Paper presented to the Royal Society of Canada Symposium on Women in Scholarship in the Information Age, Calgary, June 1994. Published in **Transactions of the Royal Society of Canada** pp.113-131, 1994

Women, Scholarship and Information Technology: A Post-Modern Perspective

Mildred L G Shaw
Knowledge Science Institute
University of Calgary
Calgary, Alberta, Canada T2N 1N4
mildred@cpsc.ucalgary.ca

Post-modern perspectives emphasize the social construction of meaning and seek to overcome the bias that arises from linear thinking and a narrow focus on objective values. This provides a framework for the analysis of information technology as arising out of social needs rather than being an autonomous phenomenon that has created an 'information age'. However, education in University Computer Science departments generally provides disciplinary perspectives which emphasize the technology and pay little attention to its social origins. This is both an effect of the male-domination of these departments, and a cause of the unattractiveness of computer science degree courses to female students. It also leads to a cognitive bias in the discipline such that major phenomena, such as the growth of the Internet, are neglected because they cannot be understood in technological terms but reflect major social needs. This paper discusses these issues and ways in which the impact of the gender bias can be alleviated through approaches to teaching core material in the computer science curriculum, such as software engineering, which has an intrinsic and accepted social dimension. Information technology is too important to society for it to be presented to students in a way that fails to address the significance of social issues.

INTRODUCTION

When I was asked to organize this session on "women, scholarship and the information age", my initial reaction was to wonder whether computer and information systems has any relevance to gender issues in scholarship. It has always seemed to me that the notion of an "information age" in which computers and information systems make a major difference to socio-economic structures is based on a reversal of cause and effect. Technology arises out of social needs (MacKenzie and Wajcman, 1985; Bijker, Hughes and Pinch, 1989) and "its effect" is most readily understood through a comprehension of those needs rather than the study of the technology as itself an autonomous causal agent (Winner, 1977). That is not to say, that there are not unexpected side-effects of any major innovation—Popper (1968) has noted the autonomy of the world of ideas detached from their human origins—Ellul (1964) has provided a significant and stimulating perspective from which technology is viewed as autonomous.

Beninger (1986) provides a socially-grounded rationale for computing and information technology as yet another step in the "control revolution" commencing in the 1800s as a response to the increase in the speed, volume and complexity of industrial processes:

> "The Information Society has not resulted from recent changes but rather from increases in the speed of material processing and of flows through the material economy that began more than a century ago. Similarly, microprocessing and computing technology, contrary to currently fashionable opinion, do not represent a new force unleashed on an unprepared society but merely the most recent installment in the continuing development of the Control Revolution." (Beninger 1986))

This analysis has seemed to me to provide an adequate systemic and historical account for the majority of the significant phenomena associated with the "information age." It is a significant, but not isolated or revolutionary, stage in an ongoing process of industrialization which is itself grounded in the social needs generated by human population growth beyond a level sustainable without technological support. What came first, the population growth or the technological support for it, is too simplistic a question to have a meaningful answer—one is dealing with a system having strong positive feedback loops that seem themselves adequate to account for much of the perceived autonomy of living systems (Ulanowicz, 1991). Toulmin's (1990) thoughtful and provocative account in *Cosmopolis* of the modern era as a response to a sixteenth century social crisis is in itself sufficient to undermine any concept of autonomous origins for the seventeenth century enlightenment that resulted in science, industry and the information age.

However, the *social—technological* dichotomy is a major dynamic underlying the development of computing systems and their applications in our society. Computer Science departments present computing primarily from technological perspectives, and students are often attracted to computing as a discipline and a career because interaction with computers is a substitute for inter-personal interaction. The academic discipline of computer science is not one favorable for the development of social understanding, and this is unfortunate in that the technologists trained in the discipline have a major influence on the development and delivery of computer systems. This bias towards the technological and blindness to its role in satisfying social needs also induces a gender bias in the discipline, making it less attractive to female students. Since women, who would most naturally adopt perspectives of values satisfying social needs, are already a small minority in both academia and industry the bias tends to get reinforced.

This paper proposes no major solution to these problems. It illustrates ways in which the impact of the bias can be addressed in teaching, and situations where the lack of social perspectives has led to, and continues to lead to, lack of understanding of major trends in computing. Hopefully, making these issues more overt may itself serve to undermine the worst effects of the bias.

INDIVIDUAL AND SOCIAL CONSTRUCTIVISM

My research during the past 20 years has been primarily concerned with psychological models of the individual as a *personal scientist* (Shaw, 1980), constructing a model of the world based on personal experience and interaction with others, directly and through media. I have developed computer-based tools to elicit from people their personal models of the world, and to reflect back to them structure and implications in these models that may not be apparent to them. The roots of this work lie in George Kelly's (1955) *personal construct psychology*, and his geometry of anticipation in psychological space (Kelly, 1969; Shaw and Gaines, 1992a) whereby an individual uses *constructs* as filters through which she perceives events:

> "Man looks at his world through transparent templets which he creates and then attempts to fit over the realities of which the world is composed.
>
> Constructs are used for predictions of things to come, and the world keeps on rolling on and revealing these predictions to be either correct or misleading. This fact provides the basis for the revision of constructs and, eventually, of whole construct systems." (Kelly, 1955)

Kelly's studies were primarily in clinical psychology, but the elicitation of personal models of the world has applications in a wide range of disciplines such as education, market research and management (Shaw, 1981). In the 1980s the development of *expert systems* became a major topic in the sub-discipline of artificial intelligence within computer science, and knowledge elicitation tools based on personal construct psychology have become significant to the development of expert systems (Shaw and Gaines, 1987; Shaw and Gaines, 1993).

