AN OVERVIEW OF FORMALISMS, ISSUES, LANGUAGES AND SYSTEMS FOR THE REPRESENTATION OF KNOWLEDGE

Bibel, W., Bruynooghe, M., Freksa, C., Haddawy, P., Moreau, M.

Report ATP-22-XII-83

December 1983

AN OVERVIEW OF FORMALISMS, ISSUES, LANGUAGES, AND SYSTEMS FOR THE REPRESENTATION OF KNOWLEDGE[+]

Wolfgang Bibel[*]
Maurice Bruynooghe[§]
Christian Freksa[*]
Peter Haddawy[*]
Michel Moreau[£]

[*] Technische Universität München
[§] Katholieke Universiteit Leuven
[£] Belgian Institute of Management

Abstract.

This report surveys the state of the art in knowledge representation, a central field in the area of artificial intelligence. Five basic approaches to knowledge representation are discussed; issues relevant to representation systems are identified; prototypes of implemented representation languages are presented. The report contains an extensive bibliography.

[+] This work has been supported by the ESPRIT Project of the European Communities.

CONTENTS

0 INTRODUCTION

Knowledge representation is one of the core areas of research in Artificial Intelligence (AI). Initially unformalized world knowledge is to be represented in such a way that artificial reasoning systems can assimilate this knowledge, process it, and convey results to the user or the application domain in a natural fashion.

This report is a survey of knowledge representation systems and of the issues central to the development of representation formalisms. Relevant representation formalisms are identified along with the control concepts used therein. No attempt has been made to cover all work in this broad field; rather, we aim at exposing key concepts and trends. The report contains four parts:

1) a description of existing knowledge representation formalisms and their properties;

2) a discussion of the issues particularly relevant to the development of representation systems;

3) a selection of implemented knowledge representation languages and expert system shells; emphasis is on a high-level description of their capabilities and limitations;

4) an extensive bibliography.

A method of systematically representing knowledge is called a knowledge representation 'scheme' (Hayes 1974, Mylopoulos and Levesque 1983). A fundamental problem consists of grasping the <u>meaning</u> or <u>semantics</u> of an expression within such a scheme. Different representation schemes can only be compared with the aid of a formal definition of their semantics (Hayes 1974). First-order logic is a very well developed language which has been much used to describe the semantics of (expressions in) particular representation schemes.

All representation schemes are biased towards specific ways of using the knowledge. They favor certain inference schemes. This study surveys the biases inherent in different representation schemes. We also survey to what extent these inference schemes have been described as metalevel knowledge. This is important for the comparison of different formalisms and for our goal of designing formalisms which are flexible in the choice of their inference schemes. In chapter 1, we will review basic classes of existing knowledge representation formalisms and in chapter 2, we will expose issues which are relevant towards the development of the knowledge representation system for the LOKI project.

We wish to acknowledge the discussions with our partners in the ESPRIT 3.3 pilot phase, particularly with Horst Adler and John Mylopoulos. John Mylopoulos provided us with much material and advise which was incorporated into this project.

1 BASIC APPROACHES TO KNOWLEDGE REPRESENTATION

We consider five basic knowledge representation schemes: logic, semantic net, procedural, frame-based, and direct representation. Each of these approaches is discussed here separately.

1.1 Logic

Classical first-order logic is the oldest and best investigated approach to knowledge representation. It consists of a declarative component: facts or axioms about the world; and a procedural component: rules of inference to derive new facts from existing ones. In early knowledge representation systems based on logic, the rules of inference were not goal-directed; therefore they created a large search space. A skilled human expert was required to establish a proof of a particular problem.

The introduction of the resolution principle (Robinson 1965), a single inference rule allowing for all possible deductions in first-order logic, was creating great expectations towards automating the construction of proofs. However, the developed systems were extremely inefficient. This resulted in the dismissal of logic by some schools. A number of researchers, for example Hayes (1973, 1974, 1977), Kowalski (1974, 1979), and Bibel (1975, 1983c) argued that not the logic was at fault but that it lacked the information necessary to constrain the search space.

Kowalski (1974) proposed to use a subset of first-order logic, the so-called Horn clauses, and gave them a simple procedural interpretation. Horn clauses allow the formulation of pieces of knowledge in such a way that they are dedicated to solve particular problems, thus allowing a specialized theorem prover to derive proofs efficiently. Colmerauer implemented the programming language PROLOG (Colmerauer et al. 1973) which is based on Horn clause logic.

Kowalski (1979) subsequently showed that different computational behavior may arise through different formulations, although logically equivalent, and also by applying different control structures to the same logic formulation.

In the logic programming field, some workers stress the logic aspect, others emphasize the programming aspect. Several attempts have been made to enlarge the expressive power of logic programming. This work takes place on several levels,

- on the pragmatic level: enhancement of the expressive power, e.g. negation, set-of;

- on the theoretical level: giving a theoretical underpinning to pragmatic extensions, e.g. negation as failure (Clark 1978), amalgamation of language and meta-language (Bowen and Kowalski 1982);

- on the control level: development of more flexible control mechanisms, e.g. IC-Prolog with its annotations (Clark and McCabe 1979), Colmerauer's Prolog II (Colmerauer 1983) with its suspension of insufficiently instantiated inequalities, Naish's MU-Prolog (Naish 1982) with its more general suspension of insufficiently instantiated calls, intelligent backtracking (Bruynooghe and Pereira 1981).

Hayes' work is of particular relevance to us (Hayes 1974, 1978). He attempts to uncover the semantics of different representation formalisms by translating them into logically equivalent formulations in first-order logic. Hayes is aware that there is a problem of controlling the inferencing process but he claims this problem is independent of the particular representation formalism (Hayes 1981).

The work of Reiter (1978b, 1983) is similar to Hayes' but oriented towards data bases. Reiter investigates the semantics of various assumptions such as disjunctive information, null values, is-a hierarchies and inheritance of properties, aggregations. He axiomatizes these concepts by logical reconstruction, thus defining a theoretical framework

in which to judge the correctness of their handling in a practical system.

Weyhrauch (1980) has developed a reasoning system FOL using full first-order logic with equality and with sorts as its knowledge representation formalism. In contrast to other deductive systems (like PROLOG, AURA, etc.) FOL does not aim at a full automation of the reasoning process; rather it is intended to support the user's reasoning in a dialogue. Its reasoning capabilities are based on the natural deduction rules of inference (cf. section 1.2.4.3) as well as a syntactic simplification (rewrite rules, section 1.2.4.4). This system has several features of interest for future knowledge representation formalisms; they are described in section 3.

The LOPS system (Bibel and Hoernig 1983) also uses full first-order logic as its knowledge representation formalism. It is a programming support system with a bias towards a strong deductive component.

Logic has been advocated particularly as a formalism which allows the integration of knowledge and meta-knowledge, an issue addressed by Weyhrauch (1980), Gallaire and Lasserre (1982), Bowen and Kowalski (1982), and Smith (1982).

1.2 Semantic nets

Semantic networks represent knowledge as structures of nodes connected by links. The nodes correspond to concepts (types) or objects (tokens) and the links correspond to binary relations or predicates over these. Semantic nets were first introduced by Quillian (1968) as a psychological memory model for encoding the meaning of words and have since been used extensively for research in cognitive psychology (Anderson and Bower 1973, Rumelhart and Norman 1975) and linguistics (Simmons 1973), as well as in artificial intelligence. They offer a natural, easily comprehended framework for representing knowledge.

