Peter B. Andersen, Mihai Nadin,
Frieder Nake (editors):

Informatics and Semiotics

Dagstuhl-Seminar-Report; 135
19.02.-23.02.96 (9608)

ISSN 0940-1121

Das Internationale Begegnungs- und Forschungszentrum für Informatik (IBFI) ist eine gemeinnützige GmbH. Sie veranstaltet regelmäßig wissenschaftliche Seminare, welche nach Antrag der Tagungsleiter und Begutachtung durch das wissenschaftliche Direktorium mit persönlich eingeladenen Gästen durchgeführt werden.

Bezugsadresse: Geschäftsstelle Schloss Dagstuhl
Universität des Saarlandes
Postfach 15 11 50
D-66041 Saarbrücken, Germany
Tel.: +49 -681 - 302 4396
Fax: +49 -681 - 302 4397
e-mail: office@dag.uni-sb.de
url: http://www.dag.uni-sb.de

Informatics and Semiotics
Introduction

Frieder Nake

In recent years, the theoretical and philosophical foundations of informatics have been questioned. While no one really disputes the roots of computing in formal logics, specialists in various fields have noticed that there is more to a theory of writing programs than formal logics and algorithmic theory could provide. With the rapidly growing and accelerating proliferation of computers and of complex application software into many areas of human activity, one particular aspect of software has begun to emerge: its intrinsic semiotic nature.

It should not come as a surprise that a distinction is necessary between „data" and „information", and a borderline has to be established to „knowledge", if basic concepts of informatics are to be defined in a rigorous way. Without such a clarification of concepts, informatics is not likely to be accepted as a proper scientific discipline.

With the concept of a „sign" and, perhaps even more importantly, with that of „semiosis" (sign process), semiotics offers a very promising and tempting conceptual basis to informatics. Peter Bøgh Andersen (1990) and Clarisse Sieckenius de Souza (1993) have published definitions of computer signs, i.e. signs as they appear on the screen of a computer display unit, or within the processor and memory of the computer. Their approaches, as well as those of others, indicate a new way of looking at the interactive use of computing machinery by humans.

In a simplistic manner, semiotics already played some role in informatics during the sixties, when programming languages were defined by the dozen. The distinction, by Charles Morris, of syntactics, semantics, and pragmatics, was then imported into programming language theory. A few authors, like, most prominently, Saul Gorn, and Heinz Zemanek explicitly referred to semiotics.

Hardly noticed by anyone, a small number of researchers in informatics have always maintained that any real-world situation or process has to undergo a semiotic transformation before it may be subjected to treatment by a computer program, and thus become a „computer application". This is usually referred to as *representation*: Without reducing a real world situation or process to signs – i.e. without representing it – it cannot become a subject matter of computation. The computer has therefore been called a *semiotic machine*. This characterization seems to gain in acceptance, but it was first suggested back in 1977 by Mihai Nadin. From this perspective, informatics can be viewed as *Technical Semiotics*. *Semiotic Engineering* is an alternative, and maybe equivalent, form to express the same idea (used by de Souza, and Jorna). *Semiotronics* was suggested even earlier, in 1988, by Pierre Maranda.

By itself, this observation does not distinguish informatics much from any other science or field of study. Indeed, in scientific endeavors, we always first replace that aspect of the real world that we intend to study with some specific model, and then continue to view that model as if it was the world. But a model is composed of signs that in turn form supersigns etc. Hence, models are always of a semiotic nature.

What is new, however, about informatics is its primary concern for computable functions and for the special type of machines (namely digital machines) that evaluate computable functions. Therefore, informatics deals with *algorithmic semioses*, i.e., sign processes that are computable and, moreover, are carried out by machines.

The last statement contains a contradiction. That contradiction corresponds to the complex nature of signs. Signs, in the proper meaning of the concept, appear as rich relational entities. They are pragmatically interpreted, semantically established, and syntactically processed. Computability, however, applies to the syntactic dimension of signs only. Therefore, in order to be processed by computer, signs have to be reduced to their syntactic dimension. When we perform that reduction of a sign to its syntactic component, the essential semiotic characteristics are strongly affected.

On the syntactic level, the sign relation appears as a physical entitity only. But we are hardly ever interested in the sign as a physical entity. Rather, our interest is in the sign as a something that stands for another something. We are those who establish that relation by an interpretative act. Interpretation is essential to signs. A sign owes its unique nature to the never-ending capacity of being interpreted. On the syntactic level, however, virtually all interpretation ceases, because we want computable processes to end with predictable results.

Therefore, „algorithmic semiosis" appears to be a contradiction in itself: the contradiction is that of predictable algorithmic behavior and indefinite pragmatic interpretability. Traditionally, central concepts of informatics have been programs and data, automata, languages, and algorithms, and systems. By and large, they still define important parts of many undergraduate curricula. They are sufficient when we want to establish a *functional* theory of programming, for which mathematics and engineering are responsible.

A second layer of informatics is connected to metaphors like tool or media. They became important when the „user" entered to scene. Sloppy, soft concepts like usability, interface, „look and feel", or participatory design surfaced. They indicate a turn toward a *systemic* theory of programming. Such a theory relies on social science and psychology, even on aesthetics.

From a third tradition, the concepts of information and communication still appear central to informatics, and in fact they have recently gained new strength.

Convincingly, „sign" appears as a notion common to all those concepts of informatics. Whereas Jorge Bogarín, in his 1989 Ph.D. thesis at the University of Stuttgart, discussed the traditional theoretical foundation of informatics (i.e., formal languages, automata, and recursive functions) from an explicitly semiotic perspective, others like, e.g., Aaron Marcus, make use of semiotic terminology in dealing with interface issues.

Forerunners usually get detected only after a phenomenon is declared to be a new one. In our case, pioneering work was done, on a very general level, by Max Bense, Elisabeth Walther, and their students at the University of Stuttgart during the 1960s. The first congress on semiotics at Milano, in 1974, already was aware of the impact of computers on semiotics. The famous, yet unsuccessful Fifth Generation Computer Project in Japan contained a semiotic aspect. Hiroshi Kawano, a pioneer of computer art, has long ago published on aspects of our topic. Mihai Nadin and Leif Allmendinger conducted an interface evaluation project for Apple from 1983 to 1985 in which semiotics played a role.

The Dagstuhl Seminar addressed the contradiction mentioned above. It had to find and sustain a balance between contributions by semioticians who take informatics into consideration, and by informaticians who refer to a semiotic perspective. Their views showed differences in theoretical background, as well as in practical conclusions. It was not easy to precisely identify such differences and discuss them with the aim of putting detailed questions onto the research agenda.

The seminar started out with two sets of related questions. The first set was:

- In which precise sense does semiotics belong to the foundations of a theory of informatics? What are the semiotic foundations of informatics?
- What happens to signs and sign processes when they are submitted to the computer and thus to computation?
- Which new kinds of signs appear on computer displays? What can interface designers learn from semiotics?
- What are repercussions of informatics on semiotic research? Does informatics constitute an empirical basis for semiotics?
- How does interactivity reflect the semiotic condition of informatics?

The second set of questions was:

- What does semiotics offer to informatics? What is a minimal amount of semiotics that should be studied in informatics? How should that be studied?
- Do semioticians welcome informaticians in their field? Why and what for? Can informatics contribute to advance semiotics?
- Does the concept of a sign contribute to an understanding of information, communication, organization?

Despite a strong attempt by all participants to move from their field of specialization towards that of the others, a gap was felt by many. In a coarse simplification, the gap could be described as the difference between a need for constructive design, and an interest in theoretical description. The intriguing elegance of semiotics became as obvious as the ostensible simple-mindedness of software problems that could be discussed within the confines of such a seminar. Interactivity and the quasi-autonomous state that signs take on when processed by computer, proved to be the two central fields where informatics and semiotics are bound to meet and probably give birth to some really interdisciplinary research. The challenge lies in combining a constructive engineering approach with an interpretative humanities approach. Such a combination will definitely advance our understanding of *design* and of *media*. –

As the list at the end of this report documents, there were 38 participants (8 of them were women) representing 12 countries: 18 participants from Germany, 6 from

Denmark, two each from Brazil, Great Britain, the Netherlands, and the US, and one each from Australia, Canada, Finland, Norway, Romania, and Sweden. The actual sequence of presentations, demonstrations, and special discussions developed on a day-to-day basis. There were 29 presentations, 27 of which are documented by abstracts in this report. Abstracts were provided by participants after their talks. Participants were given a chance to edit their abstracts before they went to print. We kept the sequence of presentations as they were delivered to the seminar. In addition to the abstracts, there is a list of „problems and questions" that was produced during the seminar in response to a request.

One highlight of the seminar was the presentation, by a group of informaticians, of a typical short sequence of interactive operations by the user of a graphical user interface. That sequence later became referred to as „Dag's problem". Presenters invited the crowd to semiotically analyze Dag's problem. After short hesitation, this effort produced some remarkable suggestions and insights. Due to time limitations, the exercise had to be cut off but everyone was asked to produce a written analysis of Dag's problem. Unfortunately, only two such analyses were handed in – probably because the excellent wines of Schloß Dagstuhl prevented people from writing down their ideas.

Unanimously, the seminar was called a success. Its major result appears to be a cross-fertilization between the two fields. A number of concrete projects were initiated by small groups of participants. One textbook project got its final kick during the seminar week. The probability of a follow-up meeting is deemed high.

I gratefully acknowledge the tremendous support by the wonderful staff of IBFI, both in preparing, carrying through, and wrapping up the seminar. Angelika Hoppé deserves a hearty thank-you as the decisive force behind editing this report.

Representation in Semiotics and in Computer Science

Winfried Nöth

Representation has been a key concept both in the history of semiotics and in computer science, but while there has been talk about a „crisis of representation" in the postmodern theory of signs, computer scientists have unswervingly pursued their project to model the representation of knowledge by means of intelligent machines.

