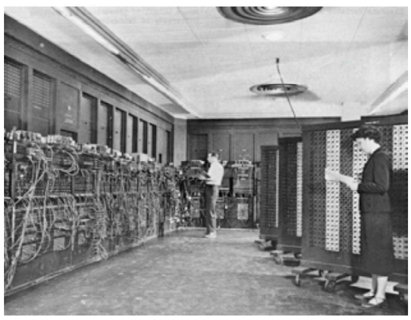
These are all interesting discussions. In this paper I will mainly focus on one aspect: the epistemological, because I think this has been missing from the debates. Epistemology is “the study or theory of the nature and grounds of knowledge especially with reference to its limits and validity” (Webster 2008). What do we mean when we talk about IT? Are there different ways of seeing knowledge in informatics?

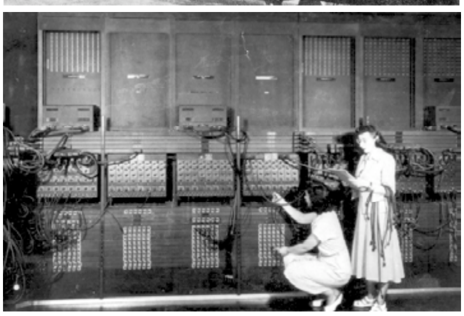
Who does IT?

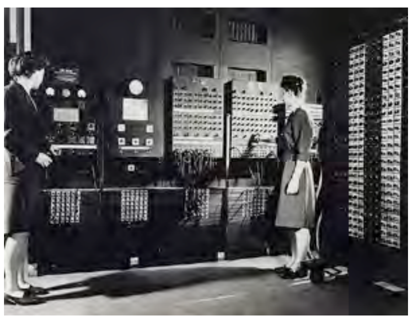
As a prelude to the epistemology discussion I would like to spend a few moments on equity (# 1), pointing to the fact that there are many women in computer science and has always been (Greenbaum 1976). There are of course more men than women, but there have been women in computing from the very start. Computing as a discipline was established in the 1970s, after some years of research on computing in engineering, physics and mathematics. The Department of Informatics at The University of Oslo, for example, was established in 1977 by researchers from The Department of Mathematics (numerical analysis, operational analysis and mathematical verification of programs) and The Department of Physics (signal processing and electronics). The new Department of Informatics aimed at a broad approach to informatics as a constructive discipline based on “studies of information processes in nature and society” (Nygaard 1986) and included courses about the social and political aspects of computing in the curriculum.

The very first programmers were mostly women. Below are pictures of the programmers in the ENIAC team. The ENIAC: *Electronic Numerical Integrator And Computer* was the first electronic computer used for general purposes al-









though it was made for the US Army's Ballistic Research Lab in 1946. It was huge: it was 2,6 x 0,9 x 26 m and weighed 27 tons, and consisted of 17,468 vacuum tubes, 7,200 crystal diodes, 1,500 relays, 70,000 resistors, 10,000 capacitors and approx. 5 million hand-soldered joints (Goldstine & Goldstine 1982). A team of six women did most of the programming of ENIAC: Kay McNulty, Betty Jennings, Betty Snyder, Marlyn Wescoff, Fran Bilas and Ruth Lichterman (see figures above). Programming at that time meant moving dials and cables manually. The team worked together with Adele Goldstine, who wrote the complete technical description of the ENIAC.

Another, even earlier women was Ada Lovelace (see picture below), who is known for writing the description of the very first computer. Ada Lovelace is recognized for writing the first programs for Charles Babbage's mechanical Analyti-

cal Engine in 1843. She was the first to understand the possibilities of the analytical engine, not yet possible to build.

I will also mention Rear Admiral Grace Murray Hopper (1906-1992) (see picture above). Grace Hopper invented the first compiler for a computer language (a compiler translates a computer program text to machine code): the A compiler in the early 1950s while working on UNIVAC I. She is even more well-known for her work on the programming language COBOL: Common Business-Oriented Language and its complier. She was active working until her death in 1992. There are many stories about her and she is claimed to be the originator of the term "bug": while she was working on a Mark II computer at Harvard University, her associates discovered a moth stuck in a relay, impeding operation. Hopper's commented that removing the moth was "debugging" the system.

