Michael Huth
Department of Mathematics
Technische Hochschule Darmstadt, Germany

Symbolic and Sub-symbolic Knowledge Organization in the Computational Theory of Mind

Dr. Michael Huth was born in Germany and studied mathematics and computer science at Darmstadt University. He received his PhD degree from the Graduate School of Tulane University of Louisiana. He was a post-doctoral fellow at Kansas State University, Manhattan, KS, and at the Imperial College of Science, Technology and Medicine in London. Currently, he is a scientific collaborator at the Semantics Group at Darmstadt University.

Huth, M.: **Symbolic and Sub-symbolic Knowledge Organization in the Computational Theory of Mind.**
Knowl.Org. 22(1995)No.1, p. 10-17, 14 refs.
We sketch the historic transformation of culturally grown techniques of symbol manipulation, such as basic arithmetic in the decimal number system, to the full-fledged version of the Computational Theory of Mind. Symbol manipulation systems had been considered by Leibniz as a methodology of inferring knowledge in a secure and purely mechanical fashion. Such 'inference calculi' were considered as mere artefacts which could not possibly encompass all human knowledge acquisition. In Alan Turing's work one notices a crucial shift of perspective. The abstract mathematical states of a Turing machine (a kind of „calculus universalis" that Leibniz was looking for) are claimed to correspond to equivalent psychological states. Artefacts are turned into faithful models of human cognition. A further step toward the Computational Theory of Mind was the physical symbol system hypothesis, contending to have found a necessary and sufficient criterion for the presence of 'intelligence' in operative mediums. This, together with Chomsky's foundational work on linguistics, led naturally to the Computational Theory of Mind as set out by Jerry Fodor and Zenon Pylyshyn. We discuss problematic aspects of this theory. Then we deal with another paradigm of the Computational Theory of Mind based on network automata. This sub-symbolic paradigm seems to avoid problems occuring in symbolic computations, like the proble *frame problem* and *graceful degradation.* (Author)

1. Dr. Frankenstein: An Archetypal Myth

Creating one of our own kind has been one of the great myths of man. The motivations and desires driving this myth are manifold and complex. One of the intriguing aspects of such a putative feat is that it would put us on equal terms with gods. We would have convincingly demonstrated that we are able to figure out all that is needed to construct, from first principles, a 'thing' that could pass as a human being—at least within some clearly defined boundaries. The contemplation of such a possibility causes immediate emotional and philosophical reactions. If we were ever confronted with such a successful 'product' of human engineering, it would severely affect our beliefs and attitudes. The most compelling questions which come to *my* mind are:

– Is it possible, or even inevitable, that such a 'creature' possesses a soul?
– What is consciousness and is it a necessary feature of any such 'creature'?
– Do I have to or do I want to meet this 'creature' with the same respect and should it be treated under the same code of human rights that we take as our principal set of values in human interaction?

This is just a small sample of possible reactions and, luckily, these questions are up to this day purely speculative.

These questions are weighty, nonetheless, since they target the very center of our philosophical or religious beliefs. For example, what if such a creature would not need to have a soul in order to pass some test of 'being human'; would this mean that *we* don't have to have a soul as well?

I recall an interview I watched on television in which a researcher in Artificial Intelligence said that his community is now in a position to ''play God'', that they could design and construct artefacts which meet any cognitive skills that they chose to specify. I felt that 'playing God' was a rather arrogant statement; after all, we have been thrown into this universe with all its physical constraints and all we can possibly do is to make use of the stuff we find and of the laws which govern its interaction. We cannot even change these laws! What poor gods we are ... [1]

The statement of a scientist above shows that the myth of creating our own kind has been transformed from its mystical state (which performed, like any other myth, an important psychological function) into an engineering programme for which it seems to be only a matter of time and money to fulfil this archetypal myth to any degree of accuracy.

2. Symbol Manipulation: A Historic Perspective

Transforming this particular myth into a scientific programme was a continuous process which may not be separated from our cultural development over the past few centuries. One thing we notice right away is that the scientific programme of Artificial Intelligence is not interested in 'producing' something which would physically be indistinguishable from any other human being. It has no interest in producing beautiful skin or dazzling hair for its devices; (which would have its own questionable commercial relevance) rather, it focuses on the construction of units which are able to process information from their surrounding environment (a sensory environment or

some other kind of organized information), to make inferences based upon the perceived information and to act according to those inferences.

The consideration of such projects made us re-think how cognition works within our own heads and bodies and helped shaping the *Computational Theory of Mind*, which can be condensed into the thesis:

> Cognition is nothing but the computational manipulation of mental representations.