Once one adopts a personal constructivist perspective, one of the most fascinating questions to be answered is, "given their idiosyncratic models, how can individuals collaborate and communicate?" How do social systems interact with individuals to create shared construct systems? Theoretical perspectives which address these issues lead to the conclusion that it is the relatively coherent construct systems we ascribe to *roles* that are the bridges between individuals and societies. A person as a physical entity generally supports many roles as psychological entities, and construct systems are not so much person-specific as role-specific (Shaw, 1985). *Role conflicts* correspond to incoherence between the construct systems appropriate to different roles within the same person. *Role stereotyping* corresponds to social expectations that certain psychological roles will be played by people with certain physical characteristics. The personal construct psychology approach to role theory and social systems provides a basis for analyzing many issues in the feminist and post-modernist literature—for example, Mary Heller's (1990) model of the modern world in terms of the location of the individual within various social spheres, and Rose Coser's (1991) emphasis of the importance of access to a "plurality of life worlds" in the emancipation of woman from gender stereotyping.

From a constructivist viewpoint, what is interesting about the *social needs—autonomous technology* dichotomy discussed above, is not which analysis provides the 'correct' explanation, but rather that there are *two* viable bases of explanation. Each provides a significant model of the information age, and where the models differ in their predictions the two perspectives together provide richer insights into possibilities, problems and opportunities than would either alone. This to me is the essence of post-modernism, a transcendence of linear thinking that presupposes one correct basis of explanation for any phenomenon, leading to a pluralistic world view that presupposes many models together providing a rich repertoire of insights into the phenomena of the life-world (of which technology is part). It is not that the world is meaningless, but rather that it is capable of sustaining many meanings. We have a choice among meanings, and we also have the choice not to choose—to retain a rich plurality of meanings as leaving open both choices and perspectives. Actions can be based on many models, and the necessity of making choices does not remove alternative perspectives—what we see as 'correct' is forever open to change.

What is frustrating about gender bias in all walks of life, but particularly in teaching, is that it reduces the perspectives available. In western culture, where the male perspectives are associated with presuppositions of the value of *individual power* and *competitive action*, students in male-dominated disciplines are likely to miss the *social needs* and *cooperative action* perspectives, even when these provide valuable alternative bases for understanding significant phenomena. The imbalances in a society that ignores the viewpoints, the metaphors, models and insights of half its members are not conducive to adaptability to a rapidly changing world. We need the creativity that arises from multiple perspectives to be able to invent viable futures. We need to value variety in our society and realize that meanings, cultures and patterns of behavior are human inventions, not the imposition of the necessities of a deterministic universe.

PROVIDING SOCIAL PERSPECTIVES IN COMPUTER SCIENCE TEACHING

As I have already indicated, computer science, like other science and engineering disciplines, is male-dominated. A 1989 survey shows that Computer Science and Computer Engineering departments in the US and Canada have 6.5% female faculty with one third of the departments having no female faculty at all (Frenkel, 1990). In recent years, the proportion of women receiving doctorates in computing disciplines has been around 15% and declining, so that the research base in both universities and industry is also strongly male-biased. This leads to a cognitive bias and models of computing that are inadequate to account for many significant phenomena. For example, the significance of the growth of the Internet was missed for many years because it primarily provides a medium for cooperation and mutual support, a relevant nurturing environment supporting the growth of communities. This is not a conceptual framework that comes naturally to a male-dominated profession.

The perspectives under-represented through gender bias are of vital importance to the future of the computing industry and to its role in our society. It is important to provide modes of teaching that makes alternative perspectives accessible to computing science students. In particular, the 'received wisdom' approach of much existing science teaching is singularly inappropriate to the post-modern employment environment of computer scientists. They need to become reflective practitioners in Donald Schön's (1983) terminology, because they will need to continually adapt throughout their careers to social change in which their discipline plays a major role. Computer science teaching generally follows the traditional view of "privileged knowledge" in which it is the business of the professor to impart the knowledge embedded in texts. Students are fed "knowledge" in measured portions, expected to digest it, and give evidence in the form of assignments and examinations that they have done so.

While this model may have been effective in the past, the post-modern world in which our students are being prepared to take part is characterized by rapid change rather than the application of well-established knowledge. A more operational view of knowledge takes a constructivist approach to learning which is exemplified by the work of Jean Piaget (1972), Seymour Papert (1980) and others. This theory of knowledge implies that students learn through active involvement in the social processes of the construction of meaning. Understanding is based on active participation in the subject matter and reflection on conversations with others on the topics of a course. This is the Schön's "reflection-in-action" in which research and learning is a joint enterprise, and leads to a less authoritarian model of professionalism.

As John Sculley (1991) has emphasized, the key strength of 21st century organizations will be their ability to unleash and coordinate the creative contributions of many individuals; over-specialization and a limited perspective can be a dead-end trap; individuals will need to have tremendous flexibility to move around; a diverse educational experience will be the critical foundation for success; what we will need is not just mastery of subject matter but mastery of learning; we must have access to the unbounded world of knowledge; we must create a learning environment in which research and instruction are integrated. He specifies the requirements for lifelong learning:

- It should require rigorous mastery of subject matter.
- It should hone the conceptual skills that extract meaning from data.
- It should promote a healthy skepticism that tests reality against multiple points of view.
- It should nourish individual creativity and encourage exploration.

- It should support collaboration.
- It should reward clear communication.
- It should provoke a journey of discovery.
- It should be energized by the opportunity to contribute to the total of what we know and what we can do.

The following examples illustrate how these objectives may be achieved with existing computer science curricula.

Software Engineering

Software engineering is an excellent topic within which to introduce social perspectives because it has long been widely accepted that the problems of harnessing the talents of individual programmers into collaborative teams is a major one for the computer industry (Brooks, 1975; Mayrhauser, 1990). In recent years the gap between customer and user requirements and computing system specification has also become of major concern—*requirements engineering* is now a major sub-discipline in its own right (RE, 1993; Shaw and Gaines, 1994). Thus, it is natural to introduce psychological and social issues in a software engineering course, and to design project work that gives students personal experience of social phenomena in overt and discussible form.

The curriculum I have developed for CPSC 451, a required software engineering course for all Computer Science majors at this University, is centered on projects that involve the students playing roles in teams representing customer and supplier organizations. There is a lecture component of 3 hours a week for 13 weeks, and a laboratory (practical) component of 3 hours a week for 13 weeks. The lectures are of standard format covering classical software engineering topics and methodologies, including the SEI levels of maturity and continuous improvement in an organization (Humphrey, 1989). The practical component is of more interest here.