Implemented semantic network languages include Shapiro's SNePS system (Shapiro 1979), Hendrix' partitioned semantic network formalism (Hendrix 1975, 1979), Schubert's network formalism (Schubert 1976), and Brachman's KL-ONE (Brachman 1979). Semantic networks offer the important capability of globally structuring the knowledge base as well as some special inference rules over this knowledge. This structuring has been broken down by Mylopoulos and Levesque (1983) into four 'organizational axes':

- classification
- generalization
- aggregation
- partition

Classification is the association of each token with a generic type whereas generalization relates a specific type to more general types. Nilsson (1980) makes this distinction by representing the former with 'element-of' links and the latter with 'subset-of' links. Other researchers combine these two axes, exemplified by so-called 'is-a hierarchies'. The distinction between types, tokens, or intensional & extensional entities is important, as Woods (1975) points out. Any representation scheme, however, must in certain instances be capable of using types as tokens and vice versa. This is an important factor in the flexibility of human reasoning (Hofstadter 1979). Brachman (1977a) proposes a representation scheme which allows this.

Aggregation is the relation of an object with its parts. Partitions are hierarchical structures in which everything visible to lower partitions is also visible to those higher up. This is useful for representation of time, hypothetical worlds, and belief spaces. Hierarchical representation of knowledge is also the basis for special inference rules for inheritance of traits and default reasoning applicable to semantic nets (Nilsson 1980). Inheritance of traits allows attributes of objects or types higher up in the hierarchy to be passed on to those lower down.

Despite the immediate appeal of the semantic network formalism, serious problems surface upon closer investigation. There is an inherent difficulty in representing relations between more than two objects. Woods (1975) points out the need to differentiate between different types of links such as attributes and relations. Brachman (1977a) proposes a solution to this in form of special purpose primitive links. In addition, the lack of standard terminology and formal semantics in the area serves to cloud the discussion of these and other problems. The relationship between logic and semantic nets has been studied by Schubert (1976) and by Deliyanni and Kowalski (1979).

1.3 Procedural representation of knowledge

In contrast to declarative representations which stress static aspects of knowledge (facts, events, and their relations), procedural representations focus on dynamic aspects of knowledge (how to find facts, how to make inferences, etc.). Representation formalisms like logic and semantic nets became unwieldly when researchers attempted to move from small toy problems to more complex applications because of the need for complete problem descriptions. The need for investigating how to efficiently manipulate knowledge became a dominating concern; the usefulness of domain knowledge was found to be intrinsically bound with the specialized knowledge about how it is to be used (Hewitt 1975).

Procedural representation systems incorporate ways for explicitely expressing control information in order to direct their problem-solving activity. Thus, partial solutions to problems which are not defined on the entire problem space can be admitted. As a consequence, heuristic information which typically is based on incomplete knowledge can be easily treated (Reiter 1978). In addition, procedural systems allow for specifying which knowledge should be used to solve a given problem.

PLANNER, the prototype for a procedural representation language (Hewitt 1972), was designed to expedite the inferences that were expected to be needed within an actual reasoning process rather than facilitate the class of inferences that were logically possible. The nature of the possible inferences did not change by these inference-guiding control features. However, other inference methods beyond those found in classical logic systems can be implemented. This includes non-monotonic reasoning and the various forms of default reasoning.

Production systems (Newell and Simon 1972, Waterman and Hayes-Roth 1978) offer a procedural scheme that is in many ways similar to PLANNER. A knowledge base is a collection of production rules and a global data base. Production rules, like PLANNER theorems, consist of a pattern and a body involving one or more actions. The data base begins in some initial state and rules are tried out in some prespecified order until one is found whose pattern matches the data base. The body of that rule is then executed and matching of other rules continues. This account is an idealization of production systems and most of them vary in the form of rules and in the order in which they are tried (Davis and King 1977). Production systems have been used extensively for the implementation of expert systems (Nilsson 1980).

1.4 Frame-based representations

Since 1974, when Minsky proposed the notion of frame (Minsky 1975), it has played a key role in knowledge representation research. A frame is a complex data structure for representing stereotypical situations such as being in a certain kind of living room or going to a child's birthday party. The frame has slots for the objects that play a role in the stereotypical situation, for relations between these slots, and also for frames themselves. Attached to each frame are different kinds of information such as how to use it, what to do if something unexpected happens, default values for each slot, etc. A knowledge base is now a collection of frames organized in terms of some of the organizational axes discussed earlier but also other 'looser' principles such as the notion of similarity between two frames. Schank and Abelson's (1977) scripts have much in common with the concept of frames.

Examples for frame representation languages include FRL (Goldstein and Roberts 1977), KRL (Bobrow et al. 1977), OWL (Szolovits et al. 1977), AIMDS (Sridharan 1978), KL-ONE (Brachman 1979). Frames can be viewed as a procedural extension of the semantic net formalism. It is unclear how knowledge can be segmented best to obtain 'natural' or efficient representations based on the frame concept.

1.5 Direct (analogical) representations

Direct representations are schemes in which "properties of and relations between parts of the representing configuration represent properties of and relations between parts in a complex represented configuration, so that the structure of the representation gives information about the structure of what is represented" (Sloman 1971). For some problems, direct representation has significant advantages (Palmer 1978, Dirlich et al. 1983). In particular, the problem of updating the representation to reflect changes in the world is simpler, since changing one fact may cause several relations to become implicitely updated. Hayes (1974) points out the difficulties of making precise the distinction between descriptive and direct representations.

Direct representations can facilitate search (Gelernter 1963) and are usually more exhaustive and specific; thus admitting fewer models (Funt 1976) and making for more efficient problem solving. This can be a disadvantage, when more generality is needed for problem solving processes. Certain kinds of incomplete information are difficult to represent in direct representations.

2 ISSUES RELEVANT TO REPRESENTATION SYSTEMS

We now focus on specific issues related to the mapping of real-world knowledge into a representation formalism; the processing of this knowledge; aspects of notation and system environment; forms of logic-based reasoning; and aspects of control interaction between different system levels.

2.1 Representation aspects

During the development of representation formalisms, a number of topics have attracted particular attention. The present section discusses methods and problems related to those topics.

2.1.1 Representing incomplete knowledge, default reasoning, common sense reasoning

Except for situations where a knowledge base models artificial microworlds (e.g. Winograd 1972), it cannot be assumed that the knowledge base is a complete description of the world it is intended to model. This observation has important consequences for the operations defined over a knowledge base (inference, access, matching) as well as the design methodologies for knowledge bases.

In the beginning of AI, much of the work on knowledge representation ignored the issue of incompleteness or dealt with it in an ad hoc way. Examples for systematic approaches to dealing with incomplete knowledge are given by Collins et al. (1975), Moore (1975), Levesque (1983) and Reiter (1983).

Closely connected with the issue of incomplete knowledge is the issue of default reasoning. In every day situations, decisions have to be made in absense of explicit information about certain facts. In

these cases, general knowledge can be very useful for inferring reasonable conclusions. Nilsson (1980) presents a method for default reasoning in semantic nets realized through hierarchical mechanisms. Reiter (1978) and Levesque (1983) develop systematical approaches for default reasoning. Minsky's (1975) frames offer elegant ways for representing and handling incomplete knowledge by means of "slots" and "default fillers".