We want to sketch the origins and the gradual development of this thesis, thereby following closely Sybille Krämer's account in (5). Philosophical roots of such thinking can be found in the writings of Th. Hobbes and G.W. Leibniz. For the latter, all correct inferential thinking was a process of the formal creation and manipulation of symbolic expressions. If we compare such a suggestion of definition with the activities of LISP programmers[2], the gap from Leibniz to the sixties appears to be narrow indeed; yet, it is hard for us to imagine what tremendous difficulties people had to overcome in formulating what 'symbol' and 'manipulation' mean and in realizing their full expressive power. For example, it was a big step ahead to use letters as representations of arbitrary number values while doing complex arithmetic manipulations of expressions (Viète 1540-1603). It was an even bigger step of abstraction when René Descartes used this technique for translating problems from analytic geometry into a purely algebraic form, thereby solving geometric problems by algebraic (= symbolic) means. (This possibility was announced in (3). The obtained solutions could then be re-interpreted in the geometric world.

One of the first broadly accepted formal systems of symbol manipulation was the establishment of basic arithmetic in the decimal number system, occuring in Europe in the 14-15th century. This technique required a fairly high education in reading and writing; yet, these skills sufficed: one could learn to compute correctly without having to know why the scheme worked at all, as long as one remembered all the rules involved! This system permeated the entire culture of human activities (business, academics, planning, etc.) and its success stems largely from the *compositionality* of its underlying semantics.

2.1 Compositionality of Meaning

In order to explain the notion of compositionality we consider a simpler representation of numerals as finite strings of zeros and ones. For example, '100' represents the number four, '110' the number six (since 0 x 1 + 1 x 2 + 1 x 4 = 6), and '111' the number seven. We can specify all such strings by defining a *transformation grammar*

<bin_dig> ::= 0 | 1
<bin_num> ::= <bin_dig> | <bin_num><bin_dig>

introducing two syntactic sorts, that of a binary digit <bin_dig> and that of a binary numeral <bin_num>. The first line consists of two rules saying that '0' and '1' are binary digits. The second line expresses that every binary digit is also a binary numeral and that the syntactic concatenation of a binary numeral with a binary digit results in another binary numeral.

The point is that we are given four rules for constructing binary numerals. Dually, these rules can be viewed as *inference rules* which we can use to prove that a given string of symbols is indeed a representation of some binary numeral. Moreover, and most importantly, the structure of these rules determines the meaning of complex representations as a function of the meanings of its finitely many subexpressions. For example, the meaning (=the natural number represented by the expression in question) of '0' is the number zero, the meaning of '1' is the number one (these being the atomic cases), but the meaning of '101' is two times the meaning of '10' plus the meaning of '1' and so on.

It is this compositionality (according to the logician Frege a vital ingredient of any concept of „meaning") which is the driving mechanism of the success of *Denotational Semantics*, a mathematical approach to Programming Language Semantics developed by Dana Scott and Joseph E. Stoy at the Programming Language Group at Oxford University (12). For the same reason, but a few centuries earlier, the decimal system for arithmetic became established as an accepted cultural technique that is still practiced today. We have to stress, however, that this symbol system and the rules of its manipulation constitute a genuine *artefact*, a product of an evolving culture; further, that very product might only have been developed because increased trading and communication needs made the invention of easy and fast computation schemes necessary. Roman numerals had been used before, but their meaning is far from being compositional (try it!); they are merely a means of representing numbers but the intrinsic architecture of these representations does not suggest how arithmetic computations have to be carried out; for that, you need an abacus!

2.2 The Search for an „écriture universelle"

To Leibniz inferential thinking was nothing but the concatenation and substitution of symbols. However, Leibniz did not claim that *all* human cognitive activities are done in such a deductive fashion. For example, Descartes' fundamental insight (3)

> „Je pense, donc je suis."

is a direct and sudden awareness of the truth of this assertion and cannot be 'proved' from assumptions we arrived at using previously concluded knowledge. Leibniz only wanted to model a certain kind of thought, namely propositional assertions, and he looked out for a secure methodology that allowed him to reach new knowledge based on previously established facts. In that sense, a symbol manipulation system is an *artefact*, an external tool facilitating our perception of the world we live in.

Leibniz' epistemological view is even stronger: we can perceive the world only through symbolic representations (6a); it is God's privilege to be capable of a „cogitatio asymbolica" (6b). A similar thought is expressed in Wittgenstein's writing (14):

> „Wenn wir über den Ort sprechen, wo das Denken stattfindet, haben wir ein Recht zu sagen, daß dieser Ort das Papier ist, auf dem wir schreiben, oder der Mund, der spricht."[3]

To summarize, we can say that symbol manipulation started out as an artefact, indeed as an art, a creation of artificial languages which were more conducive to supporting cognitive goals than our natural languages. Such formal systems allow us to represent and manipulate cognitive states and goals in a very efficient way. These systems are inventions and consequently cultural achievements and not part of our biological 'equipment'. Having said that, this does not rule out the presence of similar systems within our bodies.