Representation is often a mere synonym of a sign in general. Sometimes it is a sign vehicle, a mental concept or schema, an iconic sign, or the relation of signification or denotation. Can computers be said to represent in any of these senses?

On the basis of C. S. Peirce's theory of signs, I argued that the processes taking place within the computer are in fact not only energetic processes, but genuine sign processes (processes of semiosis). The computer's internal capacity of decoding and transforming its symbolic input exemplifies the transformation of a sign vehicle (representamen) into its interpretant. From this perspective, computers establish relationships of signification. However, in such processes, computers do not establish object relations, relations of denotation. The computer has no „window to the world". However, a robot with perceptual modules to give a symbolic representation of the world, with action modules to move around in this world, and also with the ability to learn from its interaction with its environment, is in fact a semiotic machine which manipulates signs not only by establishing relations of signification, but also by the ones of denotation.

Semiotically Relevant Aspects of Informatics

Solomon Marcus

1. Informatics is so genuinely related to the other information fields, that it is necessary to adopt as a framework the general paradigm of *information*. We take an historical approach, by distinguishing five different waves of information and computation fields and problems. Since semiotics too emerged in the context of information fields, corresponding interactions have to be expected.

2. Towards the middle of the 20th century, it became clear that the old paradigms of matter and energy are no longer able to cope with many of the new evolutions in science, technology, and society. A new paradigm had to be invented and it received the name *information*.

3. The *Information Era* can be segmented into five waves. The *first wave* in the thirties, but mainly in the fourties, is dominated by a mathematical and engineering perspective. The logical preliminaries of computation were clarified, under four different, but equivalent variants, suggesting the general idea of an *algorithmic semiotics*. Recursiveness became the model of an intuitive process applied to some primitive operations and suggesting a similar procedure in approaching iterative sign processes. Turing machines represent a semiotic event: calculation is no longer obligatoryly related to numbers, it concerns abstract qualitative symbols. The first programmable electronic computer (von Neumann) is challenging us to invent suitable ideas of semiotic machines, covering the whole evolution of computation devices, from the old abacus till the most recent supercomputers (a first suggestion in this respect is due to Nadin, *Semiosis* 1977). Cybernetics called attention to some isomorphic sign processes in animals and in machines. Shannon's information theory lead to the discovery that *information* can be separated from *meaning*. The λ-calculus (Church) developed the idea of an indefinite semiosis, later exploited by LISP: one can write programs able to work on data-programs and to produce result-programs. Shannon's theorem calls attention to genuine restrictions operating on different components of the communication process: message, source, code, channel, noise. Molecular genetics shows that heredity is a semiotic-informational itinerary from DNA to RNA to proteins to metabolism.

4. The *second wave*, related to the fifties, is bridging science, engineering, and the humanities. Chomsky generative grammars point out an isomorphism between language competence and logical competence and bring a new perspective to the study of learning processes, viewed rather as innate-acquired interactions. AI and Cognitive Science articulate the computational, the logical, the biological, the linguistic, the psychological sources of information science, adding to them the perspective of cognitive philosophy.

5. The *third wave*, in the sixties, includes: The emergence of the theory of programming languages, whose syntactic behavior proves to be isomorphic to the syntactic behavior of natural languages. Kolmogorov and Chaitin initiate the algorithmic information theory, by adopting an *individual* perspective in the study of randomness of strings, in contrast with the previous *global* perpective due to Shannon. Carnap &

Bar-Hillel begin the study of semantic information, while Zadeh initiates the study of systems with incomplete information (fuzzy set theory). In the same period, semiotics emerges, reaching however slowly its awareness as an information field. Many works are devoted to the semiotic status of programming languages, not yet clarified. The phenomenon of parallelism (see Lindenmayer systems) is challenging the sequential nature of computation. Informational, algorithmic and computational complexity are suggesting the problem of various possible concepts of complexity of a sign system and of a semiotic machine.

6. The *fourth wave*, in the seventies, is dominated by both *complexity* and *self-organization*. Synergetics points out some rules by which order arises in complex systems; catastrophe theory proposes a common pattern for biology and linguistics, under the form of discrete singularities in a context of continuous development; feed-back processes, in wich the same operation is carried out repeatedly, the output of one operation being the input of the next one, lead to the fractal geometry of nature. The science of chaos points out the yet unseen behavior of nonlinear dynamical systems, and reminds us of von Neumann's remark that in contrast to simple machines, complex machines may have an unpredictable behavior. Complexity problems in semiotic machines may lead to similar difficulties.

Communication emancipates now from information, while the latter emancipates from objective information. In this respect, the semiotic perspective becomes clear in Nauta's 1972 book. Dissipative structures, the rhizome, and autopoietic structures point out the conflictual nature of life processes. The traditional paradigm of objective information gets into a crisis. The logarithmic semiotics of man-nature interaction is contrasting with the antilogarithmic semiotics of man-computer interaction. The deterministic-nondeterministic dilemma (P = NP?) may suggest similar dilemmas in semiotic machines.

7. The *fifth wave*, in the eighties and nineties, includes: The development of computational biology (with the challenge of possible DNA-like computation); Chaitin's theorem giving an informational interpretation of Gödel's incompleteness theorem; the emergence of the study of cognitive models and cognitive metaphors; the new perspective of including quantum physics in an information-semiotic perspective (H. Stapp 1993, K. Svozil 1993) and, last but not least, the semiotic limitations resulting from the fact that most recursive functions are not primitive recursive; most strings are random; most dynamical systems are chaotic etc.

Semiotic Principles and Systems in Human-Computer Interaction

Udo L. Figge

Energy or matter taken in by a system can be processed as an environmental influence by a peripheral component sensitive to such influences, producing a particular state of this component. This processing can be continued, resulting in a particular state of a component that cannot be directly reached from outside and that is autonomous in this respect. The resulting inner state is always quite dissimilar from the original peripheral state. In such cases the energy or matter is taken in from a medium and processed as a sign according to a *receptive semiotic principle.* On the other hand, certain states of inner components of a system, which are not perceptible to its environment and which are autonomous in this respect, can be coupled with states of peripheral components in a way that certain forms of expending energy or excreting matter through these components function as manifestations of these inner states. In such cases, energy or matter is transmitted to a medium as a sign according to a *productive semiotic principle*.

Both semiotic principles are realized by *semiotic systems* as subsystems of more comprehensive (biological and technical) systems. *Human-Computer Interaction* is based on a combination of four different semiotic systems. A human can manifest the need to have a problem solved by a computer according to the productive semiotic principle, normally by expending mechanical energy through movement of his hands. This results in certain states of an input device of the computer, which then undergo further processing as instructions, triggering off an internal working state according to the receptive semiotic principle. The computer can manifest certain inner states, typically working results, through states of output devices according to the productive semiotic principle, mostly by setting free optical energy. The human can process this energy by his sense of sight and then gain knowledge according to the receptive semiotic principle.

The Semiotic Engineering of Concreteness and Abstractness

Clarisse Sieckenius de Souza

Most successful User Interface Languages have been designed following two important guiding principles: task specificity and direct manipulation of graphic objects. Programming languages, in their turn, have often been pursuing such goals as general-puposeness and efficient symbolic manipulation of linguistic objects. When it comes to End User Programming Languages, features that are apparently in conflict with each other must be combined to allow non-programmers to write extensions to existing applications and to design and implement completely novel applications and programs.

In view of increasingly strong evidence for the need of engaging users in the programming of software tools, some researchers advocate that the interface language should become a real programmming language, whereas others remain skeptical about this possibility. We propose that an integrated interface environment should be designed within a unified semiotic framework that accounts for interaction with and specification of, computer applications. A brief case study about a successful text editor and its extension language reveals some of the features this unified semiotic framework should have and provides important interactive and programming profiles for research in interface system design. The integration of both profiles is the object of Semiotic Engineering of concreteness and abstractness in computer-mediated interpersonal communication via software applications.

Combining Informatics and Semiotics. Four Questions.

Peter Bøgh Andersen

1. Why should informatics do it?
2. Why should semiotics do it?
3. How should they do it?
4. Should only the two of them do it?

Answer to 1: Why should informatics? Informatics should do it because of the way computers have developed. They are increasingly used as media, they acquire a still richer repertoire of means of expression (sound, picture, movie), and they merge with analogue media. In addition, the integration of computers into all parts of society requires informatics to integrate analyses of technical and social systems. Semiotics has the advantage that its concepts can be applied to aspects of the human organization as well the technical system, namely in their capacity of signs. Psychology and sociology can describe users, but not computers. Finally, many techniques used in system analysis are really disguised semiotic techniques (conceptual analysis, e.g.), and therefore will get a better theoretical foundation.

Counteranswer to 1: Why should informatics not? Because the theory is too difficult to apply, it deals only with a subset of relevant issues, and says trivial things in a complicated way.

Answer to 2: Why should semiotics? It is a professional duty of semiotics to take an interest in any new type of sign systems, especially when the sign system has the social importance of computers. Its theoretical foundation can be broadened by entering new data, and – well, it is easier to get money for research in computer systems.

Counteranswer to 2: Why should semiotics not? The subject of computers is too narrow to yield insights, and computer science often tends to put the *how* question before the *what* and *why* questions. Finally, if successful, it runs the risk of being gobbled up by technology.

My opinion is that the pros outweigh the cons.

Answer to 3: How should they combine? Making a new field is, in my opinion, not realistic, and neither field should be seen as a subfield of the other. Instead, both fields should maintain their own identity, but create windows within themselves, enabling each part to view a part of the other discipline. Borrowing concepts is a natural thing to do (and is in fact extensively done), but the borrowed goods will soon be digested and integrated into the reproductive cycle of the borrower. The original source of the loan-words will soon not be able to recognize the loan. This goes for both disciplines, and may not necessarily lead to novelties. Instead, the two disciplines should enter into a regular interaction and communication, irritating and challenging each other, while remaining strange, and therefore interesting, to each other.