Each student is assigned to two different groups of 10 to 12 students. In one group she is one of the supplier team, and in the other one of the customer team. The students are assigned by me based on a number of factors, as they are personally unknown to me at this point. Some of these factors are: having taken the human-computer interface course, having taken a theory course, and length of time in the program, to try to make each group as varied as possible but at the same time as similar as possible to the other groups. The total class size in the past has been around 50 to 60 students, but in the coming year is likely to be around 120 (due to financial constraints).

The process starts in the last 5 minutes of the very first class, when each customer group is given a slip of paper on which is written a very short, informal and vague description of a problem. For example:

> Write a specification for a system linking supermarkets to a grocery supplier to process the ordering from one end and the invoicing from the other. Both sides should have strategies e.g. reorder when the stock drops to a certain level; do not invoice until the stock has been received. This requires the minimum data entry, but complete security.

Each group gets a different problem, but they are all of a similar level of complexity. Each customer group has two days to prepare an informal requirements document for the project. They are subsequently responsible for its evaluation and criticism as it progresses; that is, they are the customer for the system. They are expected to be present at all presentations to ask questions, and comment on all the write-ups and documentation. Each grade, given by me not by the students, depends on how thoroughly the evaluation is carried out, the extent to which it is fair and reasonable and the extent with which it agrees with a well-founded methodology. Groups are advised to show all drafts to the teaching assistant (TA), and discuss any problems or disagreements. It is not very long after the start of the project that the customer and supplier groups reach the conclusion that the interchange cannot be done entirely by documents, and that they need to meet and negotiate problems, expectations, and what will be included in each version of the software.

The supplier group works with a different problem to receive informal requirements for a system, to produce a formal specification, a management plan, produce the analysis and design in the form of an overall and a detailed design document including test plans, a user manual, code a prototype, evaluate and refine the prototype, and present a final (prototype) product according to the details given. Public (to the whole class) oral presentations and discussions are required at various points within the project, and are evaluated and assessed by the customers. It is certainly not required, but usually the students will arrive in their best business clothes for the presentations, and fully enter into the roles they have been given. Every 3 to 5 days another part of the project becomes due for submission to the customers (and to me and the TA), and students are quickly made aware of the social pressures to conform to due dates. This may be the first time that any of them have considered that due dates are not altogether arbitrary, and that other people's deadlines depend on them. In turn, each student may be inconvenienced by other people's last minute rush to complete work, not only in the other group, but also in their own where, for example, an editor may require input from several people before the final document can be prepared.

No marks are given for coding. This is where much of the effort goes, and this practice seems to the students to be unfair. However, I know that all the students have a thorough grounding in programming, and what they are learning through me is the application of what they already know and is being covered in lectures, the management of their time, how their own working style fits into the range of working styles among their peers, and how their own time and work management affects others. Each student keeps a log of activities in the form of a diary with dates, what was done, and time spent on each item. At the end of each month, every student prepares a set of reports assessing each member of their supplier group. This requires a paragraph per person, including themselves, outlining who did what during the month, and how their work can be assessed, using some sort of grading scheme on one or more criteria. This means that they must get to know the people in their group as soon as possible, and decide how each has contributed to the group work during the month. This is not optional, but is a required part of the assessment to pass the course. In general the reports are thoughtful and responsible. The students do not always give themselves top marks, but say things like: "x did not contribute much to the group discussions — maybe I spoke too much of the time and didn't give x a chance"; or "I thought y was really stupid at first because his section of the documentation was such a mess — but I soon found out that he was a very good programmer and just had difficulty expressing himself in English".

In order to prepare students for the project, apart from technical considerations, I spend one hour talking about personality variables, the range of possible working styles, and how a person's strong points should be built on rather than concentrating on their weaknesses. We carry out a short form of the Myers-Briggs Type Indicator for each student to find out their own personality preferences, and discuss how each type can benefit from input from other types. The students invariably get the message, without it being made explicit, that their job is to encourage the smooth functioning of the group and to involve every member as equally as possible without requiring everyone to take part in every job.

This is a course with a heavy workload, and students need a lot of encouragement, especially towards the end of term. The TA acts in an advisory capacity, making suggestions but not giving definitive answers, buffering me from any interference or undue influence on any of their decisions. The groups can use any methods, equipment, and language of their choice, but each must be justified as to its suitability for the project. There is no doubt, however, that after the course has finished the experience is highly valued, as an extract from a letter sent recently from an ex-student will demonstrate:

> ...although I didn't fully appreciate it at the time - much of what I learned in the class project...helped me to better understand and cope with the working world. For the last seven months I have been working as a contract programmer at Deutsche Bank's Regional Head Office in Singapore....I currently have to code according to design documents....work in a team of over 20 people all making changes to the system simultaneously, rushing to meet deadlines - like what I went through during the CPSC 451 project. Thanks for all you've taught me....

It is not necessary to make explicit psychological and sociological perspectives part of the academic curriculum in CPSC 451. In any event, the science curriculum to which most of the students have been exposed encourages linear thinking and objectivist values, and is a poor foundation from which to understand the life world. The students *experience* the significance of roles, construct systems and inter-personal interactions. The alternation of their own customer and supplier roles brings them to terms with the nature of construct systems, both their subjective artificiality and their ethical implications in terms of role consistencies, responsibilities and accountabilities. Being responsible for conceiving and articulating *requirements*, in particular, is a new experience for most students, and leads them to be more thoughtful about how those requirements arise.

Advanced Information Systems

Advanced information systems provide a topic where social perspectives are readily seen to be essential to redress technological bias. When I developed the curriculum for CPSC 547, an optional course on advanced information systems for Computer Science majors, I knew that the final year students already had theoretical foundations for technologies such as object-oriented programming and databases. I was also aware that many of the students who were attracted to this course also had substantial industrial experience. For example, they understood object-oriented technology in terms of type theory, modularity, and so on, and they understood that it was having a major impact on industry, but they had few sources available on how to bridge the gap between theory and practice: for example, to be able to see object-oriented databases as providing a more effective enterprise modeling technology than relational databases; from there, to go on to the questions of the interplay between organizational needs and technological capabilities; from there, to go on to the question of the influence of the technology on organizational design; and so on.

