Time for Thinking Big in AI

W. Bibel

TU Darmstadt, Computer Science, Intellectics Group
Alexanderstr. 10, D-64283 Darmstadt, Germany
bibel@informatik.tu-darmstadt.de

Abstract. The paper reminds of the vision behind Intellectics, the discipline of Artificial Intelligence (AI) and Cognitive Science, and argues that the time has come for attacking fundamental questions with a number of big interdisciplinary and international projects.

1 Introduction

Today it is commonplace to talk about the revolution going on in our society, driven by information technology. Through talking about the ongoing changes people seem to compensate their uneasiness with the restlessness in this world. Few of those heralding the future developments are actually looking further ahead than perhaps five or ten years. And even fewer still are pondering about the consequences the coming technological changes will have for society and prepare for actions to be planned for coping with these changes.

In this paper written in honor of Jörg Siekmann's sixtieth birthday I want to alert the community for some of the grand tasks which for our field lie not so far ahead on our road into the future. The main message is that our field carries a great responsibility for attacking fundamental questions which have been on the minds of people throughout thousands of years. Only now do we have the methodology and technology to make real progress towards satisfactory answers.

The article starts in the next section with a brief summary of the vision behind Intellectics, the field of Artificial Intelligence (AI) and Cognitive Science, and of the fundamental scientific questions which have come under our responsibility through the success of the research paradigm underlying Intellectics. In Section 3 some of these successes are briefly described which indicate a great maturity of the theoretical and methodological basis of the field reached during its brief history. If after so many successes we would now become more courageous and pursue grander goals than hitherto, a major technological issue is the direction we should follow: do we need just smart ideas to build really intelligent systems or will any such system be necessarily huge. In Section 4 we give arguments for a combination of both directions.

On the basis of all these preparations we finally describe a number of challenge projects with enormous importance for mankind as a whole. Due to the general relevance of intelligence and hence of AI-technology these projects affect basically all other disciplines, in the spirit of the great intellectician Woody Bledsoe who said: “In the end AI will be the only science”. Which does not at all mean that our field wants to overtake all these other disciplines. Rather it means that the Intellectics paradigm is the one which finally has emerged as the succeeding one.

2 Intellectics

Scientific research today produces knowledge at an enormous scale. The volumes of thousands of journals are filled each year with the achieved results. No single scientist is able to oversee the resulting body of science.

However if one looks closer at this confusing variety one recognizes that the scientific enterprise as a whole actually poses only a very few really deep questions. All the other explorations may be seen as sidetracks from the road to answering these basic questions.

What is matter? is one of these fundamental questions. *What are the forces?* and *What is space and time*? are further ones. In fact these fundamental questions have still no really satisfactory answers. And because of that physicists are continuing with their observations of nature and with their experiments. In the pursuit of searching for these answers myriads of results have been found and applied successfully – sometimes to the benefit of mankind. In this sense the fundamental questions proved to be extremely fruitful although perhaps too hard to be ever solved completely. Besides physics other disciplines were created along the way to account for special aspects of matter and forces, such as chemistry, astronomy, geology and so forth. These "physical" disciplines altogether provide insight into many phenomena in this world, but not into all.

What is life? is another fundamental question and one of those which (so far) has stayed out of reach for any physical theory. There must be something particular beyond matter and forces which characterizes species such as man. Of course, a dog consists of matter which is subject to the physical forces. But why do dogs have offspring while stones don't? Biology is the scientific discipline competent for these sorts of questions. And again there are several related ("biological") disciplines such as botany, zoology, physiology and so forth. They together with the physical disciplines provide insight into even more phenomena in this world, but again not into all.

What are the mind and the psyche? is the most prominent fundamental question which has stayed completely out of reach for any biological and physical theory, and the same is true for all aspects related to it. This question as well as all those mentioned so far have occupied the thinking for thousands of years. The oldest extensive treatments which are still accessible to us are those from Aristoteles, Platon and others from that time. They may be regarded as the founders of the discipline regarding itself as competent in all of these questions, namely philosophy. In the course of time again other disciplines grew out of it, classified together as the humanities.