McCarthy (1980) addresses the issue of "common sense reasoning" from an epistemological point of view, formalizing human reasoning processes based on incomplete knowledge as a special non-monotonic inference method within first-order logic. Zadeh (1983) offers an approach to common sense knowledge representation based on the approximate reasoning paradigm of fuzzy logic. Here, specific knowledge about a restricted domain can be extended by means of "elastic constraints". The elasticity influences the degree of certainty associated with the infered knowledge.

2.1.2 Representing fuzzy knowledge and uncertainty; approximate and plausible reasoning

The role of fuzzy knowledge and uncertainty is a much debated issue in the AI community. The standpoints cover a wide spectrum of views: on one extreme, the position is held that nothing in the world is fuzzy and that efforts should be directed at representing this world rather than the fuzzy knowledge about it. On the other extreme, the position is held that our knowledge about the world is inherently fuzzy and that any intelligent process must have ways of utilizing this type of knowledge.

Among the advocates for representing fuzzy knowledge, the proposed approaches vary widely: some researchers apply probabilistic techniques developed in the context of stochastic events to model fuzziness. They argue, well-developed existing methods are preferable over new approaches which do not exhibit substantially different behavior, in most cases. Other researchers propose to model fuzziness within classical (non-fuzzy) logic (Schefe 1980, Bibel 1983a). Zadeh (1975) developed a unified approach for a fuzzy logic, maintaining that intrinsic fuzziness cannot be captured adequately within a classical logic system. In his approach, classical logic becomes a degenerate special case of fuzzy logic. Zadeh's work stimulated a lot of research into the same (e.g. Goguen 1974) or into alternate directions.

The work mentioned above focusses on formal aspects of representation. In real world applications, the correspondence between fuzzy concepts in natural language and its representation is an additional issue (Freksa 1982, Freksa and Lopez de Mantaras 1982). A qualitative relational approach to representing fuzzy linguistic concepts has been proposed by Freksa (1981).

The nature of various types of fuzziness has not yet been sufficiently clarified to evaluate the various approaches conclusively (Shapiro 1983a). An example of the debate can be found in the sequence of papers by Kandel and Byatt (1978), Tribus (1979), and Zadeh (1980).

Based on the fuzzy set approach, Zadeh (1975, 1977) developed a theory of approximate reasoning. Collins (1978) stresses the importance of meta-knowledge in plausible reasoning. Access to the state of the art in approximate and plausible reasoning can be obtained through Prade's (1983) tutorial paper. Quinlan (1983) proposes a system in which a plausible reasoning process divides an "inconsistent" knowledge base (cf. section 2.1.3) into internally consistent subsets. Rather than weighing the validity of individual propositions, relative validities of the subset divisions are considered. For example, the greatest consistent subset could be considered the most plausible choice.

2.1.3 Dealing with inconsistencies

Inconsistency in knowledge bases can occur from a variety of sources. The knowledge base may be directly supplied with conflicting information or the system itself may generate conflicting assertions. Inconsistencies may also arise because of failure to represent information-distinguishing parameters such as time. There exists an obvious necessity for recognition and proper handling of such inconsistencies.

The case of conflicts due to the user occurs frequently in building up a knowledge base. If the application requires this knowledge base to be consistent, the user must be supplied with enough information to allow correction of conflicts. It can, however, be useful to include conflicting information in the system as in the case of representing expertise from several experts (Reboh 1983). Quinlan (1983) describes a system which defines internally consistent subsets of knowledge representing several possible worlds.

The case of system-generated inconsistency is a common problem in belief systems working with incomplete knowledge in which decisions are based on default information rather than specific knowledge. These inconsistencies may be due to conflicting assumptions in choosing default values or due to updating of the data base.

Doyle (1979b) describes a "truth maintenance system" to be used in conjunction with such default reasoning. Stallman and Sussman (1977) describe a system for circuit analysis which reasons based on constraint propagation. This system remedies inconsistencies through dependency-directed backtracking. The capability to locate sources of inconsistency requires a large overhead in the form of justifications for beliefs and deductions but this information is required for the explanation component which exposes the system's reasoning process to the user.

2.1.4 Representing evidence and reasoning in belief systems

Knowledge is not absolute. Rather, it depends on the a priori disposition of the 'knower' and on the inference and updating methods he uses for forming a belief. Shafer (1976) developed a theory of evidence which acted as catalyst for research in this area (cf. Prade 1983). Theories of evidence are influenced by their underlying model of vague and approximate knowledge (Wahlster 1981). A much considered method for representing attitudes (like beliefs) towards propositions is modal logic with possible world semantics (Kripke 1959). Problems with this approach are discussed by Moore (1977) and Zadeh (1978).

2.1.5 Representing time and temporal logics

Time is a necessary component of knowledge representation schemes designed to model dynamic events. Indeed without reference to time, such phenomena as growth, motion, etc. cannot be characterized and would, in fact, lead to contradictory statements: "John weighs 3600 g"; "John weighs 70 kg". Hence, the introduction of temporal information eliminates a possible cause of inconsistency in a knowledge base.

The various aspects of time which must be dealt with are: events which have a beginning and ending time, objects which change during their lifetimes, and knowledge which arrives to a system at different times (Long and Russ 1983). There has been great motivation in the areas of linguistics and natural language understanding toward the development of formal representations of time (Crystal 1966, Bull 1962, Bruce 1972, Kandrashina 1983). Problem solving systems have approached the problem using state spaces. The inadequacy of this approach lead to attempts to incorporate time explicitely in planning (Hendrix 1973, Wesson 1977). The logic school has introduced the use of modal and special temporal logics.

Prior (1957, 1967, 1968) introduces tense operators to propositional logic resulting in a "tense logic". McCarthy and Hayes (1969) develop a "situation calculus". An approach aiming at generality is that of the time specialist. This is a subsystem which maintains temporal relations and provides the rest of the system with tools to store, retrieve, delete, and reason with temporal knowledge. More recent work in the area is exemplified by the systems of Allen (1981), Bruce (1972), Kahn and Gorry (1977), and Malik and Binford (1983). The common weakness of all attempts to date is their lack of ability to deal with the full dimensions of temporal logic, as was mentioned earlier.

2.1.6 Representing spatial knowledge and spatial reasoning

In applications which describe manipulation of physical objects or navigation in a physical environment, a system must be capable of representing and reasoning with spatial knowledge (Kuipers 1977). An overview of representation methods for spatial knowledge can be found in the Proceedings of the workshop on representing and processing spatial knowledge, held May 5, 1983, at the University of Maryland (Report Nr. TR-1275, Dept. of Computer Science, Univ. of Maryland, College Park, MD).

2.1.7 Reasoning accounting for different points of view

As a consequence of knowledge being acquired by different means or being used with different objectives, it may represent different points of view. An example for a reasoning system accounting for different points of view is the HAM-RPM/HAM-ANS system (Jameson et al. 1981).

Moore (1977) uses Kripke's possible world model for modal logic to represent different viewpoints. McCarthy (1979) develops first-order theories for the formalization of individual concepts. Applications are the representation of individual knowledge, belief, wanting, and necessity without modal operators.