Answer to 4: Should they do it alone? The advent of nets may create a need for a *menage à trois*. Global nets are not controllable and the individual contribution will keep entering into new physically different texts by linking. Nets seem to acquire an existence of their own, they feel like a natural thing, such as a landscape, an ocean or a river, that cannot be controlled by the individual. In order to understand the evolvement of nets we may need concepts neither furnished by informatics nor semiotics.

Information Systems Concepts Based on Semiotics

Wolfgang Hesse

In its work, the IFIP Working Group FRISCO (FRamework of Information System COncepts) follows the objective to better understand, clarify, and define Information System (IS) concepts including

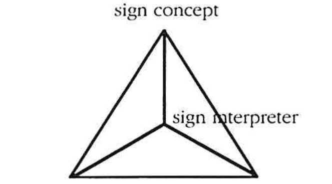
- their meaning and correspondence to real-world phenomena (semiotic aspects),
- their structure, (internal and external) dependencies, and relationships (modelling aspects), and
- their position, need, and usefulness in the organizational and societal context (systemic aspects).

This work implies the classification and establishment of a uniform IS terminology including terms like

- sign, meaning, language, knowledge, information, data, communication (semiotic terms)
- model, actor, activity, event, state, process, entity, object, attribute, relationship; system, information (modelling terms)
- system, organization system, information system (systemic terms) etc.

One of the fundamental questions for the group was to agree on a joint philosophically grounded position. In our situation, where we are dealing with an area in which many concepts are not yet well established and still disputed, a (moderate) constructivist approach seemed to be appropriate. That means: We find ourselves participating in negotiating processes going along with the construction of a piece of reality.

Semiotics is considered an indispensable foundation for building a terminological framework. For our purposes, we have transformed the traditional semiotic triad to a tetrahedron of the following form:

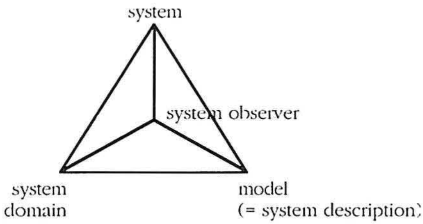


The „sign interpreter" in the center of the triad reflects the constructivist approach: There is no perception of objects, conception of ideas and thoughts, formation and interpretation of sign tokens, without an interpreter or observer. In a particular context, the interpreter might also be a group. Whether he may even be a machine, is a point of debate. With the help of the tetrahedron, many concepts can be defined more clearly and their interrelationships can better be illustrated. Examples are the definitions of

- *knowledge*: the totality of perceptions, observations, and experiences of a person[1], or a group of persons, of him/her/themselves and his/her/their environment;
- *information*: a piece of knowledge of a person[1] on things of a (conceived or imagined) world;
- *data*: symbolic representation of information.

Referring to the semiotic triad, „knowledge" and „information" have their focus on the „sign concept" vertex, whereas „data" is primarily located in the „sign token" vertex.

Another example is the triad for „system":



„System" is considered to be a mental construct and thus located in the „sign concept" vertex. It may be represented by a „system model" (some denotation or „token" of the system) and it refers to a „system domain", i. e. its counterpart in the physical or mental world.

[1] A more comprehensive (and less cautious) version would replace „person" by the neutral „actor" (also implying non-human actors).

The Computer as a Semiotic Machine

Lúcia Santaella Braga

After a brief examination of the descriptive names that have been used to characterize the functions of the computer, such as a tool, a tool-kit, a device, an instrument, a machine, an equipment, an apparatus, and a medium, this paper discusses the multiple meanings of the word medium, analyzing what it means to consider the computer as a medium from a semiotic point of view.

Peirce's definition of the sign is examined on its most abstract level and applied to the understanding of the computer as a semiotic machine. According to the logical form of semiosis described by Peirce, there are two senses for sign: an extensive and a specific one. In the extensive sense, the sign is a synonym, or mediation, functioning as an epistemological model of the world. The computer can be understood as a sign in this extensive sense. In its more specific sense, the sign, for Peirce, is the middle term in the triadic relation of sign–object–interpretant. Here, the three logical functions of mediation, determination, and representation are analyzed. The computer is a very complex machine with signic facets. As such, there are at least three basic semioses where, (1) the computer, or better, the executable program performs the role of the interpretant, the program is the sign and the subject domain is the object; (2) the computer is the sign, the program is the object and the outputs are the interpretant; (3) the computer is the object, the outputs are the sign, and the interfaces with users are the interpretant.

Software Systems

Inger Lytje

Software systems can be conceived of in one of the following ways:

(1) dynamically as „pure" information processes comprising storage & retrieval of information, computation and interaction,

(2) as representation, and

(3) as intervention.

Seeing representation as a semiotic process, three kinds of semiosis can be identified: model, text, and knowledge base. In computer science, software systems are usually considered models, and modelling phenomena and concepts from the real world according to a denotational semantics is the only kind of semiosis that is taken into account. I suggest that software systems should rather be seen as text, having a certain structure, carrying a message or a theory, and writing new contexts. Seen as intervention, software systems are situated in specific contexts – in space and time and with respect to users and the users' tasks. Different kinds of intervention can be identified: regulating, fascilitating, organizing, learning, enjoying etc. The way software systems are conceived has some consequences for the system development process. Considering software systems as text, e.g., implies that authoring rather than engineering should be the metaphor we use about the system development process.

Computation, Semioses, Anticipation

Mihai Nadin

The primitive age of computation is coming to an end. Increased performance in almost all aspects of computation and progress in programming (in particular the paradigm of object-oriented programming) make a new qualitative perspective possible. In addition, computation has diversified enormously, extending into the genetic, molecular, chemical. As a result, many kinds of new challenges need to be addressed. In this context, semiotics has yet another chance to ascertain itself. Based on its particular subject of inquiry – signs and sign processes (also known as *semioses*) – semiotics should address the new possibilities opened by advanced computation technology.

The subjects to which semiotics can contribute include:

- interactivity,
- variety of data types (e.g., multimedia and related),
- virtual interaction (work, study, research, etc.),
- hypertext and associated applications,
- computing metaphors (to replace the dominant file model).

In order to fulfill such a high order of expectations, semiotics has to interact with computer science at levels where such interaction is meaningful. Instead of the „cosmetics" of user interfaces, semiotics could initiate interface considerations at the level of the design of new operating systems.

As the person who introduced the notion of the computer as a semiotic machine, I should explain the two intentions I spoke of in 1977:

- the possibility to operate on semiotic entities,
- the theoretic/conceptual notion of a semiotic machine as pertaining to the generation or constitution of signs, sign awareness, sign interpretation.

This notion of the computer as semiotic machine seems to have assumed a life of its own. Regardless of its use, it introduced the important element of anticipation. In short, anticipation concerns systems in which the current state depends upon a future state. It is important to understand that user interface is in anticipation of computational processes that lead to meaningful interpretations of data. In order to transcend the practice of designing languages of interaction, which have a merely cosmetic function, we need to deal with the complex mechanisms of anticipation. As computation develops beyond its infancy, anticipation becomes one of its inherent dimensions.

Interpretant, Interpreter and Computer Signs

Jorge Bogarín

1. The distinction between general interpretants and individual interpreters is very important in Peirce's semiotics. One of the reasons is that he particularly avoided doing psychology in his ideoscopy and in his semiotics. Nevertheless, he definitely does include minds in all his definitions of signs and semiosis.

2. We do not need yet to expand our concept of interpreter to include digital computers. There are sign processes going on in computers, but only because human beings build the hardware, write the programs, and give the input (or build some quasi-sensory input devices). But I am not confining interpretation of signs to human beings, not even to biological systems. If we do not want to defend an animistic view, we have to accept the possibility that machines could someday, as living organisms did, make a *dialektischen Sprung* from syntax to semantics.

3. Within Peirce's system of ten classes of signs, the characteristic mode of representation of formalized sign systems – like Turing machines – corresponds to the rhematic symbolical legisigns. If we use the list of 66 classes of signs, such systems belong to the class of formal argumentical pragmatical indicative conventional relative symbolical collective copulant legisigns.

Iconic and Notational Models in Formal Logic and Automata Theory

Julian Warner

This paper was concerned with the contrast between models used in formal logic and automata theory. It announced itself not as declaration of intent, asserting the potential value of semiotics to informatics, nor as an empirical study, but as a consideration of theoretical issues in informatics informed by a semiotic perspective. Its understanding of semiotics was influenced by de Saussure, Barthes and Eco, not by Peirce, except in some qualifications made, and this affected both the terminology (*notational* might be replaced by *symbolic* in Peirce's terms) and other aspects. The contrast between de Saussure and Peirce could be said to echo the fields about to be described: of two adjacent but no fully intersecting areas of study.

Automata theory was valued for the understanding of the computational process which could be obtained by avoiding real world complexity and which could then be brought back to bear on the practical world. The separate development of working computers and automata theory was noted and it was suggested that the distance between these could be reduced by recognizing the technology as a human construction critically dependent on the exactness offered by graphic signification for its development. It was suggested that formal logic and automata theory had also tended to develop separately. The contrast between these had been encapsulated by Gödel as a dichotomy between a definition relative to a given language (for formal logic) and an absolute account of the computational process (for the Turing machine model). This dichotomy was questioned: Wittgenstein's almost humorously emphatically syntactic view of logic (whoever dreamt of defining a bracket?) was alluded to; the presence of iconic elements in a notation was discussed (for instance, q_2 q_3 ... q_n are iconic with respect to q_1); an implicit connection to the connectives of formal logic in the Turing machine was noted; and analogies between the methodology of construction and reasoning used in formal logic and automata theory were noted.

It was suggested that the models used in automata theory could be regarded as an alternative formulation for formal logic as well as/instead of an absolute account of the computational process. Automata theory and formal logic could both be regarded as socially constructed modes of discourse. The curious nature of diagrams, posed between the evidently conventional system of writing and the seemingly more natural iconicity of aspects of the pictorial, was briefly discussed. The denial of the possibility of innocent perception, not mediated by a paradigm or sign system, in the philosophy of science (Fleck and Kuhn) and in branches of semiotics, was discussed. The final suggestion was that the contrast between the models used in formal logic and automata theory could be understood as a contrast between types of sign (along a continuum from notational to iconic), not as a dichotomy between a relative definition and absolute formulation, and this was a more satisfying formulation of the contrast.