The inflationary development of such artefact has changed the way scientific communities communicate. We tend to state facts and rules in symbolic and condensed terms, trying to eliminate as much natural language as possible, thereby trying to rule out any source of misunderstanding our ambiguity. If we compare Riemann's prose with Cauchy's formal account of calculus (using the \epsilon-\delta formalism), then each of them talks about calculus but Cauchy already tries to live up to Leibniz' vision. It has to be said that even Leibniz thought about a„calculus universalis" which was not based on symbols but on images and natural language, although he soon abandoned such plans in favor of some operational, symbolically oriented code of inferential thought (which he never succeeded in specifying).

Leibniz' vision pre-dated the birth of symbolic logic (1) since it reduced truth (of an assertion) to validity within a given calculus (i.e., the assertion has a 'proof' that refers only to the rules of the given calculus). With such a „calculus racionator" one could decide the truth of assertions by a blind and mechanical manipulation of symbols within a calculus of proofs. This „cogitatio caeca vel symbolica" would then be the only cognitive principle which could lead us blind cognizers through the seemingly dark cosmos (6).

There are meta-mathematical results which second the contention that not all insight may be gained by the exclusive usage of inferences. If we deal with a formal system of some minimal expressiveness then the logician K. Gödel has demonstrated that there have to be statements which are not deducible within the system unless it is inconsistent (= it cannot be attributed a meaningful semantics). Nonetheless, such statements are 'true' under a suitable interpretation.

2.3 From an Artefact to a Model of Cognition

To proceed from the formalization of a cognitive process in some calculus to its mechanization on a suitable carrier is not a big conceptual step. More surprising is that such an implementation could be done on any material, respectively architecture, as long as it provided sufficient means for faithfully reflecting the properties of the calculus. Now, as long as we construe such calculi as artefacts and epistemological tools we won't run into deep philosophical water; but as soon as we think of such symbolic representations as reflecting mental states (a view endorsed, for example, by J. Locke) we enter the realm of psychology and one could then imagine an 'implementation' of human cognitive processes on a machine and one could study these dynamical phenomena in any wanted detail on an inanimate device. Psychology would turn into purely empirical inquiry.

Such a crucial shift of perspective could be observed in Alan Turing's writing (13). First of all, he formulated an abstract mathematical concept of a Turing machine. Such a machine consists of an infinite tape of cells which are filled with atomic symbols; it also has a device that can read the content of the cell it is positioned at. Moreover, such a machine has a *finite* set of rules of the form:

> If I (=the machine) am in state number five and if the content of the cell I am currently looking at is the letter 'a' then I replace this letter by the symbol '5' and move my reading device one cell to the left with the resulting state being number two.

Thus, such a machine manipulates symbols on an infinite tape according to the rules coded up in the body of the machine, beginning in a distinguished initial state. Such a definition expresses the essence of what people can do externally using finite amounts of pen, pencil and paper[4]. It also gives the notion of computability a mathematically precise and canonical form; the latter notion lacked such formal foundations and a whole community of researchers worked intensely on such a basis during the twenties. We now know of a multitude of different definitions of computability, e.g., Lambda-calculi and μ-recursive functions. All systems suggested up to this day could be shown to be equivalent in the sense that a 'computation' in one system A could be simulated accurately in another system B and vice versa. This does not settle the issue of *what* computability should mean, it merely suggests that we have encountered an extremely robust proposal.

Turing also proved the existence of *universal* Turing machines, the theoretical and conceptual proto-types of the von Neumann computers we all use today. We can represent an entire Turing machine as a sequence of symbols, as long as we agree on a fixed syntactic structure of such a code allowing us to recover all transition rules of the respective machine. So what prevents us from writing down such a sequence on an infinite tape? A universal Turing machine can read such a sequence and simulate the

rules encoded in it to manipulate some other input on its tape; in fact, it can do this with *any* sequence representing a Turing machine. This universality is essentially the working principle of a regular computer. Programs are stored like any other data and can be read and used to transform a specified set of data.

The proof of this theoretical result led to the construction of computers that were realizations of Leibniz' „calculus universalis". We are in no need to point out the dramatic impact this invention had on our daily lives and our societal structures. Yet, Turing turned the mathematical analysis of machines into something with genuine psychological significance. He claimed that if a Turing machine carries out calculations, say, to multiply two natural numbers, then the sequence of abstract mathematical states observed while the machine performs this task should reflect a corresponding sequence of mental states of some human being solving the same problem. In short, he claims that the way we utilize space and recourses for problem solving outside of our bodies is re-interpreting the structure of our internal cognitive processes. Hence, it should be possible to reconstruct the behavior of a 'human calculator' mechanically.