My last pictures (below) show my colleague Christina Mörtberg operating a computer system in Stockholm in 1972, starting the computer (left) and controlling the program (right). The illustrate that during several decades computers were really big machines and that today's computers – the iPod, the mobile phone, the hand-held, mobile systems – may appeal to other people and other interests.

There are women in all fields of informatics, but more women in multidisciplinary areas such as project management, user training, organizational change, design. These areas have lower wages and lower status – similar to the status of other fields concerned with clients and users (health care, schools). This may be changing as we see more technical development outsourced to countries with less expensive labour while user-oriented work cannot easily be outsourced. Cultural and social skills are being recognized as important as a prerequisite for designing the right solution and understanding the problem to be solved.

What is IT?

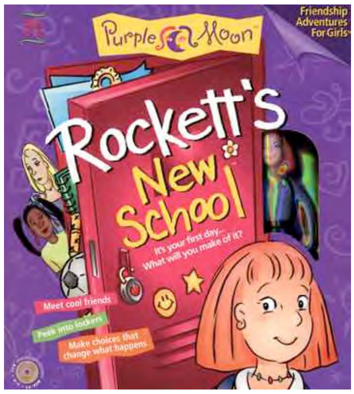
The question about what IT is and what is knowledge about IT – the epistemology of IT – is not as simple as it may seem. IT – information technology – is defined as the study, design, development, implementation, support or management of computer-based information systems, particularly software applications and computer hardware. The computer is at the centre of attention. We have seen images of the computer as a box standing by itself, as the big computers since the ENIAC above, to micro computers like the Norwegian NORD machines, to the small and personal computers in the 1980s like the Macintosh (first made by Apple in 1984). Today, however, computers are embedded in many everyday things, like cars and washing machines: the computer is not only a separate piece of machinery, it is embedded in other things, adding to them some of the characteristics of computing.

Computer characteristics

The computer has some characteristics that are maintained through generations of technical development (Bratteteig 2008). The first and most important characteristic of computers is automation: computers do things; they are processes performing the operations described in the code, transforming input to output according to specified rules. Programs are descriptions of (structures for) processes. Computers compute for you, but they can also do other things, which lead us to the second important characteristics: abstraction. Computers are abstractions all the way from the difference in voltage variations represented binary as ones and zeroes to the printer icon on your screen representing a series of commands involving many pieces of machinery (the computer as well as the printer) enabling you to get ink marks on paper representing what was before light on your screen (Dourish 2001). At the core of computing is the work of modelling and representing parts of the world as abstractions and categorizations, simplifying the real world complexity. A complex human being may be represented as a patient in the hospital system, by numbers of body temperature, body fluids, test results etc. and by a diagnosis. The categories in an information system are invented for a purpose (by someone, for someone) to support use activity. The category of a diagnosis constitutes the basis for treatment and care, for the physical localization of the patient in a ward, in a bed, and even for the social welfare (which makes not getting a diagnosis a problem, see Bowker & Star 1999).

The third important characteristic of computer systems is that it is distributed: data can be distributed simultaneously in endless copies (Bratteteig 2008). The abstractions are de-contextualized as representations in the computer and re-contextualized as they are given meaning in a particular context (like 41 as a ℃ temperature means fever).

The models and categories in computer systems are designed because they are possible to make: designers design what they can imagine (Mörtberg 2001) or what is possible or easy to build (Bratteteig 2004). Gee (2003) claims that the fact that most computer games are shooting games are because this is easy to program. Games especially made for girls are rare (Huff & Cooper 1987). An alternative game for girls 8-14 years old was *Rockett's New School*, a sort of a "visual novel" where you engage in a strategic game with social skills made by Purple Moon in 1997. Brenda Laurel and her colleagues interviewed 1100 children and 500 parents about their favourite things, translating these ideas into the game play for Rockett's New School (Laurel 2001).