The answers of the old philosophers are completely outdated today as far as the physical and biological questions are concerned. Philosophy therefore retracted its focus to the mental and psychological sphere in which Aristotle's and

Platon's answers are still discussed in ernest and cited along with modern publications. This shows that the progress in the pursuit of studying mind, psyche and the phenomena related with them such as human relations, society, history and so forth has been less spectacular to put it mildly.

The advent of the modern computer immediately caused a revolution in the paradigm with which these issues were attacked, now by scientists trained in disciplines not related with philosophy and the humanities. Alan Turing, John von Neumann, John McCarthy, Marvin Minsky, Herbert Simon, Alan Newell are some of those who in the middle of the last century laid the foundation for the new paradigm with which these questions are studied under a completely new perspective. While there was literally no measurable progress in two thousand years of research under the philosophical paradigm, the last half century has already provided us with revolutionary insights into the functioning of mind and psyche. On this basis it is now possible to penetrate into further domains which to this day continue to be dominated by the old paradigms.

John McCarthy is usually mentioned as the inventor of the name for the discipline which has been founded on this new paradigm, namely Artificial Intelligence or AI. But that is not the full story. He prepared a proposal for the famous Dartmouth workshop in 1956 with a title mentioning "artificial intelligence" which was not meant by him as the name of such a new discipline however. In fact later he tried to introduce Cognetics as such a name. But the buzzword AI had already caught on and has stayed until this day.

The goals of the discipline of AI are to understand intelligence, mind and psyche in a way so as to be able to produce these phenomena even artificially. In other words, artificial intelligence is what this discipline wants to achieve (among other things) and is therefore a perfect name for its *goal* rather than for the discipline itself. But for strange reasons AI became the only discipline which uses the name of its goal at the same time for its own name. Since this is strange indeed and unique in the concert of disciplines and their names, I proposed *intellectics* as its name instead of AI. As long as no better proposal is put forward I will continue to stay with this proposal which, after two decades of familiarity, to me still sounds perfectly.

Intellectics then is the discipline for the study of mind and psyche in biological creatures such as man under the paradigm of intelligent information processing which, as we said, includes the development of artificial systems with intelligent features. Although we mentioned the key players whose ideas directly lead to Intellectics as a discipline they were of course embedded in the context of a particular Zeitgeist which produced this shift in paradigm and could be traced in various disciplines. Warren McCulloch, Walter Pitts, Donald Hebb and Claude Shannon are usually mentioned as pioneers in the years before 1956. But there are many more who would deserve to be mentioned in a comprehensive analysis of the history of Intellectics. Konrad Zuse would be among them with his Plankalkül and his chess program, the first in history (prior to Shannon's), the German school of cognitive (especially Gestalt) psychologists with names such as Otto Selz, Max Wertheimer and Karl Duncker, the physicist and physiologist Herrmann von Helmholtz, and, even further back into history, the logical tradition from Gottfried Wilhelm Leibniz to Gottlob Frege.

In the seventies it turned out that one can earn even money with intelligent systems, in fact a lot of money. This experience distracted the discipline somewhat because from there on the public identified AI with that part of intellectics which builds smart systems. Due to this one-sided focus the researchers with a remaining interest in the original questions about mind, psyche and cognition reassembled under the new label of Cognitive Science which focuses on the "non-AI" part in Intellectics. So today Intellectics may seem to consist of two parts, namely AI and cognitive science. One could as well think of these three as different names for the same thing, a position I prefer since a separation of AI from Cognitive Science would certainly hurt both.

Even when one restricts the focus on the Computer Science part of AI, its perspective has been different from, in fact complementary to, that of CS (or Informatics). Intellecticians from the beginning thought of systems simulating people in their mental functions (how they play games, prove theorems, understand language, etc.). In other words they imagined artificial agents performing such functions and tried to realize them on existing computers. Note the semantic ordering in this statement: given are first the functions and then we try to model them with current machines. Computer scientists think the other way around. They take the functionalities of computers as granted and then think about what other functionalities could be achieved with them, for instance functionalities for a more comfortable human interface. This thinking is bottom-up-oriented while that of intellecticians is top-down where man is taken to be top and machine down.