In natural language understanding, accounting for different points of view is essential (Schank and Abelson 1977). Certain difficulties arise if one reasons based on knowledge of other agents. A distinction has to be made between knowing something and knowing that someone knows something. Approaches to such problems are given by Bibel (1983b). This has important applications for distributed data bases (if they are considered as reasoning agents).

2.2 User-related aspects

In developing representation systems, the necessity to include the user as part of the system has become evident. The present section deals with facilities to support user participation.

2.2.1 Explanation facility

An explanation facility is that part of an interactive computer system whose task is to generate for each particular dialogue situation an explanation that makes sense to the user. Such explanations may refer to both, the object-level and the meta-level of a reasoning system (Wahlster 1981). Through studies of user interaction with expert systems it has been found that acceptance of the systems is highly dependent on adequate explanation facilities. Such facilities also make the system useful for instructional purposes (Clancey 1983).

Implementation of an explanation facility requires the system to have 'self knowledge' (Mylopoulos and Levesque 1983). Most experience in design and development of explanation facilities has been obtained in the area of expert systems (cf. Barr and Feigenbaum 1982).

2.2.2 User model

The interaction between a computer system and its user has been dominated by technical considerations about the computer system, in the past. There is a trend to focus more on cognitive aspects of the user. For example, at the University of California, San Diego, a project UCSD ('User Centered System Design') is being carried out (cf. Dirlich 1983). Such research may lead to the development and implementation of a user model which enables natural and individually optimized human - machine interaction (Norman 1983).

2.2.3 Programming environment

Due to the complexity of artificial intelligence programming, a knowledge engineer requires powerful tools. Such tools include language-oriented editors, tracing and debugging facilities, knowledge base consistency checkers, multiple window-process monitoring facilities, etc. Examples for such systems are the INTERLISP environment (Teitelman and Masinter 1983) and the LISP-Machine. Current PROLOG implementations offer only minimal programming environments. Shapiro (1983b) advocates implementation of portable programming tools written in PROLOG.

2.3 Control aspects

In a knowledge representation system, we can distinguish three levels: 1) knowledge expressed in a particular formalism; 2) a set of applicable inference rules allowing for the derivation of not explicitly stated knowldege; 3) a control component describing in which order the inference rules are applied. The subject of this section is the interrelationship between levels 1) and 3). Confronted with a certain problem, the system attempts to derive a solution as efficiently as possible, e.g. by avoiding derivation of redundant or otherwise useless knowledge. Two related topics dominate the rather controversial discussions on control: programming vs. theorem proving; the interrelationship between knowledge and control.

2.3.1 Characteristics of programming and theorem proving

We can conceptualize a continuous space between the two extremes of programming and theorem proving. Theorem proving (Loveland 1978) has the following typical characteristics:

- the problem formulation consists only of knowledge;

- the control is hidden from the user;

- usually, the inference mechanism is complex and unintelligable to the user.

As a result, the behavior of a theorem prover in solving a given problem is rather unpredictable. Sufficient knowledge cannot guarantee efficient behavior. To take a rudimentary example, knowledge about 'permutation' and 'ordering' will not suffice for a theorem prover to solve sorting problems intelligently. To obtain efficiency, the formulation must be transformed into a better behaving sorting axiomatization ('program transformation').

In contrast, programming is typified by the following characteristics:

- a careful formulation of relevant knowledge and control;

- problem formulation with the intention of solving a predicted set of problems with predictable efficiency;

- a simple inference mechanism with visible control, i.e., the user is in control of the behavior (a procedural semantic exists).

The borderline between programming and theorem proving is moving. Our understanding of the nature of the ongoing process is increasing. Slogans as 'computation is controlled deduction' (Hayes 1973a) and 'algorithm = logic + control' (Bibel 1975, Kowalski 1979a) illustrate this trend. The use of Horn clauses with a specialized theorem prover is now considered as programming (Kowalski 1974). As a result of the better understanding, the level of programming languages is increasing. All current approaches to expert systems can be classified on the programming side of our space. This classification is not restricted to logic-based systems but also holds for other formalisms such as production systems, semantic nets, etc.

2.3.2 The relationship between knowledge and control

The amount of separation between knowledge and control also can be viewed as a continuous space. Knowledge and control can be completely separated as formulated by Lasserre and Gallaire (1982) where control is considered as metalevel knowledge and can be very general. This tends to be computationally inefficient. Efficiency can be restored by restricting control to well-chosen primitives as in MU-Prolog (Naish 1982). Although this restricts the expressive power, it is important to realize that unrestricted control is not a "philosopher's stone" (Shapiro 1983b). The equation 'algorithm = logic + control' (Bibel 1975, Kowalski 1979a) has its limitations: twiddling with control cannot turn a search for an ordered permutation into quicksort.

Control information may also appear as an annotation of the knowledge, without affecting the content of the knowledge. A typical example is IC-Prolog (Clark and McCabe 1979). Prolog also, if carefully used, falls into this class. Typical is its dual semantics: on one hand, declarative semantics allows each fragment of the problem description to be a meaningful piece of knowledge; on the other hand, procedural semantics defines how the system behaves in solving a particular problem.

At the other extreme, knowledge and control is intertwined. This is a powerful and efficient approach but has severe drawbacks. The fragments of the problem statement can only be understood in the context of the whole. Reading, modifying, and maintaining programs is much harder. The location of particular formalisms between the two extremes is not absolute. A formalism can be carefully used or abused.

The kind of control which can be efficiently exercised also heavily depends on implementation aspects, especially on chosen data structures. This is the case with semantic nets and frames (Hayes 1979, Reiter 1978a, Schubert et al. 1983, Brachmann et al. 1983). As a final remark, we point to the work of Weyhrauch (1980), Bowen and Kowalski (1982), Davis (1980a), Smith (1982), which discuss the relationship between meta-level and object-level knowledge.

2.4 Reasoning methods

Various representation formalisms favor certain reasoning approaches. The present section gives a brief overview over logic-oriented deductive reasoning methods.

2.4.1 Reasoning based on simple implications

Current approaches to knowledge representation in expert systems are dominated by so-called rule-based systems, whose 'rules' are implications of the form

$$A \Leftarrow B_1, B_2, \ldots, B_n.$$

Best known examples are Horn clauses (Prolog, where A, B_i denote predicates) and production rules, where they denote actions and conditions. The limitation of knowledge to such rules simplifies the required reasoning. Simple forward or backward chaining is sufficient. In the case of Horn clauses, where the resolution method is used, expressive power may be enhanced by use of the 'negation as failure' convention (Clark 1978) and by allowing for the possibility of using variables as calls; this approach permits the use of higher-order predicates.

2.4.2 Resolution-based reasoning

Extending from Horn clause to full first-order logic requires complex resolution-based systems (Robinson 1965) which are driven by the goal of deriving the empty clause. Control is complex and hidden from the user. Some systems, most notably graph-based systems, intertwine knowledge, control, and data structures, e.g. connection graphs.

2.4.3 Non-resolution-based reasoning

In all deductive reasoning methods, including those discussed in the present section, the given formulas (representing knowledge, assertions, etc.) are processed by combining their parts into (sometimes vast amounts of) newly generated parts. In contrast to this redundant approach the connection method in automated theorem proving performs the same processes on a single copy of the given formulas. A concise introduction may be found in Bibel (1983a), while Bibel (1982) provides a comprehensive treatment.