However, my agenda for CPSC 547 has gone far beyond these simple techno-social considerations. The students in this course are preparing for a new industrial infrastructure which is itself radically different from that of a few years ago. It is 'post-modern' in the sense of Paul Ekins and Manfred Max-Neef's (1992) *real life economics* recognizing the plurality of economic sectors including environmental and domestic capital, of Hans-Jürgen Warnecke's (1993) *fractal* *company* designed to encourage the growth of complete and robust sub-organizations self-similar in their functionality to the whole, of Margaret Wheatley's (1993) *emergent organizations* recognizing the adaptability of the creative chaos of the life world. In Calgary, the recession has seen the end of the large-scale information systems divisions of the oil majors that has dominated computer science employment opportunities in Alberta. Hundreds of information systems professionals have already had to come to terms with a new industrial environment that emphasizes small, adaptable, entrepreneurial organizations. Our graduating students need skills that go beyond mere technical proficiency to cope with the new challenges and opportunities.

In building a classroom environment suitable for reflective learning I have been influenced by the recommendations and beliefs of Carl Rogers (1961) for generating a positive atmosphere in which students exhibit mature everyday behavior, are less defensive, more adaptive, and more able to meet situations creatively. This involves treating each student as an individual, making myself available to discuss problems individually and help with students' decision-making, creating a supportive and empathic class atmosphere in which each student is given positive encouragement to discuss issues of concern, and making my own thoughts and views genuinely available for discussion. According to Rogers this allows each student to experience and understand aspects of her/himself which may not have been previously available, to become more integrated and more able to function effectively, to be more self-directing and self-confident, to become more self-expressive, to be more understanding and accepting of others, to be able to cope with new problems more adequately and more comfortably.

This is what I attempt to do in CPSC 547. However, it is not simple to switch to this type of classroom interaction for those with years of experience with a traditional approach to learning. For one thing, it threatens the view of the "authority" of the professional who is the ultimate source of all knowledge, and hence requires a high degree of competence and understanding of the subject matter and its ramifications. It also involves a personal commitment to knowing every student in a class of 50 by name by the end of the second week of the course, and to be willing to support requirements for resources and equipment that cannot be planned. The students do a great deal of reading the literature, thinking and discussing issues. Emphasis is on cooperation, mutual acceptance and support for differing points of view.

After a few lectures, the students in CPSC 547 take over the course and run it through their own group research, presentations and demonstrations addressing major issues in advanced information systems. The students work extremely hard, are incredibly motivated and enthusiastic, achieve a very high standard of work, and think deeply not only about the technology but also about the social and ethical implications of its applications.

In CPSC 451 they learn experientially from playing the relatively well-defined roles of customers and suppliers. In CPSC 547 they learn both experientially and intellectually from playing the open-ended roles of being researchers and educators in their own right. Each presentation tends to set a new standard of excellence which those in the later groups are determined to transcend, and find they must cooperate strongly to do so.

Whereas CPSC 451 is a compulsory course, CPSC 547 is optional, and the fact that it is by far the most heavily subscribed of our 500-level courses attests to its perceived value by students. It provides a bridge from their roles as students to their roles as industrialists, managers, researchers, members of, and contributors to, our rapidly changing post-modern age and information society.

Graduate Creativity

It is significant that both 451 and 547 are senior courses. I have taught junior Computer Science courses and found the students not yet ready to face the issues raised above. I believe this readiness involves their having already become computer science professionals, involved in the technology and its understanding, and their becoming aware, even if tacitly, that the technological perspective leaves much that is significant to them unexplained. For graduate students the situation is somewhat different because, for some, the involvement with the technology remains their primary motivation, and, for others, the university provides a haven that avoids contact with social issues. It is still, however, relevant to introduce these issues to those who may come to have a major influence over new directions in the development of information technology.

In graduate courses on cognitive science and artificial intelligence (CPSC 679 and 671), I have found it possible to use the strong acceptance of knowledge elicitation methodologies based on personal construct psychology to introduce students to the psychological and social foundations through the tools. One of the applications projects has been for each student to develop a conceptual model of their research domain in the form of a computer knowledge base using knowledge acquisition and representation tools to represent it computationally and graphically, linking it with hypermedia to annotate it, and comparing the elicited structures with someone else in the same field (Shaw and Gaines, 1992b; Gaines and Shaw, 1993). The tools, which were developed within the knowledge acquisition research community to support the knowledge engineer in developing knowledge-based systems, are based on constructivist methodologies from psychology, education and management designed to support people in developing overt conceptual structures.

RepGrid is a computer-based implementation of methodologies from personal construct psychology providing an integrated set of tools for elicitation, modeling and comparison of conceptual structures in a given domain (CPCS, 1993). The graduate students in the CPSC 679, Cognitive Processes in Artificial Intelligence, 1990 class were given access to the system for some two months with the assignment of developing their research ideas for their MSc research. Figures 1, 2 and 3 give examples of the interactive graphical elicitation of constructs using RepGrid by a graduate student organizing her framework of learning strategies as exemplified by some of the publications from the Institute for Research on Learning in California.