A published version of this argument can be found in Julian Warner: From writing to computers. London and New York: Routledge, 1994 (pp. 65-74, 101-103, 107-112, and 115-116).

Computer Support for Human Creativity

Ernest Edmonds

The paper presents some current problems in human-computer interaction that might obtain help from semiotics. My research concern is with the problem of developing computer support for human creativity. Thus the concern is less with the results generated by the computer and more with the thoughts generated in the human. The question can be put as „which behaviours should the computer exhibit in order to support creative thought?". Of course, in the real world, „computer" should be seen as including, e.g., the network and the people connected through it. However, for this paper, I will concentrate on just two problems that we face today in the case of interaction between one human and one computer.

The top-level concern is with the „human machine" machine in the context of creative tasks. The approach is to study empirical evidence and conduct empirical studies (of a variety of types) of creative people, such as designers working on new design concepts. From this guidance, it is possible to formulate some of the requirements for the decision support systems.

The paper discusses two characteristics of creative work that each present significant challenges to human-computer interaction.

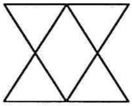
1. *Reflection in action*

Donald Schön has shown how professional workers, such as designers, reflect upon what they do whilst they do it. The key point is that they learn about their methods and strategies during practice. Hence practice is always changing. In terms of computer support this introduces the problem of a very heavy maintenance requirement. In fact the only solution is for the user to maintain the system. Thus the user needs access to deep aspects of the computer system. The need is, therefore, for a solution to the „end user programming" problem. I argue, however, that the use of the word „programming" is not helpful. Programming is about how to compute things whereas our user is concerned with what is the case. The user wishes and needs to communicate with the machine about the domain of the knowledge. In other words, the need is to approach end user programming from the user's domain and its language and constructs rather than, e.g., program control constructs. This is known as the problem of designing and building „Knowledge Support Systems". Perhaps, semiotics can help?

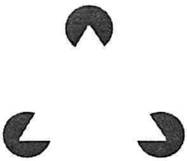
2. *Emergence*

At a much lower level of granularity, we can readily note that during creative processes new and unexpected concepts emerge. This can be illustrated in terms of emergent shapes. To take an example, consider two triangular objects as in the following sketch:

Now, place them together like



We see the emergent diamond shape. If we consider doing this in a computer drawing system, we will certainly see the diamond but the computer will have no knowledge or representation of it. Thus it may be ridiculously hard for us to, e.g., copy it. In certain systems, region growing algorithms are provided that could find this particular emergent shape, but consider the triangle in the following well-known sketch:



The problem is to understand the process of emergence and to provide support for the user so that the computer does not get in the way of the discovery of emergent concepts.

How Do Signs Get Meaning in Computer Science?

Dirk Siefkes

Individually and collectively living beings develop through two interrelated types of interplay: between the living (process) and its representation (form), and between the general and the particular. In our mind, fleeting thoughts harden into concepts which may carry new thoughts, generalizing into imagination and knowledge, resp. Thus conceptual learning interleaves with corresponding changes in imagination. The process is strongly influenced by communication, where we seemingly exchange words, but join in imagination as well: It is stories we understand – language pieces that make sense.

Second, learning involves feelings and values, intuition and conscience, in a similar way. We communicate visually, through gestures and other signs. Feelings and thoughts come together, conceptual and emotional growth are interwoven. Through perception and interaction, individual learning depends on development in social groups, from where it spreads out into culture and comes back to both support and constrain the individual.

Thus semiotics without semantics and pragmatics does not make sense: Representations, thus signs, get their meaning in social development. Problem: How do verbal and visual representations relate? What role do diagrams play?

In science, we order the world through generalization of experiences. We use descriptions for this. A result are theories and formalisms made up of rules and definitions. This strong bias hinders the interplay and thus learning, and leads to both anxiety and power phantasies when people face formalisms and machines. This is particulary true in traditional computer science, since formal objects fit into a general environment only where human particulars do not count. My prescription is to work locally, so I can know what is going on. In both software development and teaching prototyping is helpful. Find out what orients people towards certain goals, and thus determines the development of computer science.

Computer related art does not appear to be „about" anything; there are no themes, only strategies. Why care for a medium, if there is no message?

Some relevant publications:
W. Coy, F. Nake, J. Pflüger, A. Rolf, J. Seetzen, D. Siefkes, R. Stransfeld (Hrsg.): Sichtweisen der Informatik. Braunschweig: Vieweg 1992

Dirk Siefkes: Formale Methoden und kleine Systeme - Lernen, leben und arbeiten in formalen Umgebungen. Braunschweig: Vieweg 1992

Dirk Siefkes: Ökologische Modelle geistiger und sozialer Entwicklung. Beginn eines Diskurses zur Sozialgeschichte der Informatik. Discussion Paper. Wissenschaftszentrum Berlin für Sozialforschung FS II 95-102, 1995

Semiotic Foundations of Communication

Roland Posner

It is shown that the traditional opposition of natural signs (symptoms) and conventional signs (symbols), advocated by Plato, Locke, and Husserl, is misleading and has to be replaced by a new logically based hierarchy of sign concepts.

Four increasingly specialized sign types are introduced on the basis of the interpreter's reactions: If something causes the interpreter to respond with an overt or covert change of state, it is a *signal*. If something causes the interpreter to have a specific belief, it is an *indicator*. If something causes the interpreter to believe that its producer is in a specific state, it is an *expression* (of the state). If something causes the interpreter to belief that its producer has the intention to produce a further action, it is a *gesture* (announcing that action).

These sign types occur on increasingly complex levels of sign use: Responding to an inarticulate cry by assuming that its producer is in pain, is taking it as an expression and involves what we call *sign interpretation*. Uttering such a cry in order to bring about such a reaction amounts to expressing, and involves what we call *sign production*. When the cry is stylized (e.g. manifesting some language sign such as German „aua", English „oh", or French „ai"), it is not only received as an expression but also interpreted as a product of an expression attempt (*interpretation* of *sign production*). When such words are pronounced without crying, they are no longer taken as an expression, but only as indicating expressing (*indication* of *sign production*). Expressing one's state by merely indicating an expressing attempt is a case of *communication*. On the level of communication, signals become *imperatives*, indicators *assertives*, expressions *expressives*, and gestures *commessives* (in the sense of John Searle).

Now, expressing is often taken as signaling to get help. When someone relies on this tendency and articulates an expression with the intention of having it interpreted as a signal, he performs *indirect communication*. By uttering an expressive he produces an imperative.

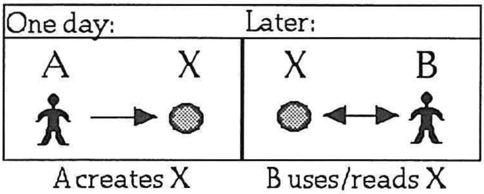
It is demonstrated that all these sign concepts can be defined on the basis of the predicates *believe*, *cause*, and *intend*. This approach also allows an explication of the concepts of *action*, *interpretation* and *indication of action* and *declaration*.

The given hierarchy of sign concepts is exhaustive, covering all possibilities of semiosis, but open to further differentiation. It is offered as a theoretical basis for the reconstruction of sign processes in Artificial Intelligence research.

Understanding Interactivity

Dag Svanæs

In the late seventies and early eighties the developers of Xerox Star, Lisa, and the Macintosh made it easy for ordinary users to make use of the computer. My aim is to contribute to doing the same to programming in the nineties and after. As I see it, this requires a deeper understanding of interactivity. Let me give an example to clarify: Consider a user A constructing a piece of software X to be used by user B.

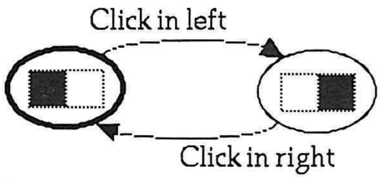


What is the best way to describe this? One way is to see it as user A composing a message X and sending it to B. This is the semiotic approach. This leaves out the fact that the message is interactive. It contains a potential for interaction. Constructing interactive software is similar to writing, but the writer does not take the reader on a journey with start and end, but creates a possibility for a lot of journeys.

In this manner it is more similar to the situation where a designer A is creating an artefact or environment for B to be used/lived in. The problem here is that we lose the sign aspect – artefacts are not usually intended as carriers of meaning. We see that neither communication metaphors nor material metaphors give a complete account of the phenomenon.

Where should we look for inspiration for a new understanding? My current answer is phenomenology (Merleau-Ponty), abstract art (Kandinsky), and detailed empirical studies of how interactive artefacts are perceived.

To find out how interacting is perceived I have constructed some abstract interactive artefacts and observed non-programmers exploring them. An example is indicated by the following state transition diagram.



From this experiment I have found the computer science concepts object, state and time to be non-intuitive to non-programmers. This new understanding I will apply to the design of end-user-programming tools.

Worlds, Modelling, and Semiotic Engineering

René Jorna

In the development of management information systems (MIS), the mainstream approach is to start from reality, make a depiction of it into functional terminology and end up with program(ming) code. The program is a representation of the situation in reality. There are at least three major problems (issues) with this widespread view.

The first is that it gives no answer to the question of decision alternatives in modeling. Why is one model better than another model? „Ultimate" criteria are difficult to find in classical information system methodologies.

The second is that the perspective of the task being executed by a cognitive system or human information processing system normally is not taken into account. This has harmful effects for the success and acceptance of the resulting computer.

The third problem is one that I consider to be the most fundamental. It is the debatable ideological view that building information systems and programs means copying reality with the help of what has been called „ontological engineering" (Lenat & Guha, 1990). In contradiction to this I would like to call the enterprise of building information systems a matter of creating worlds (in the sense of domains or models) with the help of several types of sign structures. The name for this approach is „semiotic engineering" (also see C.S. de Souza, 1993).