Digital artefact characteristics

The computer as an artefact can be characterized through its

- function: its usefulness – toolness – with respect to a human activity
- meaning: its symbolic value given to it in a cultural and social context
- communication of the function and meaning through form and structure (Bratteteig 2002)

We understand how a concrete, material artefact is part of human activities by analyzing its function and meaning in that activity. The function of the artefact has to do with the activities it is designed to be used in; the activities "carry with them an intention of what those objects will do and how they will be perceived and used." (Winograd 1996: xv). The functions and the way these are communicated are often referred to as the "affordances" of an artefact. It is easier to communicate how to use an artefact for designers who are themselves members of the target group' culture and language. For accidental users who are not members of the target group, the cultural codes may not communicate the functionality very well: even if most technology aims at enabling human beings and reducing disabilities by extending our muscular or memory powers, some artefacts contribute to further disabling some social groups (Bratteteig 2002).

The artefactual meaning comes from interpreting its form (or its very existence) as a sign or symbol within a particular culture. Consumers in Western societies are presented with a range of very similar products that do about the same thing, the main difference between them are the different cultural signals they give (often symbolized by differences in price). Choosing one brand before another has more to do with its cultural meaning than with its function.

Function and meaning of an artefact can only be understood as part of a use context. Use happens as purposeful activity, as work, learning, everyday life. The artefacts that are (made) part of the activity contribute to—and receive—function and meaning through this. Human activity takes place in a larger context, in a group, an organization, a society/culture, and can be analyzed at all levels – and between them. Human activity contributes to societal change and technology development. Analyses of these processes need to consider power (decisions), meaning (symbolic communication), activities (usefulness), change processes (use over time), etc. – as would any socio-cultural analyses of human activities. Looking for gender aspects will address many of these dimensions and levels.

What do informatics people know?

In addressing epistemology it makes sense to discuss what informatics people know: what distinguish informatics knowledge from other knowledges. Informatics is about creating design ideas and realizing them in digital form, i.e. utilizing the characteristics of computers mentioned above: automation, representation and distribution.

Automation is addressed by designing structure(s) for processes, making procedures, routines and recipes. Processes that involve interaction with human beings or other machinery need dialogues or sequences of operation and responses to predefined sets of input. Automation is about delegating some parts of a process or activity to some machinery (an automaton), distributing the activity over people and technology. In order to enable a distribution of the activity, some tasks and task flows may be modified or changed. It is an art to design such changes so that they fit with the activity they transform.

Representation involves translations: simplifications and categorizations, made by means of a computer-oriented language and transformed into executable code. Representations can re-present anything that can act as objects of work (or other activity) and should therefore be understandable to those doing that work. Representations thus involve an ability to understand the language, the context and the logic of the users and to translate that into computer code. A truly inter- & multi-disciplinary understanding.

Distribution presupposes addressing multiple contexts and thus involves developing an empathy with unknown “readers”. The re-presentation and re-arranging of work for a context means interpreting the users and deciding what constitutes information to them, figuring out "the difference that makes a difference" (Bateson 1972).

Last, but not least, informatics people know how to make the artefact: they can program. They can "see" the process unfold as they read the program text – just as the musician can "hear" the music as s/he reads the score or the car repair man can recognize a piece of metal as the part he needs to mend a broken car (Harper 1987). Informatics people work with processes and abstractions – both are ephemeral and difficult to grasp. Furthermore, computers usually (are designed to) hide the complexity that give meaning to the abstractions, making the underlying (computer) logic consciously difficult to get at but easier to use. The basis for informatics knowledge is the building of computers, but the skilled informatics people can see the computer as a process in a larger context.

What should users know?

This tour into the informatics field: what is IT and what does it mean to know IT, should make a good basis for returning to the users of technology and ask what the users should know about the ITs they use and are made dependent of. Digital literacy should include literacy skills that address not only the ability to read and write, but also to

- interpret the representation (what kind of information is it)
- to understand the language (and how is it expressed)
- to recognize the production of the information (by whom)
- to recognize the intended audience (to whom – and why these) (Buckingham 2006)

Literacy does not equal literate: to read and write does not necessarily mean to know your literature – the authors, styles, genres etc. Young users who are "good at IT" often do not develop their skills over the years, possibly because their knowledge is rather shallow (Livingstone 2004; in press). A thorough study of users of an administrative system revealed that the users who used larger parts of the system were more comfortable using the system and mastered it better (Thoresen 1997). Users know ITs through their experiences with them, as part of activities they engage in. They may know that input of erroneous number in a particular system is crucial to get the correct results (Gasser 1986), or they may learn how to trick a system into behaving like the tool they need. Computers are, however, not designed to make you more competent through use – a consequence of the emphasis on "easy-to-use" rather than "good, specialist tool" as the design goal. The first goal is better for consumer products (see e.g. Norman 1988), the latter fits with in-house, tailored design (see e.g. Greenbaum & Kyng 1991; Bratteteig 2004).