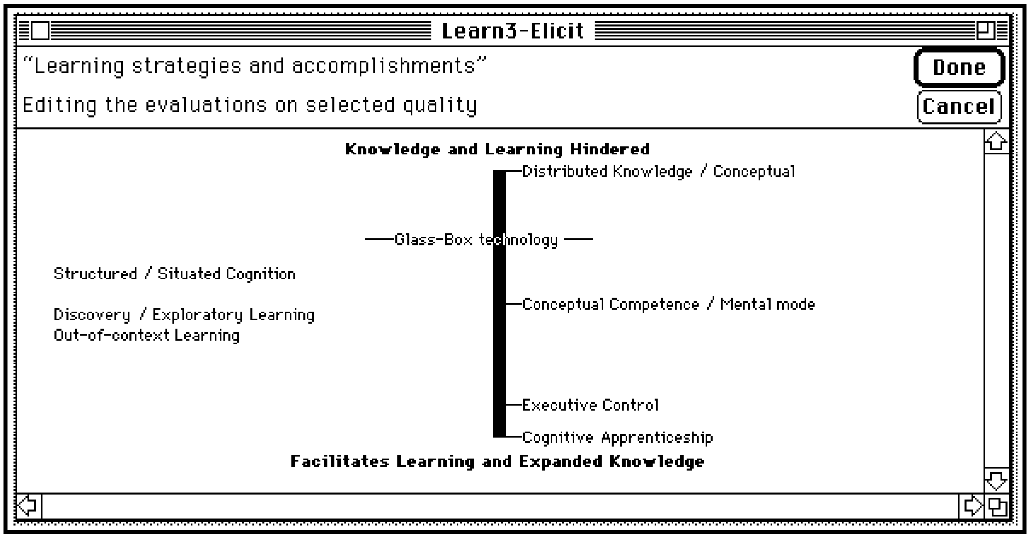

Figure 1 RepGrid Construct Rating

Figure 1 shows a construct in process of elicitation, with the elements listed on the left and being dragged on to the dimension of the bipolar construct *Facilitates Learning and Expanded Knowledge—Knowledge and Learning Hindered*. Figure 2 shows two constructs which are highly matched, showing the placing of the elements on each, and inviting the student to think of a new element which would reduce the match level.

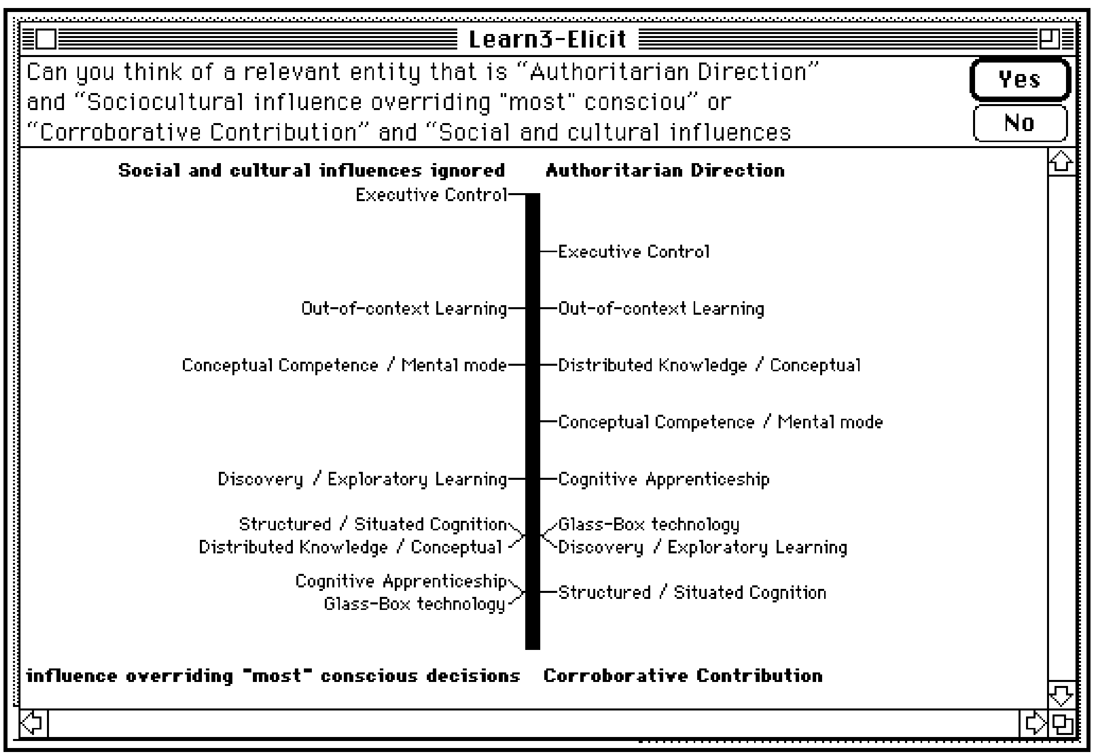

Figure 2 RepGrid Construct Match

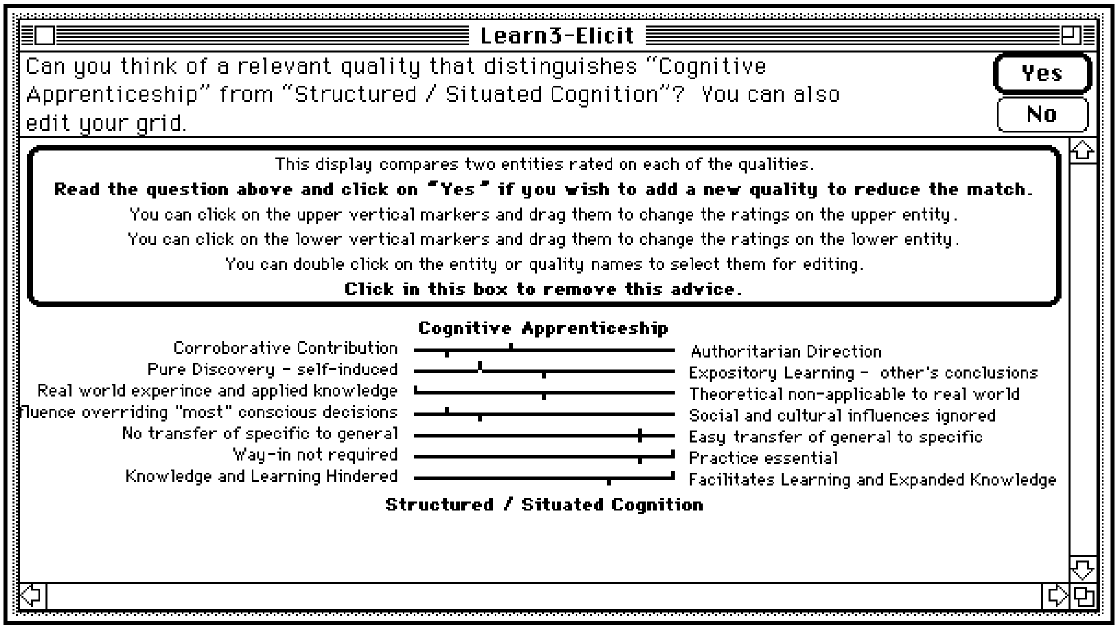

Figure 3 RepGrid Element Match

Figure 3 shows a similar situation with two elements which are highly matched. The marker on top refers to the element *Cognitive Apprenticeship* and the marker below to the other element *Structured / Situated Cognition.* Elements and constructs can be moved or changed at any time.