3 The success of the new paradigm

Why does the mind pose such a hard problem for science? Due to research in Intellectics – or more specifically in cognitive psychology, computational neurology and computer science –, we know today quite a bit about the functioning of the brain. For instance, the processes involved in color seeing have been discovered to an extent that they can be modeled and simulated with computers at a rather low level of detail, namely that of individual neurons. On the basis of these and many other insights it is clear that computational processes are heavily – if not even exclusively – involved in the functioning of mind. That is why often the analogy between the mind and a computer is drawn. Let us take this analogy to illustrate the particular difficulty in understanding the mind.

Imagine aliens coming to earth and watching us using our computers. Before they return home they take lots of notes about their observations of the functional behavior of these machines and carry some machines for further studies with them. Their goal is to understand how the machines work, without having access to our computer science literature, manuals etc. They would be in a similar position as we are with our own minds except that we in addition are conscious of our own thinking. What could they do to find such an understanding? Continue making observations about the functional behavior and forming hypotheses concerning the way this behavior is generated, just like the psychologists in the case of mind. Or open the machines, observe their static structures, trace the currencies through the chips and analyse this internal functioning of the gates, just like the neurologists. Wouldn't it be a formidably difficult task to discover this way the principles on which modern computers are based and the details of their design? Since brains at least to some extent are computational like computers we are faced with the same kind of problems at a far higher quantitative level of complexity.

The nature of this problem consists in discovering the ideas behind a program from its functional behavior and from fragmentary observations of the flow of signals. Even for relatively simple algorithms we have not yet the right methodology for solving such a problem, let alone for "algorithms" involving trillions of neurons. As a first step it is therefore reasonable to try to analyze the functional behavior by inventing algorithms which produce it. These algorithms might be rather different from the actual ones, but at least we would get a first idea about their structure.

That is what we intellecticians do with our AI systems. For instance, neural nets were invented in this vein which became extremely helpful in modeling neural processes at a far greater detail than possible before. But basically all AI systems may serve such a purpose. These systems give us a lot of clues about human problem solving so that we can go on from there with our studies of the real system, ie. the brain. There is already considerable progress in this line of attacking the fundamental questions about mind and psyche. On the other hand, these systems have turned out to be extremely helpful in numerous applications in the real world. It is especially in this respect that Intellectics so far has been particularly successful. Since it would fill many books to compile all these successes, we can only mention a few selected ones.

Scientific disciplines often organize their research around the study of compactly understandable problems. The problem of understanding the physiological functioning of the *drosophila* in full detail is such a paradigmatic problem in biology. The fly drosophila is complex enough to make this study really interesting; on the other hand it is not that complex that with the current methodology progress is out of reach. Intellectics has had two "drosophilas" so far. The first one was chess. The study of playing chess lead to many interesting concepts for modeling intelligent behavior. On the basis of these concepts and the resulting theories eventually systems were developed which reached even the level of the best human chess players in the world. In 1997 the worldmaster in chess, Gary Kasparov, was beaten by such a chess machine (deep blue). No one claims from this that now we would "understand" how the mind is able to play chess. At the lowest level of detail there is still a huge gap between the algorithms realized in deep blue and those presumably at work in Kasparov's mind, although they behave very similar at the highest functional level. Nevertheless the high level success provides a starting point from which we now can proceed to lower levels of details for a deeper understanding (and possibly better technological performance).

The second “drosophila” of Intellectics is robocup. Once the chess problem was “settled” the next challenge needed to take a step closer to the human condition in toto. We humans are not just minds which solve problems like chess but we also have a body which must be coordinated in addition to solving real problems in the world. Also we are social beings cooperating with others in our problem solving. All these features are given in the task of soccer playing. So in the mid-nineties the idea came up to develop artificial teams of soccer players. The longterm goal of this second “drosophila” in AI is to beat the worldmaster team in 2050. So leagues were set up and matches among teams of robots held, altogether called *robocup*. Given the few years of research, the progress is fascinating already now.