Concluding remarks

The ambition of this paper has been to address the IT in "gender and IT" from the inside of the informatics field. I have made two main points.

The first point is that IT is basically presented too narrowly as very technical work like programming (hacking) and as only box-like computers. IT includes much more both as a field of working, a knowledge area and as technology. The narrow presentation of IT and informatics makes

- women in informatics become invisible
- traditional “female” skills and interests in informatics become invisible
- and the myths about informatics as hacking live on

and contribute to strengthen the myths about informatics and IT. The image of a hacker does not appeal to most women or men (Bratteteig & Verne 1997) as it is a socially and culturally very narrow role. We should ask: Who do we want as designers for the information systems we are surrounded by and dependent on? If we want to recruit more different designers we may have to challenge informatics to present itself (and be presented) differently. A more varied population of informatics designers may make different kinds of ITs – different automations, representations, and distributions.

The second point is that knowing IT is more than being able to search and type on a keyboard: being literate is a higher ambition. IT is “never” the goal, it is always the means: IT is a tool or a prosthesis supporting human activity – only as art is it a goal itself as an expression. Informatics is therefore more than programming: at its core is the translation of human activity to running computer programs. To be a translator requires skills of both what you translate from: the use context, and what you translate to: computer logic. Informatics people should be able to

- translate the use logic to computer code
- create good interaction
- create good use environments

and also (more than today) present computer functions and logic in ways that enable users to become computer literate.

With these points in mind a discussion of gender and IT needs to address both how gender is performed in informatics and how technology is not neutral in ways that can be analyzed with a feminist perspective. Unfortunately, few gender studies include technical know-how, and end up enforcing rather than questioning technically based myths. A more nuanced view on IT would put more focus on epistemology rather than equity studies and perhaps lead to different perspectives on automaton(s), representation(s), and distribution(s).

References

Adam, A. (1995): Women and Computing in the UK. *Communications of the ACM* (special issue on Women in Computing) 38 (1): 43

Bateson, G. (1972): *Steps to an Ecology of Mind*. New York: Ballantine

Bjerknes, G. & T. Bratteteig (1988): Computers—Utensils or Epaulets? The Application Perspective Revisited, *AI & Society*, 2(3): 258-266

Bowker, G. C., and Star, S. L. (1999). *Sorting things out: Classification and its consequences*. London, England: The MIT Press

Bratteteig, T. (2002): *Bringing Gender Issues to Technology Design*, in Floyd, C et al (eds): *Feminist Challenges in the Information Age*. Germany: Verlag Leske + Budrich

Bratteteig, T. (2004): *Making Change. Handling Design and Use of Information Systems*, Dr.Philos dissertation, Dept. of Informatics, University of Oslo

Bratteteig, T. (2008): Does it matter that it is digital?. In Lundby (ed) *Digital Storytelling, Mediatized Stories: Self-representation in New Media*, New York: Peter Lang Publ.: 271-284

Bratteteig, T. & G. Verne (1997): Feminist of merely critical? In search of a Gender Perspective in Informatics. In Moser, I. et al (eds) *Technology and Democracy – Comparative Perspectives*, Oslo: University of Oslo: TMV

Buckingham, D. (2006): Defining digital literacy. *Digital kompetanse. Nordic Journal of Digital Literacy* 1 (4), 78-91/263-276. Oslo: Universitetsforlaget

Camp, T. (1995): Diversity Recruiting. *Communications of the ACM* (special issue on Women in Computing) 38 (1): 61-65

Cockburn, C. (1983): *Brothers. Male Dominance and Technological Change*. London: Pluto Press

Cockburn, C. (1985): *Machinery of Dominance: Women, Men and Technical Know-how*. London: Pluto Press

Dourish, P. (2001). *Where the Action Is: the Foundations of Embodied Interaction*. Cambridge: MIT Press