This method also may be regarded as a link between resolution (section 2.4.2) and natural deduction (section 2.4.4) since a natural encoding may be provided for both. Further, the approach for solving deductive problems by rewriting and generalized unification, treated extensively in the literature, can be easily incorporated into the connection method as a (deterministic) control for the actions of the connection prover (cf. section V.4 Bibel 1982). Similar remarks hold with respect to a special handling of equality, induction, higher-order features, and others.

2.4.4 Natural deduction inference rules

In mathematical logic, calculi for natural deduction have been originated by G. Gentzen and further studied by D. Prawitz (among many others). They are meant to formalize the natural human way of exact reasoning. In AI there has been a constant interest in these calculi in order to provide a more natural interface for interaction with human users than for instance via resolution. Bledsoe gives a comprehensive overview of such approaches. A recent work in this direction is Pellentier (1982). The naturalness of such calculi may be combined with the efficiency of more technical calculi by specifying a transformation from one into the other and vice versa. Such a transformation is given in chapter 4 of Bibel (1982).

3 LANGUAGES

In this section, we point to several implemented representation languages which exemplify important developments for knowledge representation. The section is structured into logic-based, semantic net - based, procedural, and frame-based languages, followed by expert system development environments. All systems considered are based on symbolic rather than arithmetic computation; they implement list processing as a basic function; they utilize recursion as a way of describing processes; and they are interpretative and interactive.

3.1 Logic-based languages

ABSYS ('ABerdeen SYStem'; Foster and Elcock 1969, Elcock 1983) is an early assertative programming system which contains features that were later included in the popular PROLOG language.

PROLOG was developed in 1973 by Colmerauer and Roussel (Colmerauer et al. 1973) as a theorem prover based on the resolution principle (Robinson 1965). This language represents knowledge in the form of IF - THEN rules and implements a depth-first control strategy with automatic backtracking. Horn-clause logic can be directly represented in the language and does not require additional constructs. The system attempts to prove goal clauses by means of pattern matching. Control in present-day PROLOG systems is limited to execution of the clauses in static order. Developments for an enrichment of the control facilities are in progress, e.g. in IC-PROLOG (Clark and McCabe 1979), in PROLOG II (Colmerauer 1983), MU-PROLOG (Naish 1982).

PROLOG is being used extensively for expert system development and it is the target language for the Japanese Fifth Generation Computer Project.

FOL In section 1.1, FOL (Weyhrauch 1979) has already been mentioned as a knowledge representation formalism based on full first-order logic. Since there is no restriction on the formulas, FOL is considerably richer in expressiveness than PROLOG. FOL has a flexible declaration mechanism so that the user can define his own language constructs like sorts, etc.

A feature in FOL of particular interest is its notion of a simulation structure, intended to be the mechanizable analogue of the logical notion of model. It allows the integration of semantic computations within formalized reasoning ("semantic attachment"). It also provides a neat way to incorporate efficient procedures (e.g. LISP functions) into the logical formalism.

LOGLISP
POPLOG
LISLOG These three languages are examples of recent attempts to integrate logic and functional programming, represented by Prolog and LISP, respectively.

3.2 Semantic net based languages

SNePs ('Semantic Network Processing System'). This system was one of the first semantic networks to have the expressive power of first-order logic (Shapiro 1979b).

KL-ONE is a knowledge representation language developed by Brachman (1977b) that grew out of semantic network formalisms. The primary unit of information in KL-ONE is called a 'concept', which denotes a set of objects. Unlike early semantic networks, domain-dependent relations are not represented as links, but as concepts and 'rolesets'. A roleset is analogous to a slot in a frame-like language. There is only a small number of types of links in KL-ONE, each corresponding to a concept-forming or roleset-forming operator.

KRYPTON (Brachman et al. 1983, Brachman and Levesque 1983) has two representation languages, one for forming descriptive terms and one for making statements about the world using these terms. The latter was not present in the KL-ONE system. Further, KRYPTON provides a functional view of a knowledge base, characterized in terms of what it can be asked or told, not in terms of the particular structures it uses to represent knowledge. KRYPTON is now being implemented in INTERLISP-D.

PSN ('Procedural Semantic Network') has been developed by Mylopoulos et al. (1983). This system aims at integrating semantic network and procedural notions. With every generic object PSN associates programs which specify how to operate on instances of that object.

OMEGA is designed to facilitate common sense reasoning (Attardi and Simi 1982). Its mode of expression include descriptions, inheritance, quantification, negation, attributions, and multiple viewpoints. Attributions and inheritance are the basic mechanisms for knowledge structuring.

CSRL ('Conceptual Structures Representation Language'; Bylander et al. 1983) is designed to facilitate the development of expert diagnosis systems based on the paradigm of cooperating diagnostics specialists. CSRL is a language for representing the concepts of a diagnostic hierarchy and for implementing the establish - refine process. The knowledge to establish or reject a concept is factored into knowledge groups which correspond to specific decisions in the diagnosis.

3.3 Procedural languages

PLANNER was developed by Hewitt (1969) for goal-directed reasoning in problem solving. It implements full automatic backtracking: the PLANNER run-time environment takes a goal to be achieved and a collection of theorems and attempts to find a theorem to achieve the goal; the search strategy is depth-first. Procedures are activated by pattern-directed procedure invocation. An important program written in (MICRO-) PLANNER is Winograd's (1972) SHRDLU.

CONNIVER was developed as an answer to PLANNER's automated control structure (Sussman and McDermott 1972): sometimes the automatic backtracking resulted in the worst algorithms for solving a problem. Therefore, in CONNIVER control was handed to the programmer, allowing any search strategy to be pursued. While PLANNER forgot sub-problem solutions that eventually lead to a failure, CONNIVER allows access of such solutions from a different branch in the problem tree.

FUZZY was developed by LeFaivre (1974) to integrate the successful aspects of PLANNER and CONNIVER in a clean and efficient implementation and to provide tools for handling fuzzy knowledge, degrees of certainty, statistical information, or the like. These tools are fully integrated into the system's control structure.

FUZZY is implemented in LISP and contains the full LISP as subset. It employs a powerful pattern matcher and allows for automatic and programmer-controlled evaluation. Global control is available via procedure demons which supervise the evaluation process. For example, when a low degree of belief is associated with a computed result, a demon may suspend the actual line of reasoning and pursue another possibility.

FUZZY survived PLANNER and CONNIVER, most notably in the large natural language reasoning system HAM-ANS (Hoeppner et al.

1983). This is partly due to the transparency of the language and due to the ease of implementing a wide spectrum of mechanisms.

3.4 Frame-based languages

FRL ('Frame Representation Language'; Goldstein and Roberts 1977). An FRL knowledge base consists of frames whose slots carry information such as comments on the source of a value bound to a slot, a default value, constraints, and procedures that are activated when a value is bound, unbound, or needed for a slot. All frames are organized into a hierarchy which appears to be a combination of classification and generalization. The procedures attached to a slot are expressed in LISP.