Mathematics is too sober to catch the interest of the public to an extent games such as chess or soccer do. But from an intellectual point of view beating the worldmaster in chess is far less spectacular than beating the best mathematicians within their own domain. This is what happened in 1996 when the artificial mathematicians EQP und Otter, so-called theorem provers, finally proved a conjecture [1] which in mathematics was open for sixty years and in vain attempted to be settled by a number of the best human mathematicians such as Alfred Tarski. Recall that mathematics has been called the queen of sciences and that mathematicians are for good reasons usually kept in highest regards concerning their intellectual capacities. And then imagine that a computer system outperforms some of the best mathematicians of the century by solving a problem they could not solve for such a long time. I feel that this event is a much more impressive achievement in our discipline than the chess event mentioned before.

Society longs for the spectacular shows such as robocup matches while the real progress may be experienced in many far less spectacular achievements. Let us summarize a few more of these achievements. Doug Smith with his system KIDS synthesized a logistics algorithm – among many others – which outperformed all similar algorithms published by human specialists by orders of magnitude [2]. Unnoticed even by many computer scientists, we are entering the period in which much of programming will be carried out by the computers themselves rather than by software engineers. What is left to humans is the specification of their problems in sufficient detail but without consideration of any aspects concerning the machine which today are still encoded in programming languages like C or Java. It is therefore a safe prediction that in the near future, “problem engineers” will complement – and eventually substitute – software engineers.

There are tens of thousands of systems in daily use which to some extent are based on coded knowledge of the kind we as students learn in books about any discipline whatsoever. These *knowledge-based systems* are routinely integrated in various system environments without special notice. Thirty years ago people could not even imagine that knowledge coded in such a declarative form could be processed algorithmically. Today KB systems provide for complex kinds of reasoning modes, so far the exclusive domain of people. The system CYC [3] in this way comprises general knowledge at the scale of a million knowledge facts or rules according to the state reached this year, and can activate this huge amount of general knowledge in various kindes of applications. The system Verbmobil allows the spontaneous communication between a German and a Japanese in their respective natural languages through the system's interpretation [4].

Vision systems are used in production lines for controlling the production processes, but also analyze the pictures taken of entire continents, to mention two out of hundreds of other applications. Robots are now used for rather complex tasks far beyond the restricted production manipulations applied in the previous generation. Spectacular uses could be experienced during recent space missions or during the search for victims of the terrorist attack in New York after the 11th September 2001. Systems connected to communication networks are able to trace nearly everything happening anywhere around the globe, be it a phone call, a fax, an email message etc. and analyze the data for certain features.

One could go on and on with the technological achievements enhancing our human cognitive capabilities. What is at least as impressive as the preformance of these systems is the theoretical insight into the underlying mechanisms. Obviously the spectacular systems performance is a function of the maturity of the theorical basis on which it is founded. True, sometimes engineers seem to be ahead in comparison to theoreticians. In building complex systems like those referred to here you are quickly lost however if the gap between theory and practice is more than marginal.

In fact we may observe the following pattern in the theory of Intellectics. Since so many different phenomena are involved, the theory is fragmented in many different branches. The results in each of them are applied independently from those in others although they pertain to the common phenomenon of intelligent behavior. Imagine all these disparate results were combined within a common system!

4 Big or smart?

Huge **and** smart! In social life people tend to fight over dichotomies such as efficiency-oriented vs. socially-oriented economic politics. Is huge vs. smart such a dichotomie on the road to artificial minds? Well, like most dichotomies in politics huge and smart are not really opposites in view of intelligent systems. The central nervous system of humans has around a trillion of neurons, each forming an extremely complex physical processor, and around a hundred times as many synapses (connections). Given this unimaginable complexity one would of course not expect that all the relevant functions of the brain could be modeled by a single system comprising a few thousand lines of code. A truly intelligent system will predictably be big, even huge.

The development of such huge systems will remain a challenge, perhaps for ever. The software industry tries to make us believe that it is up to this challenge in areas like operating systems with appropriate solutions. Sincere computer scientists admit that the progress in this art is rather like that in the humanities. Not accidentally so since system development requires the cooperation of many humans hence falls in fact to some extent into the domain of the humanities. So big is a problem and huge even more so.