All the students made heavy used of the hypermedia annotation facilities to annotate their elements and constructs and to add additional material such as background notes and diagrams. Since the supplied annotation facilities were designed to operate through a well-defined protocol leaving the full functionality of HyperCard open and available, several of the students took the opportunity to use the hypermedia tool for their own purposes. One student presented his complete final report on this project in the stack embedding in it buttons that linked to examples, diagrams and references.

The students were also asked to “exchange” their grids with someone in the same field, which often turned out to be the supervisor. One student, specializing in instructable systems, completed his project, and in addition he did an exchange grid with a member of his supervisory committee. This involved using the committee member’s element and construct names, and providing his own ratings of the elements on the constructs. All the elements were machine learning systems well known to both. The resulting grids were compared by literally subtracting one from the other, then reordering the rows and columns to put the one with the smallest differences in the rating values toward the top left, as shown in Figure 4.

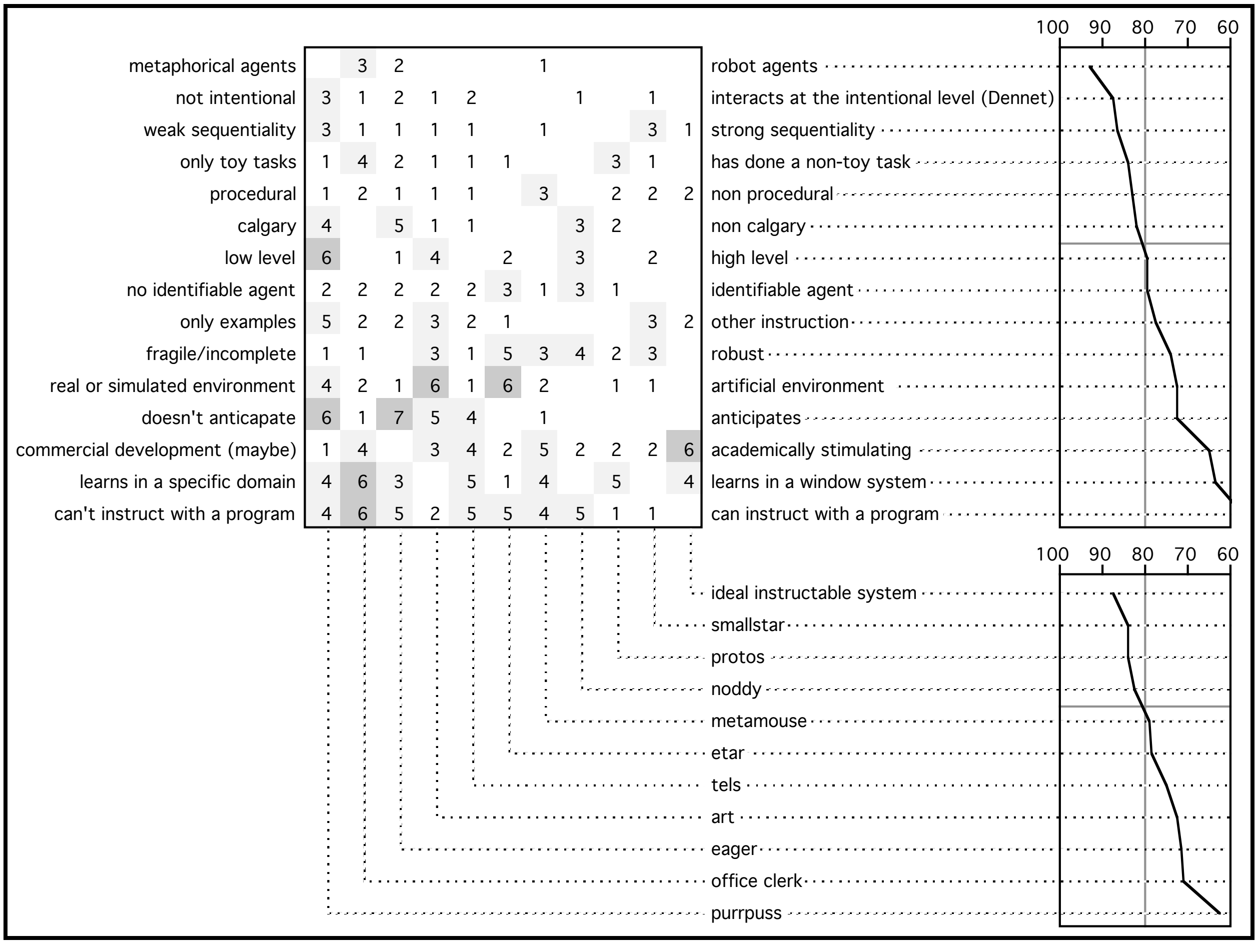

Figure 4 RepGrid Difference Grid between Researchers

The result shows that the student and supervisor construe the systems *ideal instructable system, smallstar, protos,* and *noddy* in much the same way, as described by the top six constructs *metaphorical agents—robot agents, not intentional—interacts at the intentional level (Dennet), weak sequentiality—strong sequentiality, only toy tasks—has done a non-toy task, procedural—non procedural,* and *calgary—non calgary*; but see *purrpuss* very differently as described by the constructs *commercial development (maybe)—academically stimulating, learns in a specific domain—learns in a window system*, and *can't instruct with a program—can instruct with a program.* This sort of analysis can lead to discussion of similarities and differences, and possibly a new perspective on a topic, not only for the student but for faculty working in the same field. It also confronts students and faculty with the fact that they may have been discussing a topic for some time without being aware that they are using concepts in different ways—for example, the *anticipates—doesn't anticipate* dimension appears definable by objective criteria and yet is shown by the exchange grid to be used in very different ways.