KRL ('Knowledge Representation Language'; Bobrow and Winograd 1977). This is a more ambitious project than FRL. Like FRL, the basic units of a KRL knowledge base are frames with slots and several kinds of information attached to each slot. Unlike FRL, where this information provides details about how to instantiate a frame, KRL is much more concerned with a matching operation for frames. All on-going processes at any one time are controlled through a multiprocessor agenda which can be scheduled by the designer of the knowledge base. KRL also supports belief contexts which can serve to define an attention focusing mechanism. "Self knowledge" can be included in a knowledge base by providing descriptions about other descriptions.

OWL Unlike other frame-oriented schemes, OWL (Szolowitz et al. 1977) bases its features on the syntactic and semantic structure of English, taking as founding principle the Whorfian Hypothesis that a person's language plays a key role in determining his model of the world and thus in structuring his

thought. An OWL knowledge base can be viewed as a semantic network whose nodes are expressions representing the meaning of natural language sentences. Each node, called a concept, is defined by a pair (genus, specializer) where 'genus' specifies the type or superconcept while 'specializer' serves to distinguish this concept from all other concepts with the same genus.

AIMDS (Sridharan 1978) is a frame representation language built on top of FUZZY. In AIMDS, relations which are used as attributes of frames can be characterized by properties like reflexivity, symmetry, and transitivity.

HPRL ('Heuristic Programming and Representation Language'; Rosenberg 1983) is an extension to FRL. It provides for uniform encoding of reasoning knowledge as for domain knowledge in a declarative representation. The language contains the ability to do forward and backward chaining, as well as to use meta-rules.

3.5 Expert system shell languages

AGE ('Attempt to GEneralize'; Nii and Aiello 1979) is an attempt to formulate the knowledge that knowledge engineers use in constructing knowledge-based programs and put it at the disposal of others in the form of a software laboratory. The task for AGE is divided into two main sub-tasks: (1) isolating techniques used in knowledge-based systems and programming those that are general and useful, and (2) building an intelligent agent to guide in the use of these techniques.

APES ('A Prolog Expert system Shell'; Hammond 1982) is a suite of PROLOG programs which can be used as a shell around which other modules can be added to produce a domain-specific rule-based expert system which can explain its reasoning. Missing information for the reasoning process can be supplied by the user interactively by the user through a so-called query-the-user facility. APES has been implemented on a microprocessor system.

EMYCIN ('Empty MYCIN'; van Melle 1980) consists of the basic control structure of the MYCIN expert system, but with MYCIN's infectious-disease knowledge base removed. It uses the production system formalism to represent knowledge.

BIBLIOGRAPHY

This bibliography is coded according to the following key of topics:

DB - Data Bases
EX - Expert systems
FR - Frames
FZ - Fuzzy knowledge & approximate reasoning
GC - General and classic
LO - Logic
LP - Logic programming
ML - Meta-level structures & control
Ot - Others
PR - Procedural representations and production systems
SN - Semantic nets
Ti - Representation of Time
TP - Theorem proving

EX Aiello,N. 1983. A comparative study of control stratagies for expert systems: age implementation of three variations of PUFF, AAAI-83, pp 1-4.

LP Allen,B.P.,Wright,J.M. 1983. Integrating logic programs and schemata, IJCAI-83, pp 340-347.

Ti Allen,J.F. 1981. An interval-based representation of temporal knowledge, IJCAI-81, pp 221-226.

GC Amarel,S. 1968. On representations of problems of reasoning about actions. In: Machine Intelligence 3, ed. Michie,D., Edinburgh University Press Edinburgh.

SN Anderson,J.R.,Bower,G.H. 1973. Human Associative Memory, V.H. Winston and Sons, Washington, D.C.

SN Attardi,G.,Simi,M. 1981. Consistency & completeness of OMEGA, a logic for knowledge representation, IJCAI-81, pp 504-510.

SN Attardi,G.,Simi,M. 1982. Semantics of inheritance and attributions in the description of system OMEGA, AI Memo no. 642, AI Lab. MIT.

GC Barr,A. 1979. Meta-knowledge and cognition, IJCAI-79, pp 31-33.

GC Barr,A.,Feigenbaum,E.A. 1982 Handbook of Artificial Intelligence, vol. 2, HeurisTech Press, Stanford.

LP Bibel,W. 1975. Praedikatives Programmieren, GI-2 Fachtagung ueber Automatentheorie und Formale Sprachen, Lecture Notes in Computer Science 33, Springer-Verlag, Berlin, pp 274-283.

TP Bibel,W. 1982. Automated Theorem Proving, Friedr. Vieweg & Sohn, Braunschweig, W. Germany.

LP Bibel,W. 1983a. Knowledge representation from a deductive point of view, Technical Report ATP-19-V-83, May 1983, Technical University Munich, also to appear Proc. IFAC Symposium on Artificial Intelligence, Leningrad, USSR, Pergamon Press.

LO Bibel,W. 1983b. First-order reasoning about knowledge and belief, Technical Report ATP-21-IX-83, Sept. 1983, TU Munich.

LO Bibel,W. 1983c. Matings in Matrices, CACM 26(11), pp 844-852.

LP Bibel,W.,Hoernig,K.M. 1983. LOPS - A system fased on a strategical approach to program synthesis. In: Automatic Program Construction Techniques, ed. Biermann,A.,et al., McMillan, New York.

TP Bledsoe,W.W. 1977. Non-resolution theorem proving, Artificial Intelligence 9(1), pp 1-35.

GC Bobrow,D.G. 1975. Dimensions of representation. In: Representation and Understanding: Studies in Cognitive Science, ed. Bobrow,D.,Collins,A., Academic Press, New York.

GC Bobrow,D.G. 1977. A panel on knowledge representation, IJCAI-77, pp 983-992.

GC Bobrow,D.G.,Collins,A.M.(eds) 1975. Representation and Understanding: Studies in Cognitive Science, Academic Press, New York.

GC Bobrow,D.G.,Norman,D.A. 1975. Some principles of memory schemata. In: Representation and Understanding: Studies in Cognitive Science, ed. Bobrow,D.,Collins,A., Academic Press, New York.

FR Bobrow,D.G.,Winograd,T. and the KRL Research Group 1977. Experience with KRL-0: one cycle of a knowledge representation language, IJCAI-77, pp 213-222.

ML Bowen,K.A., Kowalski,R.A. 1982. Amalgamating language & meta-language. In: Logic Programming, ed. Clark,Tarnlund, Academic Press, London, pp 153-172.

SN Brachman,R.J. 1977a. What's in a concept: Structural foundations for semantic networks, Int. J. Man-Machine Studies 9, pp 127-152.

SN Brachman,R.J. 1977b. A structural paradigm for representing knowledge, Ph.D. thesis, Harvard Univ., also BBN Report no. 3605, Bolt, Beranek and Newman Inc, May 1978.

SN Brachman,R.J. 1979. On the epistemological status of semantic networks. In: Associative Networks: Representation and Use of Knowledge by Computers, ed. Findler,N.V., Academic Press, New York, pp 3-50.

SN Brachman,R.J. 1983. What isa is and isn't: an analysis of taxonomic links in semantic networks, IEEE Computer, Special Issue on Knowledge Representation, October, pp 30-36.

GC Brachman,R.J.,Smith,B.C.(eds) 1980. Special Issue on Knowledge Representation, SIGART Newsletter no. 70, Feb.