How about smart? Let us consider an interesting experience in this respect. Theorem proving is a thoroughly studied special field in Intellectics. Hundreds of systems, so-called theorem provers have been developed, one of which we mentioned in the last section. Some of them took tens if not hundreds of man-years of development and comprise hundreds of thousands of lines of code. Jens Otten during one night in the year 2000 wrote a Prolog program consisting of three Prolog clauses which more a less amounts to the logical definition of the simplest calculus based on the connection method developed by the author [5, 6]. The surprising experience was the performance of this mini-prover, called leanCoP: in some sense it is comparable in performance to that of those mammut provers as careful and extensive comparisons have demonstrated [7].

leanCoP is surely a smart system in several respects. First, it is based on a proof method which, unlike any other known method, operates on a single formal representation of the problem description. The advantage of this locality feature has been ignored by the community which has preferred the more intuitive (and hence less focused) methods such as the resolution or the tableau methods as their logical basis. The connection method is also goal-oriented rather than an exhaustive method like resolution. The system also makes clever use of Prolog's features. In combination these features make the tiny system surprisingly efficient.

There is one more point to be noticed about leanCoP. The program forms a precise and declarative specification of the problem, nothing more. Recall that those supporting logic programming (like the present author) have dreamed of logical problem specifications as programs to be immediately processed by machines. Several fashion phases in software engineering after those dreams have begun I still believe that eventually this will be the only reasonable approach to system building. The logic will probably be different from the one used in Prolog, especially to account better for the dynamic character of the problems to be coped with in programming (cf. [8]). Also the system development environment will have to support the programmer, or rather the specifyer, in various novel ways unfamiliar from standard programming environments (cf. [9]). UML is a step towards such a direction but an insufficient one in many respects. And finally the synthesis and optimization systems which transform specifications to efficient code will have to be improved quite a bit further, heavily involving KB systems technology discussed in the last section.

The attractiveness of specifications in contrast to traditional programs lies in the additivity of specifications parts. An enhancement of leanCoP for a special handling of equality would amount to an addition of further Prolog clauses without affecting the definition given in the three current clauses. No other programming paradigm offers this fundamental advantage (including OO programming). If one envisions huge systems as we do here this advantage will become absolutely crucial for the success of the entire enterprise.

Another big advantage lies in the ease of maintenance. A change in the specification could be achieved simply by substituting the respective specificational part in the problem definition which is kept along with the resulting code as well as with all relevant synthesis information. The synthesis would therefore not need to be re-done in its entirety upon such a change but only the respective modifications would have to be traced down to the executive code.

Programming by specification may lead to smarter systems. That is the lesson which I wanted to suggest with the leanCoP experience. In contrast, standard programming results in monster systems like Windows, huge because of lack of appropriate organization, not so much because of its inherent functional complexity which could be achieved with much smaller and smarter systems. Good organization can only be achieved with a proper, elaborated specification available at the outset, or rather growing with the system development, and with systematic ways to synthesize such a specification into efficient code as studied in program synthesis.

Once we have such systematic ways to synthesize smart systems, then – and only then – will we be able to envision the development of huge systems needed to model the mind. In principle we know these ways already, but there is no incentive to realize them in respective development environments. The reason is the enormous investment needed prior to any benefits before such an approach becomes feasible. We are talking about such investments in the coming section.

5 Challenge projects for Intellectics

Society has always found it worthwhile to invest enormous sums in projects related with the fundamental questions listed in Section 1. The physical accelerators built for studying physical forces and the structure of matter cost billions of dollars. The international space station (ISS) also is only secondarily of economic interest and primarily an enterprise about answering fundamental scientific questions. It as well costs billions of dollars.

In contrast to projects at this scale Intellectics projects were rather cheap so far. In addition most of the latter soon payed off. For instance, the project CYC is in the (low) hundreds of millions as is Verbmobil. Both have a great economic perspective so that they may even produce large returns before long. Similar returns from the huge investments in accelerators are not expectable in the forseeable future.