2.4 Artificial Intelligence

If we identify computability with 'being mechanizable' we may ask whether there is a concept that relates to being mechanizable the way that intelligence relates to successful computations. More to the point, if there exists such a correspondence, what kind of concept are we looking at?

In 1976, Newell and Simon provided a simple 'solution by definition'. To them a physical symbol system is basically any device that can store, read and manipulate (representations of) symbols. It should also satisfy additional requirements of „completeness and closure" (7). After careful examination of this definition one notices that LISP, Turing machines and various non-deterministic rewriting systems all qualify as physical symbol systems. In a way this definition attempts to capture the commonalities of various models of computability with respect to their capacity of modeling psychological processes. Newell and Simon made an astonishing, and, maybe, radical suggestion (7):

> „**The Physical Symbol System Hypothesis.** A physical symbol system has the necessary and sufficient means for general intelligent action."

Such a hypothesis does not *explain* what intelligence is, rather, is postulates a precise criterion for its presence within an operating system. It has more the flavor of a mathematical *invariant* and *classifier*. It is invariant in the sense that the specific nature of the implementation of such a physical symbol system is irrelevant with respect to its cognitive performance. Therefore, notions like *thinking, cognition* and *intelligence* can be characterized without reference to some biological species, like us humans. Once more we have been deprived of the feeling that we depict something very special in this universe. Just as Copernicus made evident that the earth is not the center of the universe, Newell and Simon tell us that we are ill-advised if we view us as the 'center of cognition'. Any old machine can cognize as long as it satisfies the physical symbol system hypothesis.

These consequences resulted solely from the claim of sufficiency. To me, the stronger and even more questionable claim seems to be that of necessity. For one thing, it would entail that all human cognition is based on the implementation of some physical symbol system within our heads and bodies. For another, since our entire cognition is based on such a system we could (using the principle of invariance) re-construct such a system on some mechanical device and Psychology and Cognitive Science would be reduced to a descriptive discipline within a purely empirical science.

The Physical Symbol System Hypothesis was the slogan of a new scientific manifesto. It guaranteed that, given an arbitrarily complex bundle of cognitive tasks, one could engineer a system that is able to solve all these problems; more precisely, we could be confident that any such system relied on the same principles of symbol manipulation. If systems could only perform correctly for a *limite* scope of cognitive activities this was just because the evolution of man-made symbol systems was in its infancy; but given enough time and 'man-years' we would have to succeed inevitably without ever being in need of changing our scientific paradigm.

With an emotional distance of almost 30 years and with the grace of having been born in the sixties, it is difficult to share this initial enthusiasm. There are some impressive symbol systems around, but they all are utter specialists and, once removed from their narrow specification space, fail to function. It is interesting to note that other 'evolutionary' trends like the low cost of computing memory and the speed of processors have taken off at a comparatively incredible rate. The rate of progress in crafting a *universally intelligent* physical symbol system is steady but slow at best (2).

2.5 The Computational Theory of Mind

Philosophers and psychologists alike took an active interest in the foundational assumptions of people working in Artificial Intelligence. This interest and work resulting from it provided the foundations of a new scientific, interdisciplinary programme: *Cognitive Sciences*. Two prime sources reflecting the definition of this area are (4,8).

Jerry Fodor contends that we have a *language of thought* implemented in our heads (presumably realized by evolution); some hard-wired operative medium based on inference rules (which lack self-awareness!). If we look at Chomsky's ground breaking work in linguistics we are inclined to believe that the capacity and the fashion of constructing meaningful and correct phrases is largely

independent of the native language of a respective speaker. Thus there must be some language tool inside of us capable of adapting to the very language spoken in a newborn person's environment. A language of thought would then signify a similar vessel that could be filled and refilled with cognitive patterns and situations. So we do have freedom in what and why we learn, but at the same time we are constrained by the architecture of this language of thought, an unpleasant but nonetheless justifiable perspective.

The assumption of such a language of thought gives us explanatory leverage with which we can tackle a number of difficult issues in psychology and philosophy. For example, intensional states and propositional attitudes (schemes like „x believes that P") are notions used in folk psychology, but they could be coded up in complex symbols denoting such states or attitudes. The intrinsic inference rules would then transform and manipulate sequences of such symbols and these transitions and their 'results' could be used as scripts or recipes for action. Hence the manipulation of intensional states and propositional attitudes would be responsible for our external actions in a causal (but probably non-deterministic) way; folk psychology would turn into a serious science.