SN Brachman,R.J.,Fikes,R.E.,Levesque,H.J. 1983. KRYPTON: A functional approach to knowledge representation, IEEE Computer, pp 67-73.

SN Brachman,R.J.,Levesque,H.J. 1983. KRYPTON: Integrating terminology and assertion, AAAI-83, pp 31-35.

DB Brodie,M.L., Zilles,S.N.(ed.) 1980. Proceedings of the Workshop on Data Abstraction, Data Bases & Conceptual Modelling, ACM Sigart Newsletter no. 74.

Ti Bruce,B. 1972. A model for temporal references and its application in a question answering program, Artificial Intelligence 3, pp 1-25.

LP Bruynooghe,M. 1981. Solving combinatorial search problems by intelligent backtracking, Information Processing Letters 12(1), pp 36-39.

LP Bruynooghe,M. 1981. Intelligent backtracking for an interpreter of Horn clause logic programs. In: Mathematical Logic in Computer Science, ed. Domolki,B., Gergely,T., North Holland, pp 215-257.

LP Bruynooghe,M.,Pereira,L.M. 1981. Revision of top-down logical reasoning through intelligent backtracking, Report KUL/CIUNL, Katholike Univ. Leuven.

LP Bruynooghe,M.,Pereira,L.M. 1984. Deduction revision by intelligent backtracking. To appear in: Implementation issues of PROLOG, ed. Campbell, Ellis Horwood.

EX Buchanan,B.G. 1982. New research on expert systems. In: Machine Intelligence 10, ed. Hayes,J.E., Michie,D.,Pao,Y-H., Wiley & Sons, New York, pp 269-299.

Ti Bull,W. 1962. Time, tense, and the verb, Univ of California Publications in Linguistics 19.

ML Bundy,A.,Byrd,L.,Luger,G.,Mellish,C.,Palmer,M. 1979. Solving mechanics problems using meta-level inference, IJCAI-79, pp 1017-1027.

GC Cercone,N.,Goebel,R. 1982. Knowledge representation and databases: science or engineering, Proc. CSCSI/SCEIO Fourth National Conference, Saskatoon, Saskatchewan, pp 172-182.

FZ Collins,A. 1978. Fragments of a theory of human plausible reasoning, TINLAP-2, pp 194-201.

Ot Collins,A., Warnock,E.H., Aiello,N., Miller,M.L. 1975. Reasoning from incomplete knowledge. In: Representation and Understanding, ed. Bobrow,D.G.,Collins,A., Academic Press, New York, pp 383-415.

EX Clancy,W.J. 1983. The epistemology of a rule-based expert system- a framework for explanation, Artificial Intelligence 20, pp 215-251.

LP Clark,K.L. 1978. Negation as failure. In: Logic and Data Bases, ed. Gallaire,H.,Minker,J., Plenum Press, New York, pp 293-324.

LP Clark,K.L.,McCabe,F.G. 1979. The control facilities of IC-PROLOG. In: Expert Systems in the Micro Electronic Age, ed. D.Michie, Edinburgh Univ Press, pp 122-149.

LP Colmerauer,A.,Kanoui,H.,Pasero,R.,Roussel,Ph. 1973. Un systeme de communication homme-machine en francais, Research Report, Group Intelligence Artificielle, Faculte des Sciences de Luminy, Marseille.

LP Colmerauer,A. 1983. Prolog in 10 figures, IJCAI-83, pp 487-499.

Ti Crystal,D. 1966. Specification and english tenses, J. of Linguistics 2, pp 1-33.

LP Dahl,V. 1983. Current trends in logic grammars, Proc. of the Logic Programming Workshop '83, Portugal, pp 578-607.

ML Davis,R. 1977. Generalizing procedure calling & content-directed invocation, Proc. Symposium on Artificial Intelligence & Programming Languages, Rochester,NY., In: Sigplan notices vol 12, 8 & Sigart newsletter no. 64, pp 45-54.

ML Davis,R. 1978. Knowledge about representations as a basis for system construction & maintainance. In: Pattern Directed Inference Systems, ed. Waterman,D.,Hayes-Roth,F., Academic Press.

ML Davis,R. 1980a. Meta-rules: Reasoning about control, Artificial Intelligence 15, pp 179-222.

ML Davis,R. 1980b. Content reference: Reasoning about rules, Artificial Intelligence 15, pp 223-239.

EX Davis,R. 1982. Expert systems: Where are we? and Where do we go from here? The AI Magazine, 3(2), pp 3-22.

ML Davis,R.,Buchanan,B.G. 1977. Meta-level knowledge: overview & applications, IJCAI-77, pp 920-927.

PR Davis,R.,Buchanan,B.G.,Shortliffe,E.H. 1977. Production rules as a representation for a knowledge-based consultation system, Artificial Intelligence 8(1).

PR Davis,R.,King,J. 1977. An overview of production systmes. In: Machine Intelligence 8, ed. Elcock,E.,Michie,D., Ellis Horwood, Chichester, England, pp 300-332.

GC Davis,R.,Lenat,D. 1980. Knowledge-based Systems in Artificial Intelligence, McGraw-Hill, New York.

ML deKleer,J.Doyle,J.,Sussman,G. 1977. AMORD: Explicit control of reasoning, Proc. Symposium on Artificial Intelligence and Programming Languages, SIGPLAN Notices 12(8), and SIGART Newsletter 64, pp 116-125.

ML deKleer,J.,et al. 1979. Explicit control of reasoning. In: Artificial Intelligence: An MIT Perspective, ed. Winston,P.H., Brown, MIT Press, Cambridge,MA.

SN Deliyanni,A.,Kowalski,R.A. 1979. Logic and semantic networks, CACM 22(3), pp 184-192.

ML Dincbas,M. 1980. The METALOG problem solving system, an informal presentation, Proc. of Logic Programming Workshop, Debrecen,Hungary, ed. Tarnlund,S-A, pp 80-91.

Ot Dirlich,G. 1983. Enige kognitive Aspekte fehlertoleranter Mensch-Maschine Schnittstellen, GI Workshop ueber fehlertolerante Systeme, Erlangen.

Ot Doyle,J. 1979a. A truth maintenance system, Artificial Intelligence 12(3), pp 231-272.

Ot Doyle,J. 1979b. A glimpse of truth maintenance, Proc. of the Fourth Workshop on Automated Deduction, Austin, Texas, pp 90-96.

GC Dreyfus,H.L. 1981. From micro-worlds to knowledge representation: AI at an impasse. In: Mind Design, ed. Haugeland,J., MIT Press, Cambridge, MA, pp 161-204.

EX Duda,R.O.,et al. 1979. Model design in the prospector consultant system for mineral exploration. In: Expert Systems in the Micro-electronic Age, ed. Michie,D., Edinburgh Univ. Press, Edinburgh.

LP van Emden,M.H. 1983. Logic as interaction language, Technical note, Dept. of Computing, Imperial College.

SN Fahlman,S.E. 1979. NETL: A System for Representing and Using Real-World Knowledge, MIT Press, Cambridge, MA.

SN Fikes,R.,Hendrix,G. 1977. A network-based knowledge representation and its natural deduction system, IJCAI-77, pp 235-246.