Semiotic engineering is the construction, the design and/or development of all kinds of sign structures. Sign structures can be at the object level (symbols, program codes, characters, widgets, icons, etc.) or at the meta level (artifacts like organisations, cars, etc.). Meta level sign structures can be interpreted by intelligent systems as sign structures at the subject level. The results of semiotic engineering consist of artifacts, models and other configurations.

We applied the semiotic engineering approach to the domain of scheduling and planning (staff, production and transportation). The semiotic engineering approach resulted in a so-called Scheduling Expertise Concept (SEC). SEC takes into account aspects of task domain, task execution, task context, and task outcome. Furthermore, several scheduling supporting systems have been built, such as ZKR (nurse scheduling), IPS-Rabo (banking staff scheduling), EPL (transport planning), Planmax (construction planning), etc. (van Wezel, Jorna & Mietus, 1996).

Pliable Surfaces:
A Local Magnification Interface for Exploring Visual Information

Sheelagh Carpendale

Multi-scale views make more effective use of screen space by allowing magnification of areas of interest while compressing other sections. Studies supporting this approach show both human predilection for presenting and storing information in this manner, and increased user performance. However, user comments about disorientation and confusion are also reported. One possible explanation is that the user is trying to comprehend both the information itself and what magnification or distortion that is present. This distortion is the result of mathematics that the user may well be unfamiliar with. To avoid this situation where the object (the mathematics) of the sign vehicle (the distorted image) is not part of the user's world, we have chosen to represent it as the cumulative result of the user's own actions. To achieve this the information is placed on a 2D pliable surface which is manipulated in a 3D space. This provides the functionality of multiple arbitrary shaped foci and an intuitive interaction metaphor where the user pulls areas of interest closer in order to see them better. However, unless the surface is visually convincing it will remain as part of the mathematical solution instead of informing the user.

To make this surface comprehensible we focus on the representatum. The question arises of what visual primitives can be used. While work on this issue exists for other media (Bertin's for 2D graphics) there is no corresponding reference for the computer. Computers allow both areas of extension (time, motion, animation and interactivity) and limitation (Ware indicates orientation becomes ambiguous). However, partly because the computer is still rapidly developing as a media it is a difficult task to discover what its visual primitives might be. As a temporary solution we have followed Ware's suggestion to consider human perceptual capabilities. The current options include using perception of shape from shading, in conjunction with a cartographic grid. Together these provide both an intuitive reading and an approximate quantitative reading of relative magnification and compression. Having provided a spatial metaphor for the distortion complete with visual cues to aid in its reading, the question remains to what extent this increases the user's ability to understand multi-scale views. For instance, does this allow the user to interpret relative distance across an area of distortion. Alternatively, as the original image is the actual information of interest, the shading and the grid are just intended to support reading of the multi-scale view. How discrete can they be and still remain effective in visually defining the 3-dimensional form of the surface?

Informatics is the Formal Branch of the Social Sciences when Viewed in the Framework of Semiotics

Ronald Stamper

Information to be of any value must produce a social effect; this is evident from the semiotic framework showing that signs are formed by adding to the physical tokens layers of different properties that enable them to generate or modify features of our social reality. Informatics has confined itself to the three lower levels where its formal methods have been highly successful. I want to indicate how to extend informatics from the engineering of information-handling artifacts into the formal science of information on all levels.

Social level: norms, attitudes, commitments, contracts,
Pragmatics: intentions, conversations, power,
Semantics: meaning, validity,
Syntactics: structure, form, deduction,
Empirics: variety, channel capacity, entropy,
Physical level: signals, tokens, place, material/energy used, cost.

All signs have a physical form and consume some scarce material or energy, although usually in trivial quantities, so that on this physical level we account for the material costs of information. Whatever the physical realisation of the sign-tokens, they must be instances of sign-types that can be repeatedly recognised and these have to be transmitted and stored with sufficient empirical reliability before they can be used to make syntactic structures of any sophistication, such as software and databases. These three lower levels are the primary concern of computer science but the true character of this discipline can only be understood by noting that the signs handled by computers and telecoms have no value unless they are also correctly used on the upper three levels.

The upper levels concern the functions of signs that can only be established by human action. Well-formed signs ad sign structures can be given meanings – the concern of semantics. Meaningful signs can then be used to express the intentions of people – the concern of pragmatics. Finally, interpreting the communicated intentions results in the changing of the dispositions people have to act in various ways, whether on particular matters or according to principles applied universally. Collectively, these dispositions constitute what we experience as our social world.

The norms, the universal dispositions, can be used to model the social fabric that persists through the shifting attitudes concerning particular matters. This observation was the starting point for a study of how to understand social structures as systems of norms. For the practical purposes of engineering information systems, this concept leads to elegant solutions because the norms define the information required for the functioning of the social structure: every sign that plays a social role does so by virtue of its place in the condition of some norm and every sign that is social in origin is the direct or indirect result of the consequent of some norm. What this model suggests is that we should consider signs and norms together as the two „elementary particles" of the social domain.

Our research on formalising norm structures and their relationships with signs led us to the problem of semantics which then raised the more basic problem of what underlying metaphysical position to adopt. The work made limited progress so long as we maintained the objectivist position. But a break-though came when we adopted two metaphysical principles quite different from the naive realism that informs most work in computer science and data base work in particular. They are:

1. There is, for all practical purposes, no reality without an agent;
2. The agent only knows about reality through his/her/its behaviour.

In fact, the agent does not perceive a ready-made reality but has to construct that reality in the process of learning to perceive. The agent must take responsibility for the reality so defined, so the agent is always a person or a group. These principles give rise to a wff

agent behaviour (this is a wff in Norma)

where the behaviour should be thought of, in the sense introduced by Gibson in his psychology of perception, as an „affordance" which is a repertoire of behaviour that is invariant while the agent has access to it. By extension, norms define repertoires of social behaviour. We can distinguish the realisation of particular instances of these invariants from the realisation of universals which are the abilities to realise particulars. Working at the universal level we can model the dependencies of one invariant upon another. Given

(agent behaviour)

as a modified agent, it can then enjoy some other behaviour

(agent behaviour) behaviour – e.g. Stamper upright walk.

This illustrates a particular realisation of the persistent ability to be upright or to walk. Each of these abilities is also an affordance of a universal kind, like a data-type, enabling us to use a compact graphical notation for Norma (a language for norms and affordances), as illustrated in this social case:

```
Society _____ nation _____ company ======= contract - - - - propose
                   \_____________________________/
```

It shows that instances of items on the right can only exist during the existence of appropriate „ontological antecedents" to the left. The broken line indicates that the antecedents is not the contract but a sign for the contract. Each of the universal and particular invariants has a start and a finish event, each of which is governed by the authority of an agent or a norm, a device that links the formal structure to the social system in which it is embedded.

This model has nothing particular to do with computers; it is concerned with a social structure viewed from an information or semiotic perspective. The figure shows an ontological structure which is the basis for a formal analysis of meaning in terms of behaviour (affordances). The pragmatic aspects can be handled by treating intentions as affordances involving the use of signs (as in the instance of „propose" in the figure). The social level of analysis is provided by the agent term in the wff which, in the illustration is Society at the root, Society being the source of our commonsense meanings. But the nation and the company, or pair of companies entering into the contract, are more specific agents. Every instance of every affordance has its start and finish events with which are associated responsible authorities in the form either of agents who exercise discretion or of norms that distribute the discretion over a number of agents.

The resulting formal structure can be interpreted directly by computer. And an „ontological schema" of the kind illustrated above can be used as a schema for a semantic, temporal database which generates a technical information system to serve the people in the problem domain. The version of the system which also handles the norms we call a Normbase because it allows all the social norms to be separated from the computer programs and treated as a purely social resource.

The practical value of semiotics in informatics is demonstrated by the fact that, using these methods, we have achieved, in comparison with a package for a large administrative system, reductions in development costs by a factor of three even on our first attempt, and reductions in support and maintenance costs by a factor of seven over a period of several years.

These ideas are presented in a chapter of the book Signs at Work, edited by Andersen & Holmqvist, de Gruyter 1996, in which they are applied to the analysis of concepts on each of the semiotic levels introduced above. By means of this kind of formal analysis, I hope that semiotics can also be given greater precision than seems the rule today.

The study of information outside its social framework is incomplete. Inside that framework informatics or computer science extends naturally into the social domain where it derives its value.

Joining Icon, Symbol, and Index in One Connectionist Substrate

Bo Pedersen

In order to create some common ground for semiotics and informatics, I propose a link between the semiotics of C.S. Peirce and the connectionism of Paul Smolensky. The link has its starting point in some of the central concepts of connectionism and the Peircean categories:

1:	quality	=	neuron,
	quale	=	excitation
2:	relation	=	connection,
	relate	=	weight
3:	representation	=	harmony function,
	representamen	=	harmony value.

Peirce uses these categories to derive his sign types. If we use the connectionist interpretation of the categories above in this derivation we get a connectionist interpretation of the sign types:

Icon: A pattern of neurons that can be compared to another pattern of neu rons without considering connections or harmony functions, that is, the pure (Euclideàn) distance between the patterns.

Index: A pattern of neurons that can be compared to another pattern of neu rons by connections alone, that is, without considering the harmony function and the pure distances between patterns. This is what Peirce calls „a correspondence in fact".

Symbol: A pattern of neurons that can be compared to another pattern of neu rons by calculating the harmony of the net with respect to these two patterns, and only this way, and because this is the only way it resemb les the „imputed" character of the symbol.

If we use these interpretations we can have different sign types represented in *one* net in a non-trivial way. If we also consider the tensor product representation of Smolensky, we can represent whole structures of this mix of sign types.

This could serve as a platform for making „semiotic programming" in the connectionist paradigm, or for analysing existing connectionist systems.

The presentation can be found in full length at: http://www.daimi.aau.dk/~bop/ and comments can be send to bop@daimi.aau.dk.