Here we should pause for a moment. What do we mean by 'symbol' in the case of the language of thought? The previous sections made sufficiently clear that the emergence of the notion of symbol as an external representation was a strenuous and long process. This notion depends on how we perceive and organize our external world. Clearly, we know a symbol when we see one! However, think of the infinitely many ways you can print the letter five on a sheet of paper. Any attempt of reductionism (trying to explain the ink blot representing five as a constellation of molecules) will be futile; such a molecular protocol will not allow us to conclude that this is a description of an ink blot signifying the number five. Only if we put this blot into a context in which it can interact with other blots (like the blots saying „plus two") do we have a chance of saying something about the meaning of this first blot, provided that we introduce a higher, semantic, level of description.

When we consider a language of thought the physiological state encoding that „x believes P" must be so complex that we can only deduce indirectly the existence of such states and that they interact in some causal fashion. Further, different meanings will require physiologically different states, but different physiological states may very well 'denote' the same meaning. This is an all to familiar relationship between syntax and semantics. So how is it possible that logical-semantic relationships are in unison with syntactic-causal relationships to such a great extent that we can construe physical processes as being unquestionably semantically driven? There is a pretty convincing example: a computer! But a computer cannot be 'fully aware' of the semantic significance of what it is manipulating. By the same token, this comment applies to the language of thought (4, p.231):

> „If mental processes are formal, then they have access only to the formal properties of such representations of the environment as the senses provide. Hence, they have no access to the semantic properties of such representations, including the property of being true, of having referents, or, indeed, the property of being representations of the environment." (The emphasis is quoted as well.)

Any cognitive science that builds upon such foundations won't be able to make statements about the meaning of mental representations, it can only record the operative manipulation of these representations.

3. Two Paradigms of Knowledge Organization

3.1 The Explanatory Level of Symhols

We have sketched how culturally fostered techniques led to the progressive formulation of a Computational Theory of Mind in which cognition was reduced to the symbolic manipulation of mental representations. While such a thesis has explanatory force in a variety of issues in psychology and philosophy, it does have its shortcomings.

One problem is that physical symbol systems are just too good at what they are doing! For example, let us consider the transformation grammar describing the syntax of binary numerals represented as binary strings in section 2.1. The trouble with this recursive specification is that it will work perfectly well up to any finite depth of the recursion scheme. This is exactly what is so great about computers; they can apply recursive schemes infallibly and they could not care less about how often this scheme will call itself or other recursive schemes, unless they suddenly run out of memory. Yet, we humans behave differently. We might write down recursive schemes which correctly reflect the task we want to achieve, and we might even succeed in proving that this recursion meets its specification, but we will perform poorly if we have to apply such schemes with a recursive layer of depth greater than six or seven. This can be observed in the way we parse speech, like in:

> The man who crossed the road which was filled with people who were all dressed in red which is a color I don't like was dressed in blue.

Fortunately, we don't speak like that, for we find this difficult to parse. Imagine if the recursive layer of this phrase were twice as long! How can a physical symbol system account for this empirical decline of skillful performance?

Another but closely related issue is that of *graceful degradation.* When people try to solve problems we notice that they will either succeed or that they will fail in a graceful way[5]. For example, we have little difficulty in recognizing a living-room or a bathroom as having the respective function. Now, what happens if we enter a room we suppose to be a bathroom but which is actually

some sort of '(living/bath)-room'? We don't know, and we don't have a very good idea about what to expect. Presumably, such a room will contain a sink and a toilet, but there might also be a sofa and a coffee table. No matter how strange the combination of pieces of furniture might be, we would still be able to make *some* sense out of it.

Now, imagine that concepts like 'living-room' and 'bathroom' are encoded in a symbolic way. There are various forms and names for doings this in Artificial Intelligence *(frames, scripts).* One problem with such an encoding is that it is predicative in the sense that certain criteria *have to be* present if a room wants to qualify as a living-room. Which ones should we choose and which ones may safely be omitted? An even harder problem is the processing of negative information: the specification of attributes that will ensure that we are certainly not dealing with a living-room. Then there is the problem of *variables* or *defaults:* no two living-room are alike and scripts need to reflect this variability by being extremely adaptive while still being *finite* descriptions.

In addition, one has to be able to join scripts describing standard situations. Such a combination cannot just mean the addition of information. Consider scripts for the following two situations: the first script contains all the information I need to attend a typical lecture on biochemistry on a conventional campus; the second script encodes characteristic aspects of life in a ski resort town. How do we combine these, knowing that there will be a winter school in biochemistry in Davos, Switzerland, organized by the European Association of Biochemists? Superimposing these two scripts will severely change the meaning of constituents of each script. More scripts mean more information or the re-evaluation of previously assumed information. There is an approach in Denotational Semantics based on *information systems* (11) which could serve as a mathematical foundation for defaults and related problems (9).