Fletcher, J.K. (1994): *Toward a theory of relational practice in rganizations: A feminist reconstruction of "real" work*. Ph.D. dissertation, Boston, MA: Boston University, Graduate School of Management

Gasser, L. (1986): The Integration of Computing and Routine Work. *ACM Transactions on Office Information* 4(3): 205-225

Gee, J.P. (2003): *What video games have to teach us about learning and literacy*, Palgrave Macmillan: New York

Goldstine, H.H. & A. Goldstine (1982): The Electronic Numerical Integrator and Computer (ENIAC). In *The Origins of Digital Computers: Selected Papers*. New York: Springer-Verlag: 359-373 (original article from 1946)

Greenbaum, J. (1976): Division of labor in the computer field, *Monthly Review*, 28 (3): 40-55

Greenbaum, J. & M. Kyng (eds) (1991): *Design at Work: Cooperative Design of Computer Systems*, Hillsdale, New Jersey: Lawrence Erlbaum Associates

Hacker, S. (1989): *Pleasure, power and technology: Some tales of gender, engineering and the cooperative work place*. Boston: Unwin Hyman

Harding, S. (1986): *The Science Question in Feminism*. Itacha: Cornell University Press

Harper, D. (1987): *Working Knowledge: Skill and Community in a Small Shop*, University of California Press

Huff, C.W. & J. Cooper (1987): Sex Bias in Educational Software: The Effect of Designers' Stereotypes on the Software They Design. *Journal of Applied Social Psychology*, 17 (6): 519-532

Håpnes, T. & B. Rasmussen (1991): The Production of Male Power in Computer Science. In Eriksson, I. et al (eds): *Women, Work and Computerization: Understanding and Overcoming Bias in Work and Education*. Proceedings of the IFIP TC /WG 9.1 International Conference on Women, Work and Computerization: Kluwer: 395-406

Klawe, M. & N. Levenson (1995): Women in Computing: Where are We Now?. *Communications of the ACM* (special issue on Women in Computing) 38 (1): 29-44

Kvande, E. & B. Rasmussen (1989): Men, women and data systems. *European Journal of Engineering Education* 14 (4): 369-79

Laurel, B. (2001): *Utopian Entrepreneur*. Cambridge: MIT Press

Livingstone, S. (2004): Media literacy and the challenge of new information and communication technologies, *Communication Review* 7: 3-14

Livingstone, S. (in press): Youthful Experts? A critical appraisal of children's emerging internet literacy. In C. Ciborra, R. Mansell, D. Quah and R. Silverstone (eds): *Oxford Handbok on ICTs*, Oxford: Oxford Univesity Press.

Mellström, U. (2003): *Masculinity, power and technology: A Malaysian ethnography*, Hampshire, UK: Ashgate

Mörtberg, C. (1997). *'Det beror på att man är kvinna …' Gränsvandrerskor formas och formar informationsteknologi.* University of Luleå dissertation series, Luleå: Department of Work Science: Gender and Technique

Mörtberg, C. (2001). Abstracting, Quantifying, Classifying, Simplifying, Standardising, Building Hierarchies: What are the Systems Designers Sorting Out?. Proceedings of The conference *Information Technology, Transnational Democracy and Gender*, Nov. 17, Ronneby, Sweden

Norman, D. (1988): *The Design of Everyday Things*, Basic Books

Nygaard, K. (1986): Program Development as a Social Activity. I Kugler, H-J. (ed): *Information Processing*, Elsevier Science Publ.: 189-198

Thoresen, K. (1997): Simple, But Cumbersome, *Proceedings of DIS '97*, ACM: 385-394

Turkle, S. (1984): *The Second Self*. New York: Simon and Schuster

Wajkman, J. (1991): *Feminism confronts technology*. Oxford, UK: Polity

Waldén, Louise (1994): "Those living sewing machines …" or Is Male to Female as Technology to Humanism. In Gunnarson, E. & L. Trojer (eds): *Feminist Voices on Gender, Technology and Ethics*, Centre of Women's Studies, Luleå University of Technology: 32-42

Webster online dictionary www.webster.com, accessed August 2008

Winograd, Terry. (ed.) (1996): *Bringing Design to Software*, Addison-Wesley