Two Paradigms of Modeling

Peter Stephan

The design of software implies the use of a different type of models. Besides the distinction of models according to computer science or semiotic terms there are underlying paradigms. Two paradigms are identified: The engineering paradigm and the experimental paradigm.

They imply different methodologies and have consequences for the possible function of semiotics and computer science in the design of software. To describe the two paradigms, two examples of modeling processes were discussed.

1. The engineering paradigm

- *Task:* To model a production process
- *Question:* Which processes can be described precisely enough so they can be implemented technically?
- *Methodology:* Analysis: identify relevant factors (e.g. resources: work, energy, information, capital) of a layout grid for measurement (e.g. intensity of time);
  Data gathering: measure and write data of resources into the grid;
  Structuring: define nodes as point of time, where certain measurements are supposed to be met, describe interdependencies, develop logical structure;
  Implementation: transfer found structure to hard- and software system.
- *Achievements:* controling, steering, planning

2. The experimental paradigm

(based on Hans-Jörg Rheinberger's conception: *Experiment – Differenz – Schrift* (1992))

- *Task:* To model a research process
- *Question:* How can we model the paradoxical activity of creating a setup that „delivers surprises"?
- *Methodology:* Generate hypothetical models, implement in a „representational space." Scientific objects are identified as: unbound fields of interest, ill defined, „Machine producing questions".
  Technical objects are identified as: limiting questioning possibilities, access to the scientific object, „Machine producing answers".
  Register traces (Derrida)
- *Achievements:* transfer scientific objects into technical objects, transfer technical objects into scientific objects.

Signs Without Message – The Topographical Dimension of Mediality

Georg Christoph Tholen

The task of a non-anthropological or not-instrumentalistic reduced theory of media has become very important since the use of the computer „as" a medium.

To develop a historical and systematic philosophy of mediality it is important to reflect the concept of a symbolic machine which elaborates the complex theory of Lacan (the model of the real, the symbolic order, and the imaginary). In combination with the de-constructivist method of J. Derrida (M. Heidegger's concept of *Technik*), I will give some fundamental aspects of a theory of the media in its role of transfer and transmission. This theory of mediality as transfer avoids the restrictions of anthropological and instrumentalistic definitions of the computer and of information.

Dynamics of the Web

Jeff Nickerson

Three topics are discussed:

1. The Eventual Effect of the World-Wide Web

The implications of the web on publishing and distribution are described. The question is posed in semiotic terms – what is the final interpretant of the web?

2. Association

We can treat the web as a directed graph and analyze for, say, strongly connected components. By looking for net traffic we can treat the web as a weighted graph and apply other techniques – network flow, and perhaps neural net techniques. We can then pose questions such as: what would be the harmony function for the web? We discuss hypertext links, and observe that links are associations based on similarity that become links based on contiguity.

3. Suggestion

The tendency of web multi-media is toward threedimensional immersion with kinesthetic components. Citing Peirce, we conclude that the tendency toward immersive technology makes the web a powerful vehicle for suggestion.

Face-to-Face Communication on the Internet – Redefining Computer-Mediated Communication

Anita Noupponen

I am interested in computers as communication media between humans, how this new media affects our ways of behaving and language use. In my contribution, I discussed the evolution of video-conferencing and the possibilities that real-time video brings to human communication and collaboration over physical distances.

Human communication on the Internet is still usually seen as text-based, faceless, lacking contextualization cues, nonverbal signals etc. The ordinary Internet users have, however, already been able to test and utilize the first Internet video-conferencing and audio-conferencing systems. These add several benefits of normal face-to-face communication to communication in computer networks. So, finally, Internet as the global village, acquires two crucial aspects of village life: seeing and hearing the other villagers.

„Traditional" video-conferencing systems do not support collaborative use of the communication channel, nor do they encourage new innovative uses, or spontaneous, informal communication. Internet video-conferencing systems like CU-SeeMe, developed by Cornell University, start from the other end. They are not restricted to special studios, and do not require dedicated hardware. CU-SeeMe is being developed in close cooperation with the users. The software is free, easy to install and use, and does not require technical personnel, or much technical skill. This gives a possibility for many-to-many communication as an alternative to one-to-one or one-to-many communication. Many kinds of Internet communities have been formed around this application, because of the public or password secured meeting places, i.e. programs called reflectors, where people can meet each other without any preparation. Reflector surfing is a favorite pasttime hobby and could be compared to tv-channel surfing. A big difference, however, is that here people communicate with each other and not with a tv station.

When we bring videoconferencing from studios to desktops, many new possibilities for different fields of life from education to work and leisure time will open. There are already schools using this media. Software like CU-SeeMe or Mbone-based video-conferencing tools help us to devise and test new ways of working, studying and spending leisure time in tomorrow's world. They are developing new communication cultures on Internet. They influence our ways of behaving and communicating, but also show that human-computer interaction cannot substitute human-human communication. Internet video-conferencing systems add many properties of face-to-face communication to „traditional" text-based computer mediated communication, and so the computer-mediated communication is not anymore solely faceless written communication, but it can even be written communication integrated with a real time video. The new communication technology is a field where computer scientists and semioticians as well as linguists, psychologists and sociologists can find common research interests, which is also important now when more and more people are starting to use computers and Internet for communication.

Graphic Evolution – How Signs Develop Over Time

Leif Allmendinger

Graphic signs evolve over time
based on perceptual, cognitive, social, and cultural user needs.
In this evolution, good representations tend to live and bad ones tend to die.

What makes one form of representation better than another?
Better signs are easier to use.
Effectively, they function as cognitive tools – „in the world" knowledge
that allows people to communicate and solve problems
while avoiding complex cognitive tasks.

Both readability and writeability (legibility and sign production)
determine „ease of use",
and one form of representation may answer only some user needs.
It is therefore necessary to represent the same information in different ways to
answer different needs.

Like biological evolution,
information design does not evolve in a neat, predictable way.
In the short run, social, cultural, and political, and technical considerations
can determine whether people accept a new form of representation.

Viewing human-computer interface design as evolution
can shed light on how research may be implemented
and why societies are slow to absorb improvements in the field.
It also raises a major question for semioticians –
how do we speed up this natural process?

Future research will deal with this question.
Right now, I only know some of the reasons it takes so long.

Seven things that slow info design evolution:

1. Principles discovered in one research project may or may not extend to others.
2. Users may need extensive time to learn how to use better forms of representa tion.
3. Establishing standards may limit innovation.
4. We don't know how to tell the difference between quality of design strategy and quality of implementation.
5. Cognitive research considers users as isolated information processors.
6. Evolution requires competing approaches over time.
7. A response to failure approach to design may be slow to recognize new oppor tunities.

A Systemic Semiotic Approach to IS Development and Use

Rodney J. Clarke

A semiotic model of language called Systemic Functional Linguistics (SFL) has been developed by British linguist Michael Halliday and colleagues (Hasan, Martin, Matthiessen, Ventola, Christie and others), and over the last decade it has been extended into a general semiotics called Social Semiotics (Kress, van Leeuwen, Threadgold, Lemke, Thibault and others). Studies which combine both SFL and Social Semiotics have been referred to as Systemic Semiotics (Fawcett). Typical application domains for Systemic Semiotics have included education and cultural studies. Since 1990, Systemic Semiotics has also been applied to workplaces in general, referred to as organisational semiotics, and to the IS discipline, referred to as semio-informatics (Clarke, Cross, O'Brien, and Tebble). The application of Systemic Semiotics to semio-informatics and organisational semiotics holds great potential since traditional information-theoretic approaches lack, amongst other things, a rigorous definition of context needed to describe the conditions under which „information" becomes informative.

Systemic semiotics has been applied to informatics to theorise:

(i) work practices as textual processes;

(ii) IS and IT as involved in the production of textual products (reports, schedules, invoices);

(iii) IS and IT as intertwined with the textual products which utilise these products (management decision making, sales activities, service encounters with clients);

(iv) the textual processes and products of systems development activities including analysis and design; and

(v) the evolution of IS in organisations.

Information Field: a New World Model?

Horst Völz

We know of two ways of getting information or knowledge. The intuitive way takes up Dreyfus' notion of gray knowledge. Information is gained by seeing, hearing, and acting. The second way takes off with the ancient Greeks. They discovered that stories can be created in an axiomatic, or – as we would call it – an algorithmic, manner. The concept of truth appears, and much later that of rationalism.

Up to our times, all descriptions of (parts of) the world are contained in models. Specifically, matter, energy, and information are but three such models of the world. Matter appears to be appropriate in the case of chemical aspects, energy in the case of physical problems. Information is a new model that came up by Norbert Wiener's work. It appears to be useful for many problems of our time.

It is obvious that these three models do not describe the whole world. We need other models as well. Another abstract division of the world separates the discrete from the continuous, yet another one the static from the dynamic aspects. I maintain that a hypothetical information field may be appropriate to deal with continuous and dynamic problems at the same time. However, its definition is problematic. Physicists have, for a long time, successfully worked with the concept of field, but they have not given a definition of it. An old problem was pointed out by Faraday. He showed that fields need time and they have a velocity. Therefore he named it *fare-off-action*.

When looking for problems that could be described in terms of information fields, I first found the typical distance between persons, and later the stream of persons walking to and from locations like railway stations, theatres etc. In some of these cases it is possible to calculate behaviours.

Additional aspects are contained in, e.g.,
H. Völz: Information verstehen – Facetten eines neuen Zugangs zur Welt. Braunschweig, Wiesbaden: Vieweg-Verlag, 1994

H. Völz: Meaning Outline of the Term Information. In: K. Kornwachs and K. Jacoby: Information. New Questions to a Multidisciplinary Concept. Berlin: Akademie Verlag, 1996, 19-39

Conceptual Landscapes of Knowledge: A Pragmatic Paradigm for Knowledge Processing

Rudolf Wille

An understanding of knowledge based on Peirce's pragmatism can be activated by using the metaphor of landscape. This is outlined by discussing conceptual landscapes of knowledge within the developments of Formal Concept Analysis. Various tasks of conceptual knowledge processing are considered such as exploring, searching, recognizing, identifying, analyzing, investigating, deciding, improving, restructuring, and memorizing. All these tasks are illustrated by examples which show the fruitfulness of the landscape paradigm of knowledge. Most of them have been implemented by use of the management system TOSCANA.