All objections sketched above constitute serious challenges of the symbolic approach.

3.2 A Sub-symbolic Explanatory Level

Knowledge Organization in the symbolic approach is almost of a *discret* nature. Information is encoded in some ordered structure of cells (a datatype) and the very structure of this datatype already constrains and configures the meaning of information tokens placed into these cells. It is the programmer or the designer who determines this arrangement of information tokens. We already discussed how this causes problems, for situations and the knowledge necessary for handling such situations adequately are intrinsically *'soft'* objects, more rubber-like; in our rôle as acting agents, we need to stretch, shrink, or modify information tokens in a way that reflects the current situation we are in.

Indeed, there exists another paradigm of knowledge representation which received its foundations not from the development of digital computers but from the work on dynamical systems and neurophysiology. A comprehensive account of this approach can be found in (10).

Parallel distributed systems (PDS) are networks of nodes which communicate with each other in a fixed architecture. The communication is either synchronous or asynchronous. If the input received by a cell C exceeds its threshold value it will 'fire' and excite, respectively inhibit, those cells having input wires originating in C. A subset of nodes will be designated as *input nodes,* i.e., those nodes that will be fed with the 'encoding' of the information the system should process. Dually, one associates with such a net a subset of *output nodes* that will represent the encoding of the processed information.

In a synchronous network we initialize the input nodes and let the system run under a discrete clock such that we update all activation values *simultaneously.* If the system reaches a stable configuration (the next iteration will not significantly change the patterns of activity in the net), we can interpret this stable configuration as the result of the computation. Repeated initial input should then result in a *sequence* of stable configurations which, in turn, will be input pattern for other nets, and so on ...

It is apparent that this formal setup borrowed its terminology from systems theory and neurophysiology. But we should not be misled by these origins. Parallel distributed processing (PDP) is meant to provide reasoning for psychological evidence supported by a computational language derived from net theory (10, vol. 1, p. 11):

> „Though the appeal of PDP models is definitely enhanced by their physiological plausibility and neural inspiration, these are not the primary bases for their appeal to us. We are, after all, cognitive scientists, and PDP models appeal to us for psychological and computational reasons. They hold out the hope of offering computationally sufficient and psychologically accurate mechanistic accounts of the phenomena of human cognition which have eluded successful explication in conventional computational formalisms; and they have radically altered the way we think about the time-course of processing, the nature of representation, and the mechanisms of learning."

So what *are* representations in this framework? In pursuing this we first need to understand how nets learn. We can provide a processor which feeds a stable configuration of a net back into this net, thus blending it with subsequent input. There exist various algorithms for dynamically adjusting the thresholds of nodes depending on the previous 'performance' of the net („learning algorithms"). The net goes through a sequence of changes, an evolutionary progression which stops if the desired performance has been attained within a given limit of accuracy.

The important point is that information tokens are not really *atomic tokens* anymore; rather, they are abstract distributions of coherence and causality within a given net. Only *after* we have given this net a semantics (i.e., after we have agreed on how to interpret input and output) can we localize such patterns of distribution. Such pat-

terns in their entirety correspond to what we have previously known as a *symbol*. Therefore, this approach can justifiably be called a *sub-symbolic* one.

Let us briefly look at an example (10, vol. 2, pp.22—38): This is a network that has been 'trained' to distinguish between various types of rooms (like *living-room, office, bathroom, bedroom, kitchen*). The concept of (the presence of) a refrigerator is distributed in a group of units, and similarly this is done for concepts like 'oven', 'computer' and, all in all, about forty more such indicators. The architecture of the inhibitory and excitatory connections between these groups reflects how these concepts interact with each other.

The complex or *higher-order* concept of a bathroom can now be seen as that region in the (mathematical and abstract) phase space of this net which pushes configurations towards *that* stable configuration the net will get to when it agrees on having encountered a bathroom (sufficiently many positive attributes inhibit all other options). In fact, if we assign to each state of the net a real number between zero and one, expressing how 'well' this state approximates some higher-order concept, we can represent the set of states as a rather smooth surface with a few hills whose crests represent room types. In this model there is little problem in encountering and handling a (living/bath)-room. It is simply some state in the 'state valley' between the living-room hill and the bathroom hill.

This nicely gets rid of the problem of how to represent a possibly infinitely varied standard situation within a finite script. Standard situations or concepts are recorded in appropriate nets which accumulate an entire history of exposure to all kinds of previously met concrete situations. If the current situation is slightly different, the net will not fail to respond but its reaction will be gradually different from previous ones. At the same time this net should use its most recent exposure for updating its activation architecture accordingly.