Although this might sound like arguing against big physical experiments, this would be a total misunderstanding of my point. Rather the point is that the fundamental questions pursued by Intellectics are at least as interesting for society as those pursued by Physics. Also we will not come closer to satisfactory answers unless society is ready to invest in Intellectics projects at the same scale which is billions of dollars. According to our previous experiences we might even expect an immediate economic return for society so that the investment promises to become profitable. In this final section I will briefly outline a few projects at this scale and their potential benefits for society.

I first refer to what I already outlined in the last section concerning a systematic synthesis of computer systems through precise specifications. Building a development environment of the kind envisioned there amounts to an international project with costs at the level of billions of dollars. Its benefits are obvious and immediate. Software production would be simplified by orders of magnitude. System development would be possible in a fraction of time in comparison to today's practice. It also would lead to machine provably correct systems and thus far more reliable than current software. Maintenance would be simplified again by orders of magnitude.

The consequences of this single project alone on other disciplines and their applications would be enormous. Once we have the methodology to design and realize software at such a new level of magnitude, we could envision the grand project noted at the end of Section 3 which combines the disparate knowledge built up in the theory of Intellectics in the last decade. With the resulting smart and huge system grand problems could be attacked. Some of these are the following ones.

Following the vision and scientific work of nobel prize winner Herbert Simon, discovery science is developing at a rapid pace [10]. Imagine if we complemented the inventiveness of humans with the strengths of machines for handling massive data. Physicists, for instance, try to discover patterns in their scattering experiments in accelerators which could give clues for a unified theory of the forces. There are billions of data already available from past experiments of this kind which no human will ever study in any detail. Machines could process these amounts of data, discover such patterns and mechanically form theories accounting for them. For physicists this may sound like wishful thinking. For intellecticians this is already the bread and butter of their daily work for instance in the area of data mining where knowledge patterns are extracted from massive data. Physics is just one example. The mechanisms behind learning and theory formation are now understood good enough to apply this knowledge by machines to any of the natural sciences. But the first step for any of these applications would be to put all the knowledge available for any of these disciplines in a formalized knowledge base which can be subjected to our inferential mechanisms. This alone just for a single discipline amounts to a huge enterprise.

With our envisioned intelligent system also engineering of any kind would be revolutionized, a trend which to a limited extent and with limited system capacity is already under way. Imagine we are given an engineering problem (like building a bridge under given constraints or controlling traffic in a city). The knowledge about the domain under consideration is assumed to be available in the knowledge base of the system. The problem could then be specified using the terms of this knowledge base in the way we specify programming problems (such as theorem proving in the case of leanCoP). With the system development environment an engineering design could be synthesized like a program from this specification and the knowledge base. This is because, from an abstract point of view, the design of a program is exactly the same as the design of an engineering solution to the posed problem. Whether the quality of the solution is good enough for providing a solution in the real world depends solely on the quality of the engineering knowledge stored in the knowledge base.

Let us emphasize once more the arbitrariness of the kind of engineering applications. The fundamental design methods are all very similar at the simulation level in computers. The difference lies mostly in the relevant knowledge activated from the knowledge base if it is rich enough to cover the respective areas and includes area specific procedures. So whether we are thinking of drug design in pharmacology or of designing an optimized burocracy for handling a university or of a strategy for improved teaching to achieve better school results or what have you, all these fall – from the point of view of Intellectics modeling in computers – under the same category of engineering problems. Engineers can hardly believe this as they rarely reflect about their own methods on a meta-level. Their strength consists in embodying that specialized knowledge combined with clever solution methods. Like most experts they are however not consciously aware of them.

As indicated with the last of these examples (teaching) Intellectics technology would finally reach also the humanities. There is already a huge amount of knowledge available in these disciplines because in the humanities in lack of theories researchers collect a lot of episodical and statistical knowledge. Just think of psychology or sociology. As in any knowledge base there is a great potential in these bodies of knowledge once they become accessible for the inferential methods from Intellectics. Theories could be induced from that knowledge, possibly so complicated ones that humans would never be able to induce them by hand due to the shere (quantitative) complexity. Explanations could be provided on the basis of that knowledge. Further knowledge could be deduced. Modeling could be performed on a solid basis. In short, the available knowledge would become activated to an extent not experienced ever before.