Problems and Questions suggested by participants

During the seminar, participants were asked to do two things:

- Identify a problem in your field of research (informatics or semiotics, as may be the case) that interests you.
- Formulate a specific question that pertains to that problem, and that would help solve the problem. Address your question to the other field (semiotics or informatics).

Several persons responded, in writing, to this request. Here are their answers. They may indicate directions for further research.

Leif Allmendinger

Problem and Question:

How can semiotics speed up the evolutionary process underlying the field of human-computer interaction?

Peter Bøgh Andersen

Problem:

The relation between language usage and language system. On the one hand, language system is an emergent property of the millions of utterances. On the other hand, the system constrains the self-same utterances it is made of.

Question:

How can emergent properties become independent entities with causal effects?

Udo Figge

Problem:

It seems that the production and transmission of a sign is not enough for the sign to be received by some other organism. Rather, it must be accompanied by some key that opens the entrances of the other organism. In human language, it seems to be the intonation sentences are packaged in, that serves this function.

Question:

If a computer becomes part of a human-machine dyad like the one it was alluded to in some sessions, and if the computer assumes the role of an initiator of the interaction/cooperation, what can be done in order to equip the signs the computer produces with some force that opens the attention of the human for the signs?

Wolfgang Hesse

Problem and Question:

About the relation of

- concept (defined by the dichotomy extension/intension) and
- sign (defined by the semiotic triad).

Is there a uniform way to view these terms? Is there a (somehow enhanced) triad to cover intensional *and* extensional aspects?

Angelika Hoppé, Ulrike Wilkens

Thesis:

To become a medium in the sense of *sign*, the mediality (mediating function, the functioning as a sign) of a special technique (e.g. the computer) has to be developed (by the users, who want to use this technique as a medium for special purposes !?).

Can we describe the process of developing the medium as *dis-covering* the signs that the technique mediates?

„*Dis-covering*", because we think, a new „sign-technique" (like that of the computer) stands between the user and the sign, unless it is a fully developed medium.

Could we say, that discovering the mediality of a sign-technique (in any case?) has to deal with the three dimensions of the sign?

We would describe these phases (not necessarily strictly separated and sequential) as follows:

1. We focus on the representamen (or the syntactic dimension of the sign): we become acquainted with the „surface" (the outmost level) of the signs, we deal with the handling of this surface, becoming used to the rules of combining the representamen etc.

2. We concentrate on the relation of the representamen to the object under the special conditions of the technique we have to deal with (the semantic dimension).

We explore, what happens to „real things" in our domain, when they turn into objects a (computer-based) representamen refers to. We find out, what objects are appropriate to be represented by this special technique, whether they ought to have a special structure/character etc.

3. The interpretant comes into the center of interest (the pragmatic dimension). The mediality (or should we say semioticity?) of the technique has been developed, when we can use it as a medium in meaningful situations (work, learning): interpretants can be created, sign processes can go on, mediated by a technique, we do not recognize as such any longer (= transparent, ubiquitous ...)

Questions we are interested in:

Does this view (a semiotic one???) help to determine the degree of mediality a technique has already obtained in special domains (e.g. education, art)?

Does it help to find out what to do, to develop the technique to be a medium for our special purposes?

Does this approach allow to make assumptions/predictions about how a technique will or can be used in the future?

Questions (to semioticians, but informaticians may also answer):

- Are the phases of *dis-covering* the medium common to other „sign techniques" (e.g. print, film, hypermedia), for individuals, groups, societies in a similar way?
- Are we right in characterizing a technique as „sign", when it functions as a medium?

René Jorna

Problem and Question:

How to decide between different models in a real problem in information systems design-development. User models, program models, organizational models, task models, cognitive models, expertise models, communication models, etc.

Are there criteria, decision rules based on semiotic principles that can be used immediately or after short adjustments?

Solomon Marcus

Problem:

Elaboration of some adequate concepts of semiotic machines and their confrontation with the existing basic entities in informatics.

Question:

Formulate some basic requirements a concept of semiotic machine should fulfill.

Mihai Nadin

Problem:

Learning.

Question:

How to design effective learning environments?

Frieder Nake

Problem:

There has to be a (new) thorough and powerful theoretical foundation for informatics, if informatics is to become a real science.

Question:

In which precise and detailed sense could the concept of a sign contribute to such a theory, and which particular concept of sign is well suited for this purpose?

Clarisse Sieckenius de Souza

Problem:

I work with human-computer interaction, as a researcher, a teacher, and a designer. And the greatest challenge for me now is to be able to design good interactive systems and to teach my students how to do *better*.

Questions:

How can semiotic theory help me to understand my job more deeply (give me insight), and achieve better results? I.e., does semiotic theory have a word to say to designers?

Peirce seems to be the prevailing reference in all our discussions. Hasn't any alternative view in semiotics been able to provide useful perspectives upon „computer semiotics" from a semiotician's point of view? Why?

Adler & Winograd have written a book called „Usability: Turning Technology into Tools". Would any of our semiotician colleagues think of writing something like: „Computer Semiotics: Turning Philosophy into Technology"? Would he/she be able to give us the very high-level approach he/she would take to do it (or the reasons why he/she wouldn't do it)?

Ronald Stamper

Problem (from a book *Information& the Second World* which I am writing):

Human welfare derives from:

1	physical consumption;	1' defensive expenditure
2	use of physical capital;	2' maintenance
3	physical competence	
4	semiological competence	
5	enjoyment of semiological processes	
6	enjoyment of semiological capital (= community)	

[Items 1 through 3 comprise World-1: physical/economic man, scarce resources;
items 4 through 6 comprise World-2: semiological/social man, info resources (virtually) unlimited;
items 3 and 4 comprise selfrealisation (work, access, liberty, ...)]

GNP = 1 including 1' & 2' regarded as positive

ISW = Index of Sustainasble Welfare = 1 - 1' - 2' and term for 2 & possibly 3

But no index takes account of 3, 4, 5, & 6 which are of great importance to social semiological man.

Classical economics is content to advise destruction of 3, 4, 5 & 6 for the sake of efficient production to satisfy 1.

Question:

How do we construct an *index of semiological welfare*?

Peter Stephan

Problem:

The physical and cognitive coupling of man and environment is identified as a design problem. Our environments are penetrated more and more by computer technology (ubiquitous computing).

Question:

What contribution do semiotics and informatics have to offer for the design of intelligent environments?

Dag Svanæs

Problem:

Currently, interactive computer artifacts are represented mainly as text in a formal language. Implicit in these languages are the concepts of „objective" objects, discrete linear time, and static states. My empirical research shows that these concepts are not intuitive to ordinary users. They consequently have great problems expressing interactive behavior.

They seem to conceptualize behavior through metaphor. Their mental processes are more like operations on visual images (i.e., visual thinking). I think this is what Lakoff and Johnson call image schemata operations. For some reason, our current languages with their logical basis inhibit schemata operations like foreground/background switching and mapping.

Question:

Is it possible to design formalisms (i.e., tools for thought) that encourage „creative" image schemata operations like conscious foreground/background switching, metaphor creation, and methonymisation?

Julian Warner

Problem:

Saussurean and Peircean traditions and influences in semiotics.

Question:

Which is more valuable to informatics: the Saussurean or the Peircean model of the sign?

Dagstuhl-Seminar 9608 **List in Participants**

Leif **Almendinger**
229 West Sycamore Street
Sycamore IL 60178
USA
leif@art.niu.edu
tel: +1-815 899 3085

Peter B. **Andersen**
Aarhus University
Dept. of Information & Media Science
Niels Juelsgade 84
DK-8200 Aarhus
Denmark
pba@imv.aau.dk
tel: +4589421955

Jorge **Bogarin**
Offenburgerstr. 83
D-79108 Freiburg
Germany
tel: +49-761-50 66 39

Sheelagh **Carpendale**
Simon Fraser University
School of Computing Science
Burnaby B.C. V5A 1S6
Canada
carpenda@cs.sfu.ca
tel: +1-604-291 5938

Rodney J. **Clarke**
University of Wollongong
Dept. of Business Systems
South Coast Mail Centre
Locked Bag 8844
Wollongong NSW 2521
Australia
rodney@pcmail-bld40.uow.edu.au
tel: +61-42-213752

Wolfgang **Coy**
Universität Bremen
Fachbereich Informatik
Bibliotheksstraße
28334 Bremen
Germany
coy@informatik.uni-bremen.de
tel: +49-421-218-2731

Ernest **Edmonds**
Loughborough University of Technology
Department of Computer Studies
Loughborough LE 11 3TU
Great Britain
a.e.edmonds@lut.ac.uk
tel: +44-5092-22691

Udo L. **Figge**
Ruhr-Universität Bochum
Romanisches Seminar
44780 Bochum
Germany
Udo.L.Figge@rz.ruhr-uni-bochum.de
tel: +49-234-700-26 30

Andreas **Goppold**
FAW Ulm
Helmholtzstr. 16
89010 Ulm
Germany
goppold@faw.uni-ulm.de
tel: +49-731-501 915

Wolfgang **Hesse**
Philipps-Universität Marburg
Fachbereich Mathematik
Hans-Meerwein-Straße
D-35032 Marburg
Germany
hesse@informatik.uni-marburg.de
tel: +49-6421-28 54 50

Berit **Holmqvist**
Aarhus University
Dept. of Information & Media Science
Niels Juelsgade 84
DK-8200 Aarhus
Denmark
B\-Holmqvist@imv.aau.dk
tel: +45-89 42 19 60

Angelika **Hoppé**
Universität Bremen
FB 3 Informatik
Postfach 330440
D-28334 Bremen
Germany
hoppe@informatik.uni-bremen.de
tel: +49-421-218-31 83