Summary—Levels of Awareness of Social Issues

These three examples show that it is possible within existing computer science curricula to introduce highly relevant material and approaches to teaching that raise students' awareness of social issues and their relevance to technology. The 451 students instantiating the roles of both customers and suppliers experience the social issues and the way in which technology arises out of human needs, which may in themselves be controversial, difficult to express, and influenced by prior conceptualizations of available technology. The 547 students instantiating the roles of researchers and educators addressing topics at the frontiers of information technology find from personal experience that they have to introduce a social dimension to understand and explain the technology. They also find that the social dimension serves to focus and integrate much of what they already know about technical issues. The graduate students are at a stage in their careers where they are individualizing as professionals and know that they have to move beyond the 'received view.' Exploring their personal conceptual frameworks and comparing them with those of their colleagues, both students and faculty, makes them aware of the personal and social constructions that underlie what has previously been presented to them as 'reality.'

It is not clear to me that the teaching issues discussed are directly related to gender bias. In the humanities there are certainly male colleagues who would find the approaches I have adopted to teaching to be natural and obvious. In management science many of the issues are raised but generally in a normative sense rather than a reflective one—the management culture tends to lead students to view power as something to be gained and wielded for its own sake rather than as a manifestation of trust within effective collaboration. The technology bias and male domination of computer science are correlated and systemically related phenomena. My experience suggests that it is both natural and possible for a woman to have some impact on redressing the technology bias without having to make major changes to the essence of the discipline.

UNDERSTANDING THE INTERNET

In conclusion, let me switch from pedagogical issues to a major example of a computer science phenomenon that cannot be understood from a purely technological perspective and requires social understanding for its analysis. There is a recent phenomenon in computer and information systems that seems to be radically anomalous in terms of any existing theories of the technology as arising out of social needs, and that is the growth of the Internet (Quarterman, 1990). This technology does seem well-modeled as 'autonomous' in its massive, unplanned growth and widespread social impact, and the 'social needs' that it satisfies seem to be ones that are important precisely because they were never identified as such.

The growth of the Internet was unexpected, unplanned, falling outside commercial free-market projections and aspirations, and outside governments' frames of reference, and it is perhaps its very freedom from the constraints and value systems of such existing institutions that is enabling the network to be so many things of great importance to so many people. The Internet by-passes much existing regulatory mechanisms, whether that regulation comes from the free-market economy, or from government legislation. This is fascinating in its own right since modern economic theories of law see these two as opposing forces balancing the dynamics of modern Western society. The Internet may have the potential to support a new dynamic in our society—one that provides more flexible social groupings, roles, more choices, more bases for the creation of new meanings—in short, more support for the positive aspects of a post-modern society in which a plurality of cultures can coexist, evolve and thrive.

There are also significant gender issues arising in studies of the Internet. Digital communication distanced not only from physical proximity but also from physical representation of voice, appearance, and so on, has the potential to avoid unwanted pressures on role integrity, and the possibilities of role-preemption through existing patterns and expectations of dominance and power. In this respect, the Internet has the potential to be gender-neutral in supporting the roles that individuals play through discourse on the net, and this has significant implications for the evolution of work, education and professional disciplines.

At a different level of analysis, most of the development of the Internet has been through a grass-roots mutual-support activity and provides an outstanding example of how public goods may come into being on a large-scale with little in the way of apparent social or legislative pressure. Many of the activities on the net provide relevant, nurturing environments for individual development in which newcomers are welcome both because they bring needs which the group can satisfy and also because they bring innovation through which the group can evolve. This aspect of the Internet is one that feminist literature would identify as consistent with female gender values emphasizing the significance of a nurturing environment promoting individual growth through mutual support, and as antithetical to male gender values emphasizing the significance of a competitive environment promoting individual growth through mutual conflict (Keller, 1985; Code, 1991; Harding, 1991; Held, 1993).

It is plausible to argue that one reason the significance of the Internet was so little recognized in its early years is precisely because it was based on a freely giving nurturing society that was orthogonal in its value system to free market economics and authoritarian regulation. It is also rather surprising that it was not seen as a threat by commercial or government interests and suppressed by economic or legislative pressures.

CONCLUSIONS

To understand computer technology we must view it also as a phenomenon of the life world embedded in its processes, both generated by them and generating them (Blum & McHugh 1984). If we treat technology as autonomous we forget its roots:

> "Technology is the human's achievement, not his failing—even though the use he chooses to make of it may be fallen indeed. If the products of human techne become philosophically and experientially problematic, it is, I would submit, because we come to think of them as autonomous of the purpose which led to their production and gives them meaning. We become, in effect, victims of self-forgetting, losing sight of the moral sense which is the justification of technology. Quite concretely, the purpose of electric light is to help humans to see. When it comes to blind them to the world around them it becomes counterproductive. The task thus is not to abolish technology but to see through it to the human meaning which justifies it and directs its use." (Kohak, 1984)

The gender bias in computer science leads to over-emphasis on the technology and to the neglect of the origins of that technology in social needs. There are many ways in which the balance can be redressed, but the most important long-term foundations for them all is to achieve more balanced gender participation in computer science education. It is the plurality of perspectives that more balanced participation would provide that is the primary educational objective. Evelyn Fox Keller who has written extensively and profoundly on gender issues in science, expresses the essence of matter:

> "That philosophy has taught me to seek a science named not by gender, or even by androgyny, but by many different kinds of naming. A healthy science is one that allows for the productive survival of diverse conceptions of mind and nature, and of correspondingly diverse strategies. In my vision of science, it is not the taming of nature that is sought, but rather the taming of hegemony." (Keller, 1985)

ACKNOWLEDGEMENTS
I wish to thank Brian Gaines for helpful discussions of the issues raised in this paper, and the students on whose work I have commented.

REFERENCES

Beninger, J.R. (1986). **The Control Revolution: Technological and Economic Origins of the Information Society**. Cambridge, Massachusetts, Harvard University Press.