It is easy to imagine that a minor adjustment of thresholds or connection strengths will not modify a nets overall quality, although its quantitative behaviour will change slightly. This could lead to better explanations of the type of graceful degradation we often observe when studying the cognitive performance of people placed under physical or mental constraints.

Parallel distributed systems also explain empirical evidence of aspects of sensory perception. For example, the interpretation of missing information in speech perception is determined by its context of previously heard *and* subsequent utterances. The *Trace Model* is a neural net with such a context-sensitive behavior (10, vol. 2, chap. 15). In a certain sense a lot of our sensory perception seems to be *semantically driven*.

I hope that this succinct tour of the theory of neural nets elucidated the main differences between the symbolic and sub-symbolic view of cognition. Still, the sub-symbolic paradigm can safely be classified as an offspring of the Computational Theory of Mind; in this setting cognition is nothing but a *sub*-symbolic manipulation of mental representations. This view only departs from the symbolic interpretation of what mental representations are which further entails a novel concept of computation.

We do not have enough space to discuss weaknesses of this approach as well. Though, it should be said that both accounts could be valid to a certain extent. It is conceivable that they describe the same cognitive phenomena at different levels or 'grain sizes' of computation and meaning.

4. Outlook

Cognitive Sciences are a rapidly developing field of highly interdisciplinary academic and industrial activities. They target one of the centers of human inquiry, attempting to answer who we are by learning how we operate. They face puzzles that might not be fathomable, like: What is consciousness? Is is the entirety of all stable configurations our brains could reach, and if so, would such a mathematical answer *really* satisfy our curiosity? What can we say about emotions, feelings and love? Are they independent of our rational and cognitive skills? Conversely, a feeling like jealousy obviously interferes with cognitive processes; are feelings and emotions even *necessary* for skilled cognitive performances?

Should we be afraid of finding out more about our cognitive freedom *and* our constraints? There are as many answers to this as people; if we had a mathematical theory explaining how things work, say, when we fall in love, I am positive that it would not prevent us from falling in love in Spring in joy and wonder.

Acknowledgements
I would like to thank Dr. Radu Bogdan for introducing me to Philosophy of Mind. My special thanks go to Dr. Dahlberg and her warm reception of my talk given at the Ernst Schröder Seminar during a session on Artificial Intelligence. She encouraged me in writing up this material.

Notes:

1 Although, we are in a rare and privileged position to make moral judgements about our utilization of the universe, a rather divine, albeit, not often used capacity.

2 LISP had been invented by McCarthy in 1960; it is a list-processing language in which programs themselves are represented as lists as well.

3 This roughly translates to: „If we speak about the location where thinking takes place we have a right to say that this location is the sheet of paper we are writing on or our mouth which speaks."

4 The assumption that the tape is infinite makes the theoretical treatment more elegant. In principle we cannot specify an upper bound of the number of sheets a person will have to use to solve a problem we do not yet know, so if we want to have sufficient space for solving *all* problems we better assume an infinite pile of sheets of paper.

5 Sometimes my students try to shatter this empirical evidence during exam periods.

References
(1) Boole G.: The Laws of Thought. Macmillan 1854. Reprinted by Dover Publications in 1958.
(2) Cohen P. R. & Feigenbaum E. A.: The Handbook of Artificial Intelligence. 4 Volumes. Addison-Wesley Publishing Company, Inc. 1982. Second Printing 1989.
(3) Descartes R.: Discours de la méthode. Galerie de la Sorbonne.
(4) Fodor Jerry A.: Representations. Philosophical Essays on the Foundations of Cognitive Sciences. The MIT Press. Cambridge, Massachusetts. 1981.
(5) Krämer S.: Denken als Rechenprozedur: Zur Genese eines kognitionswissenschaftlichen Paradigmas. Kognitivwissenschaft (1991) 2: pp.1-10.
(6) Leibniz G. W. (GPIV 423): Die Philosophischen Schriften. C. I. Gerhardt (ed.). VII B"ande. 1965. Hildesheim: Olms.
(6a) Leibniz, G.W. (GP VII, 3111, 191, 204ff): Die Philosophischen Schriften. C.I. Gerhardt (Ed.). VII Bände. 1965. Hildesheim: Olms.
(6b) Leibniz, G.W. (A II 1, 239): Sämtliche Schriften und Briefe. Edited by the Deutsche Akademie der Wissenschaften zu Berlin. Darmstadt 1923f., Leipzig 1938f., Berlin 1950f.
(7) Newell A. & Simon H. A.: Computer Science as Empirical Inquiry: Symbols and Search. The tenth Turing award lecture. Published in the Communications of the Association for Computing Machinery, 19 (March 1976), pp.113—126.
(8) Pylyshyn Zenon W.: Computation and Cognition. Toward a Foundation for Cognitive Sciences. The MIT Press. Cambridge, Massachusetts. 1986.
(9) Rounds W. & Zhang G. Q.: Defaults in Databases: A domain-theoretic perspective. CWI Amsterdam. Personal Communication.
(10) Rumelhart D. E. & McClelland J. L.: Parallel Distributed Processing. Explorations in the Microstructure of Cognition. Volume 1 & 2. The MIT Press. Cambridge, Massachusetts. 1986.
(11) Scott D. S.: Domains for Denotational Semantics. In: International Colloquium on Automata, Languages and Programs. M. Nielson and E. M. Sclunidt (Eds.). Springer Verlag, 1982. Lecture Notes in Computer Science, vol. 140, pp. 577—613.
(12) Stoy J. E.: Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory. The MIT Press. 1977. First MIT paperback edition, 1981.
(13) Turing A.: Intelligent Machinery. In: Meltzer B. (ed.), Machine Intelligence 5, Edinburgh, 1969.
(14) Wittgenstein L.: Das Blaue Buch. Eine Philosophische Betrachtung. Suhrkamp Verlag, Frankfurt, 1984. p.23.