Particular goals of this nature consist in modeling (parts of) the body of particular human individuals on the basis of the entire knowledge available about body physiology in general and the specific body of the individual in particular which could provide precise explanations for symptoms exceeding the daily speculations of ordinary medical doctors by far. And the modeling could be done at rather low costs once the overall system is available. Or think of the modeling of the functioning of the brain in accordance with the knowledge neurologists have accumulated over the last decades. For parts like image and sounds processing this has already been done. The challenge is to combine these fragments to a coherent model which might vary considerably in the level of abstraction depending on the availability of insights. Similarly, the human psyche could be modeled on the basis of the knowledge about the functional relationships elaborated in numerous psychological studies. On the basis of both these models we could experiment about different strategies for school teaching, first in the computer and then tested in real life, ie. enter the discipline of pedagogy with quantified methods. Going a step higher, sociological structures like a small-size company could be modeled, linked to the model of the human psyche mentioned just before. This way humanities would become real sciences rather than carrying along the odor of being chat fora.

With Intellectics technology also decision processes in economics, politics and law could be laid on rational grounds. In individual projects the direction of such decision support systems has already been shown. But again, individual projects like PhD theses and one-man prototype systems are interesting for understanding the underlying principles; however, for practical purposes we need systems build on a much grander scale. In all applications just mentioned we know in principle how to do it. But billions of dollars are needed to realize the required programs. Billions of dollars is little money compared to the amount of waste of money caused every year by irrational and hence often stupid economic and political decisions.

Some people argue that money alone will not help. The problems in such huge projects would be so hard to cope with that not enough smart people with the appropriate expertise would be available to carry the projects through to a successful end. I consider such arguments narrow-minded because they ignore the enormous expertise available in the application disciplines. As experiences with interdisciplinary projects of this kind show, people involved get really excited about the prospects of the resulting achievements and thus are highly motivated. High motivation counts much more than the question whether some participant might have attended a certain course in Computer Science or not.

Like with all scientific endeavors dangers lure around the corner of such projects of the kind of possible misuse by villains. We must face the fact that this menace is inherent in any technology. Already in the stone age, Kain could use the stone to grind the corn or to murder Abel. The same will be true for the technology envisioned here. We have to cope with this fact now as ever by prosecuting villains in order to keep their actions as restricted as possible.

References

1. William McCune. Solution of the Robbins problem. Journal of Automated Reasoning, 19(3):263-276, December 1997.
2. Douglas R. Smith. KIDS - A Knowledge-Based Software Development System. In M. R. Lowry and R. McCartney (Hg.), Automating Software Design, Kapitel 19, S. 483-514. AAAI Press, Menlo Park, 1991.
3. R. V. Guha and Douglas B. Lenat. Cyc: a midterm report. AI Magazine, ll(3):32-59, 1990.
4. Wolfgang Wahlster (Hg.). Verbmobil: Foundations of Speech-to-Speech Translation. Springer, Berlin, 2000.
5. W. Bibel. Matings in Matrices. Comm. ACM, 26:844-852, 1983.
6. W. Bibel. Deduction: Automated Logic. Academic Press, London, 1993.
7. Jens Otten and Wolfgang Bibel. leanCoP: Lean Connection-Based Theorem Proving. Journal of Symbolic Computation, 2001.
8. W. Bibel. Let's plan it deductively! Artificial Intelligence, 103(1-2):183-208, 1998.
9. W. Bibel, D. Korn, C. Kreitz, F. Kurucz, J. Otten, S. Schmitt and G. Stolpmann. A multilevel approach to program synthesis. In N. E. Fuchs (Hg.), Proceedings of the 7th Workshop on Logie Program Synthesis and Transformation (LOPSTR-97), Lecture Notes in Computer Science vol. Springer, Berlin, 1997.

10. Lindley Darden. Discovering Mechanisms: A Computational Philosophy of Science Perspective. In Klaus P. Jantke and Ayumi Shinohara (Hg.), Discovery Science - 4th International Conference, DS 2001, Washington DC, volume 2226 of LNAI, S. 3-15, Berlin, 2001. Springer.

This article was processed using the LaTeX macro package with LLNCS style