Bijker, W.E., Hughes, T.P. and Pinch, T., Ed. (1989). **The Social Construction of Technological Systems**. Cambridge, Massachusetts, MIT Press.

Brooks, F.P. (1975). **The Mythical Man-Month: Essays on Software Engineering**. Reading, Massachusetts, Addison-Wesley.

Code, L. (1991). **What Can She Know? Feminist Theory and the Construction of Knowledge**. Ithaca, Illinois, Cornell University Press.

Coser, R.L. (1991). **In Defense of Modernity**. Stanford, Stanford University Press.

CPCS (1993). **RepGrid 2 Manual**. Centre for Person Computer Studies, 3019 Underhill Drive NW, Calgary, AB, Canada T2N 4E4.

Ekins, P. and Max-Neef, M., Ed. (1992). **Real-Life Economics: Understanding Wealth Creation**. London, Routledge.

Ellul, J. (1964). **The Technological Society**. New York, Vintage Books.

Frenkel, K.A. (1990). Women & Computing. **Communications of the ACM 33**(11) 34-46.

Gaines, B.R. and Shaw, M.L.G. (1993). Supporting the creativity cycle through visual languages. **AAAI Spring Symposium: AI and Creativity**. pp.155-162. Menlo Park, California, AAAI.

Harding, S. (1991). **Whose Science? Whose Knowledge? Thinking from Women's Lives**. Ithaca, Illinois, Cornell University Press.

Held, V. (1993). **Feminist Morality: Transforming Culture, Society and Politics**. Chicago, University of Chicago Press.

Heller, A. (1990). **Can Modernity Survive?** Berkeley, University of California Press.

Humphrey, W.S. (1989). **Managing the Software Process**. Reading, Massachusetts, Addison-Wesley.

Keller, E.F. (1985). **Reflections on Gender and Science**. New Haven, Yale University Press.

Kelly, G.A. (1955). **The Psychology of Personal Constructs**. New York, Norton.

Kelly, G.A. (1969). A mathematical approach to psychology. Maher, B., Ed. **Clinical Psychology and Personality: The Selected Papers of George Kelly**. pp.94-113. New York, Wiley.

Kohak, E. (1984). **The Embers and the Stars: A Philosophical Enquiry into the Moral Sense of Nature**. Chicago, Illinois, University of Chicago Press.

MacKenzie, D. and Wajcman, J., Ed. (1985). **The Social Shaping of Technology**. Milton Keynes, UK, Open University Press.

Mayrhauser, A. von (1990). **Software Engineering Methods and Management**. Boston, Academic Press.

Papert, S. (1980). **Mindstorms: Children, Computers and Powerful Ideas**. New York, Basic Books.

Piaget, J. (1972). **The Principles of Genetic Epistemology**. London, Routledge.

Popper, K.R. (1968). Epistemology without a knowing subject. Rootselaar, B. Van, Ed. **Logic, Methodology and Philosophy of Science III**. pp.333-373. Amsterdam, North-Holland.

Quarterman, J.S. (1990). **The Matrix: Computer Networks and Conferencing Systems Worldwide**. Reading, Massachusetts, Digital Press.

RE (1993). **Proceedings of First International Symposium on Requirements Engineering**. New York, IEEE Computer Society Press.

Rogers, C. (1961). **On Becoming a Person**. London, Constable.

Schön, D.A. (1983). **The Reflective Practitioner**. New York, Basic Books.

Sculley, J. (1991). The relationship between business and higher education: a perspective on the 21st century. Dunlop, C. and Kling, R., Ed. **Computerization and Controversy: Value Conflicts and Social Choices**. Boston, Academic Press.

Shaw, M.L.G. (1980). **On Becoming A Personal Scientist: Interactive Computer Elicitation of Personal Models Of The World**. London, Academic Press.

Shaw, M.L.G., Ed. (1981). **Recent Advances in Personal Construct Technology**. London, Academic Press.

Shaw, M.L.G. (1985). Communities of knowledge. Epting, F. and Landfield, A.W., Ed. **Anticipating Personal Construct Psychology**. pp.25-35. Lincoln, Nebraska, University of Nebraska Press.

Shaw, M.L.G. and Gaines, B.R. (1987). KITTEN: Knowledge initiation and transfer tools for experts and novices. **International Journal of Man-Machine Studies 27**(3) 251-280.

Shaw, M.L.G. and Gaines, B.R. (1992a). Kelly's 'Geometry of psychological space' and its significance for psychological modeling. **New Psychologist** 23-31.

Shaw, M.L.G. and Gaines, B.R. (1992b). Mapping creativity with knowledge support tools. **AAAI-91 Workshop on Creativity: Models, Methods and Tools**. pp.32-45. Menlo Park, California, AAAI.

Shaw, M.L.G. and Gaines, B.R. (1993). Personal construct psychology foundations for knowledge acquisition and representation. Aussenac, N., Boy, G., Gaines, B., Linster, M., Ganascia, J.G. and Kodratoff, Y., Ed. **Proceedings of EKAW-93: Seventh European Workshop on Knowledge Acquisition for Knowledge-Based Systems**. pp.256-276.

Shaw, M.L.G. and Gaines, B.R. (1994). Requirements acquisition. **Proceedings of Workshop on Requirements Elicitation for Software-Based Systems**. Keele, University of Keele.

Toulmin, S. (1990). **Cosmopolis: The Hidden Agenda of Modernity**. New York, Free Press.

Ulanowicz, R.E. (1991). Formal agency in ecosystem development. Higashi, M. and Burns, T.P., Ed. **Theoretical Studies of Ecosystems: The Network Perspective**. pp.58-70. Cambridge, Cambridge University Press.

Warnecke, H.J. (1993). **The Fractal Company: A Revolution in Corporate Culture**. Berlin, Springer.

Wheatley, M.J. (1993). **Leadership and the New Science**. San Francisco, Berrett-Koehler.

Winner, L. (1977). **Autonomous Technology: Technics Out of Control as a Theme in Political Thought**. Cambridge, Massachusetts, Riverside Press.