Address: Dr. Michael Huth, Techn.Hochschule Darmstadt, Fachbereich Mathematik, Schloßgartenstr. 7, D-64289 Darmstadt. e-mail: huth@mathematik.th-darmstadt.de

---

Cont'd Reports & Communications from p. 36

Electronic Dewey for Windows

Dewey for Windows (DFW) is an advanced prototype designed and programmed by OCLC's Office of Research. It is a successor for the Electronic Dewey product released by OCLC Forest Press in 1993 (see announcement in Knowl.Org. 93-1, p.56).

Like the Electronic Dewey, DFW consists of a database containing the DDC, ed.20, and all updates through March 1994. The user interface, however, is a completely new design based on three principles: 1) function-specific windows, 2) fixed display views, and 3) drag-and-drop interaction.

ad 1) Each basic program function is associated with a window specifically designed for that function, e.g. the DDC hierarchy centered on a specific DDC number.

ad 2) To alleviate problems associated with multiple windows, DFW provides fixed Display views. Each of these supports a particular operation or approach to using Electronic Dewey, e.g. the DDC Pages Window filling the left half of the screen and the right half split between a Search Results Window and a DDC Record Window. By this the user will be enabled to search the DDC for specific keywords, view the text of the entries retrieved and display the DDC pages for those numbers.

ad 3) An operation that uses the mouse to "grab" a data item from one window, "drag" it across the screen, and "drop" it into a second window is referred to as a Drag-and-Drop action. The data item may be a DDC number, a word or phrase, a hit list, or other data type, e.g. if a DDC number is dropped into a DDC Record window, the record corresponding to that number is displayed in the DDC Record window.

For more information contact: Diane Vizine-Goetz, Consulting Research Scientist, OCLC Office of Research; vizine@oclc.org.

TERM-LIST for Electronic Terminological Networking

TERM-LIST is an electronic discussion forum for scholars, teachers, students and others interested in terminology science, terminological research, terminology work, knowledge representation, classification, and LSP-research without any geographical or chronological boundaries. TERM-LIST is an electronic mailing list based at the University of Vaasa, Finland. Subscription is free. The goal of TERM-LIST is to provide members with a fast, convenient, and relevant electronic discussion forum that focuses on issues related to terminology science. Subscribe by sending the following e-mail message to LISTSERV@uwasa.fl SUBSCRIBE TERM-LIST Capital or lower case does not matter, but spelling does; note spelling of LISTSERV. Do not add your name in the message.

Questions about list membership, management, or direction should be sent to the list-owners: Anita Nuopponen (atn@uwasa.fl), Outi Järvi (oja@uwasa.fl). University of Vaasa, Department of Communication Studies, POB 700, FIN-65101 Vaasa, Finland, Tel.+358-61-3248-11, Fax: +358-61-3248-380.

NASA Thesaurus Listserv Established

A new NASA STI Program Listserv, designated THESAURUS-L, has been created to encourage and broaden user participation in the development of the NASA Thesaurus. More specifically, the e-mail Listserv will assume and support the following functions: Provide regular, timely announcements of new Thesaurus terms and changes, and support and encourage the electronic submission and discussion of new term requests, questions, and other issues related to the NASA Thesaurus and subject indexing. Send an e-mail message to: Listserv @sti.nasa.gov. Leave the subject line blank. The message should read: subscribe THESAURUS-L <your name>. If you need additional information, contact the CASI Lexicographer, Tel.: 301-621-0114, e-mail mgenuardi@sti.nasa.gov.