René J. **Jorna**
University of Groningen
Department of Management & Organization
P.O. Box 800
NL-9700 AV Groningen
The Netherlands
r.j.j.m.jorna@bdk.rug.nl
tel: +31-50-63 36 25

Inger **Lytje**
Aalborg University
Dept. of Communication
Langagervej 6
DK-9220 Aalborg
Denmark
inger@hum.auc.dk
tel: +45-98 15 85 22

Solomon **Marcus**
Academia Română
Secjia de Stiinte Matematice
Calea Victoriei 125
RO-71102 Bucharest
Romania
solomon@roimar.imar.ro
tel: +40-1-650-37 77

Mihai **Nadin**
Universität Wuppertal
Fachbereich 5 Computational Design
Hofaue 35-39
42103 Wuppertal
Germany
mihai.nadin@math.uni-wuppertal.de
tel: +49-202-439-30 32 (secr. -3283)

Frieder **Nake**
Universität Bremen
FB Informatik
Postfach 330440
D-28334 Bremen
Germany
nake@informatik.uni-bremen.de
tel: +49-421-218 3525 or 2894

Jeff **Nickerson**
Coopers & Lybrand
5th Fllor
399 Thornall St.
Edison NJ 08837
USA
jvn@interamp.com
tel: +1-908-906-60 95

Kenneth **Nilsson**
Umea University
Department of Computer Science
S-90241 Umea
Sweden
kenneth.nilsson@informatik.umu.se
tel: +46-90-16 56 30

Winfried **Noeth**
Universität-GSH Kassel
FB 8 Anglistik/Romanistik
Georg-Forster-Str. 3
34109 Kassel
Germany
noeth@hrz.uni-kassel.de
tel: +49-561-804-3349 oder 3379

Anita **Nuopponen**
University of Vaasa
Faculty of Humanities
Dept. of Communication Studies
Box 700
SF-65101 Vaasa
Finnland
atn@uwasa.fi
tel: +358-61-32 48-370

Bo **Pedersen**
Aarhus University
Centre for Semiotic Research
Finlandsgade 26
DK-8200 Aarhus
Denmark
bop@daimi.aau.dk
tel: +45- 89 42 44 66

Roland **Posner**
TU Berlin
Arbeitsstelle f. Semiotik
Sekr. TEL 6
Ernst-Reuter-Platz 7
10587 Berlin
Germany
posn0135@mailszrz.zrz.tu-berlin.de
tel: +49-30-314-23 115

Lars **Qvortrup**
Odense Univerity
CCS
Campusvej 55
DK-5230 Odense
Denmark
larsq@litcul.ou.dk
tel: +4566190412

Arne **Raeithel**
Universität Hamburg
FB Psychologie
Isestraße 7
20144 Hamburg
Germany
raeithel@informatik.uni-hamburg.de

Lucia **Santaella Braga**
Sao Paolo Catholic University
Postgraduate Programm in
Communication and Semiotics
Rua Monte Alegre 984
CEP 05014-000 Sao Paolo
Brasil
pucsppcs@eu.ansp.br or
lbraga@exatas.pucsp.br
tel: +55-11-651374

Peter **Schefe**
Universität Hamburg
FB Informatik
Vogt-Kölln-Str. 30
22527 Hamburg
Germany
schefe@informatik.uni-hamburg.de
tel: +49-40-54 715-427

Clarisse **Sieckenius de Souza**
PUC
Departamento de Informatica
Rua Marques de Sao Vicente 225
22453 Rio de Janeiro RJ
Brasil
clarisse@inf.puc-rio.br
tel: +55-21-529-94 62

Dirk **Siefkes**
TU Berlin
Fachbereich 13 Informatik
Franklinstr. 28-29
10587 Berlin
Germany
siefkes@cs.tu-berlin.de
tel: +49-30-314-73 501

Ronald K. **Stamper**
Universieit Twente
Faculteit der Technische Bedrijfskunde
Postbus 217
NL-7500 AE Enschede
The Netherlands
R.K.Stamper@sms.utwente.nl

Peter **Stephan**
Universität Wuppertal
Fachbereich 5 Computational Design
Hofaue 35-39
42103 Wuppertal
Germany
stephan4@code.uni-wuppertal.de
tel: +49-202-439-23 84

Dag **Svanaes**
University of Trondheim
Department of Informatics
N-7055 Dragvoll
Norway
dags@ifi.unit.no
tel: +47-73591840

Georg Christoph **Tholen**
Universität-GSH Kassel
Wiss. Zentrum
Gottschalkstr 26
34109 Kassel
Germany
tholwz2@hrz.uni-kassel.de
tel: +49-561-804-28 06

Horst **Völz**
Freie Universität Berlin
Informationswissenschaft
Malteserstr. 74 - 100
12249 Berlin
Germany
tel: +49-30-7792-855
url: Institut für Kybernetik - Berlin

Julian **Warner**
The Queen's University of Belfast
School of Finance and Information
Information Management Division
Belfast BT7 1NN
Northern Ireland
j.warner@v2.qub.ac.uk
tel: +44-1232-24 51 33 ext. 36 21

Ulrike **Wilkens**
Universität Bremen
FB 3 Informatik
Postfach 330440
D-28334 Bremen
Germany
wilkens@informatik.uni-bremen.de
tel: +49-421-218 2421

Rudolf **Wille**
Technische Hochschule Darmstadt
Fachbereich Mathematik
Schlossgartenstr. 7
64289 Darmstadt
Germany
wille@mathematik.th-darmstadt.de
tel: +49-6151-16-3415

Jun Petersen **Yoneyama**
Aarhus University
Dept. of Information & Media Science
Niels Juelsgade 84
DK-8200 Aarhus
Denmark
jun@imv.aau.dk
tel: +45-89 42 19 51

Zuletzt erschienene und geplante Titel:

Ph. Flajolet, R. Kemp, H. Prodinger, R. Sedgewick (editors):
'Average-Case'-Analysis of Algorithms, Dagstuhl-Seminar-Report; 119; 03.07.-07.07.95 (9527)

D. Gustfield, T. Lengauer, C. Sander (editors):
Molecular Bioinformatics, Dagstuhl-Seminar-Report; 120; 10.07.-14.07.95 (9528)

J. Chomicki, G. Saake, C. Sernadas (editors):
Role of Logics in Information Systems, Dagstuhl-Seminar-Report; 121; 17.07.-21.07.95 (9529)

R.S. Boyer,A. Bundy, D.Kapur, Ch. Walther (editors):
Automation of Proof by Mathematical Induction, Dagstuhl-Seminar-Report; 122; 24.07.-28.07.95 (9530)

P. Cousot, R. Cousot, A. Mycroft (editors):
Abstract Interpretation, Dagstuhl-Seminar-Report; 123; 28.08.-01.09.95 (9535)

P. Brunet, D. Roller, J. Rossignac (editors):
CAD Tools for Products, Dagstuhl-Seminar-Report; 124; 04.09.-08.09.95 (9536)

C. Dwork, E.W. Mayr, F. Meyer a.d. Heide (editors):
Parallel and Distributed Algorithms, Dagstuhl-Seminar-Report; 125; 11.09.-15.09.95 (9537)

C. Hankin, H. R. Nielson (editors):
New Trends In the Integration of Paradigms, Dagstuhl-Seminar-Report; 126; 18.09.-22.09.95 (9538)

U. Herzog, G. Latouche, P. Tran-Gia, V. Ramaswami (editors):
Applied Stochastic Modelling in Telecommunication and Manufacturing Systems, Dagstuhl-Seminar-Report; 127; 25.09.-29.09.95 (9539)

L. Hordijk, G. Korn, A. Sydow (editors):
Modelling and Simulation of Complex Environmental Problems, Dagstuhl-Seminar-Report; 128; 02.10.-06.10.95 (9540)

J. André, A. Brüggemann-Klein, R. Furuta, V. Quint (editors):
Document Processing, Dagstuhl-Seminar-Report; 129; 16.10.-20.10.95 (9542)

J. Collado-Vides, R. Hofestädt, M. Löffler, M. Mavrovouniotis (editors):
Modelling and Simulation of Gene and Cell Regulation, Dagstuhl-Seminar-Report; 130; 23.10.-27.10.95 (9543)

F. Cucker, T. Lickteig, M. Shub (editors):
Real Computation and Complexity, Dagstuhl-Seminar-Report; 131; 06.11.-10.11.95 (9545)

K. Echtle, W. Görke, J.-C. Laprie, W. Schneeweiss (editors):
Quantitative Aspects of Designing and Validating Dependable Computing Systems -- Calculations, Measurements, and Simulations, Dagstuhl-Seminar-Report; 132; 13.11.-17.11.95 (9546)

J. Buchmann, R. Loos, R. Mäder (editors):
Computeralgebra - Software, Dagstuhl-Seminar-Report; 133; 05.02.-09.02.96 (9606)

O. Danvy, R.Glück, P. Thiemann (editors):
Partial Evaluation, Dagstuhl-Seminar-Report; 134; 12.02-16.02.96 (9607)

P.B. Andersen, M. Nadin, F. Nake (editors):
Informatics and Semiotics, Dagstuhl-Seminar-Report; 135; 19.02.-23.02.96 (9608)

S. Näher, H. Noltemeier, I. Munro (editors):
Data Structures , Dagstuhl-Seminar-Report; 136; 26.02.-01.03.96 (9609)

A. Bonner, A. Heuer, L. Tanca (editors):
New Trends in Database Languages, Dagstuhl-Seminar-Report; 137; 04.03.-08.03.96 (9610)

D. Dolev, R. Strong, R. Reischuk (editors):
Time Services, Dagstuhl-Seminar-Report; 138; 11.03.-15.03.96 (9611)

R. Bajcsy, R. Klette, W. Kropatsch, F. Solina (editors):
Theoretical Foundations of Computer Vision, Dagstuhl-Seminar-Report; 139; 18.-22.03.96 (9612)