LP Filgueiras,M. 1983. A kernel for a general natural language interface, Proc. of the Logic Programming Workshop, Portugal.

ML Filman,R.E.,Lamping,J.,Montalvo,F.S. 1983. Meta-language and Meta-reasoning, IJCAI-83, pp 365-369.

GC Findler,N.V.(ed) 1979. Associative Networks: Representation and Use of Knowledge by Computers, Academic Press, New York.

PR Forgy,C.,McDermott,J., OPS: A domain-independent production system language, IJCAI-77, pp 933-939.

LP Foster,J.M.,Elcock,E.W. 1969. Absys1: An incremental compiler for assertions: an introduction. In: Machine Intelligence, ed. Melzer,B,Michie,D., Edingurgh Univ. Press, Edinburgh, pp 423-429.

Ot Fox,M.S. 1979. On inheritance in knowledge representation, IJCAI-79, pp 282-284.

FZ Freksa,C. 1981. Linguistic pattern characterization and analysis, Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, Univ. of California, Berkeley.

FZ Freksa,C. 1982. Linguistic description of human judgments in expert systems and in the 'soft' sciences. In: Approximate Reasoning in Decision Analysis, ed. Gupta,M.M.,Sanchez,E., North-Holland.

FZ Freksa,C.,Lopez de Montaras,R. 1982. An adaptive computer system for linguistic categorization of 'soft' observations in expert systems and in the social sciences, 2nd World Conf. on Math. at the Service of Man, Las Palmas.

FZ Gaines,B.R. 1975. Foundations of fuzzy reasoning, Int. J. Man-Machine Studies 6, pp 623-668.

LP Gallaire,H. 1983. Logic data bases vs deductive data bases, Proc. Logic Programming Workshop, Portugal '83, pp 608-622.

DB Gallaire,H.,Minker,J.(eds) 1978. Logic and Data Bases, Plenum Press, New York.

ML Gallaire,H., Lasserre,C. 1982. Metalevel control for logic programs. In: Logic Programming, ed. Clark,Tarnlund, Academic Press, London, pp 173-185.

TP Gelernter,H. 1963. Realization of a geometry-theorem proving machine. In: Computers and Thought, ed. Feigenbaum,E.A.,Feldman,J., McGraw-Hill, New York, pp 134-152.

FZ Goguen,J.A. 1974. Concept representation in natural and artificial languages: axioms, extensions and applications for fuzzy sets, Int. J. Man-Machine Studies 6, pp 513-561.

Ot Goldstein,I.P.,Roberts,R.B. 1977. NUDGE, a knowledge-based scheduling program, IJCAI-5, pp 257-263.

Ot Goldstein,Bobrow. 1980. A layered approach to software design, CSL-80-5, Xerox Palo Alto Research Center.

Ot Goldstein,Bobrow. 1981. An experimental description-based programming environment: Four reports, CSL-81-3, Xerox Palo Alto Research Center.

GC Habel,C. 1983. Logische Systeme und Repraesentations- probleme, GWAI-83, 7th German Workshop on Artificial Intelligence, ed. Neumann,B., Springer-Verlag, Berlin.

EX Hammond,P. 1983. APES: A user manual, Imperial College Dept. of Computing rept. no. 82/9.

LP Hayes,P.J. 1973a. Computation & deduction, Proc. 2nd MFCS Symposium Czekoslovak Academy of Sciences, pp 105-118.

GC Hayes,P.J. 1973b. The frame problem and related problems in artificial intelligence. In: Artificial and Human Thinking, ed. Elithorn,A.,Jones,D., Jossey-Bass, San Francisco.

GC Hayes,P.J. 1974. Some problems and non-problems in representation theory, Proc. AISB Summer Conference, Univ. of Sussex, pp 63-79.

LO Hayes,P.J. 1977. In defence of logic, IJCAI-77, pp 559-565.

Ot Hayes,P.J. 1978. The Naive Physics Manifesto. In: Expert Systems in the Micro-electronic Age, ed. Michie,D., Edinburgh University Press, Edinburgh, pp 242-270.

FR Hayes,P.J. 1979. The logic of frames, In: Frame Conceptions and Text Understanding, ed. Metzing,D., Walter de Gruyter and Co., Berlin, pp 46-61.

EX Hayes-Roth,F. 1983. Building Expert Systems, Addison-Wesley, Reading, MA.

Ti Hendrix,G.G. 1973. Modeling simultaneous actions and continuous processes, Artificial Intelligence 4, pp 145-180.

SN Hendrix,G.G. 1975. Expanding the utility of semantic networks through partitioning, IJCAI-75, pp 115-121.

SN Hendrix,G.G. 1979. Encoding knowledge in partitioned networks. In: Associative Networks: Representation and Use of Knowledge by Computers, ed. Findler,N.V., Academic Press, New York, pp 51-92.

PR Hewitt,C. 1969. PLANNER: A language for proving theorems in robots, IJCAI-69. PR Hewitt,C. 1972. Description and theoretical analysis(using schemata) of PLANNER, Ph.D. thesis, Dept. of Mathematics, MIT, Cambridge.

Ot Hoeppner,W., Christaller,Th., Marburger,H., Morik,K., Nebel,B., O'Leary,M.,Wahlster,W. 1983. Beyond domain independence: experience with the development of a german language access system to highly diverse background systems, Bericht ANS-16, Research Unit for Information Science and Artificial Intelligence, Univ. of Hamburg.

LP Hoernig,K.M. 1980. Generating small models of first order axioms, Proc. of the German Workshop on Automatic Deduction, Springer Verlag.

GC Hofstadter,D. 1979. Goedel, Escher, Bach. Basic Books.

LO Israel,D.J. 1980. What's wrong with non-monotonic logic?, AAAI-80.

Ot Jacob,R.J.K. 1983. Using formal specifications in the design of a human-computer interface, CACM 26(4), April.

Ot Jameson,A.,Hoeppner,W.,Wahlster,W. 1981. The natural language system HAM-RPN as a hotel manager: some representational prerequisites. In: GI-10 Jahrestagung, ed. Nilhelm,R., Springer-Verlag, pp 459-473.

Ti Kahn,K.,Gorry,G.A. 1977. Mechanizing temporal knowledge, Artificial Intelligence 9, pp 87-108.

FZ Kandel,A.,Byatt,W. 1978. Fuzzy sets, fuzzy algebra, fuzzy statistics, Proc. IEEE vol. 66, pp 1619-1639.

Ti Kandrashina,E.Yu. 1983. Representation of temporal knowledge, IJCAI-83, pp 346-348.

DB Kellog,C. 1982. Knowledge management: A practical amalgam of knowledge base and database technology, AAAI-82, pp 306-309.

LP Kornfeld,W.A. 1983. Equality for Prolog, IJCAI-83, pp 514-519.

LP Kowalski,R. 1974. Predicate logic as a programming language, Proc. IFIP Congress, Stockholm, North Holland.

LO Kowalski,R. 1979a. Algorithm = Logic + Control, CACM 22(7), pp 426-436.

LO Kowalski,R. 1979b. Logic for Problem Solving, North-Holland.

LO Kripke,S. 1959. A completeness theorem in modal logic, J. of Symb. Logic 24, pp 1-14.

FR Kuipers,B.J. 1975. A frame for frames: representing knowledge for recognition. In: Representation and Understanding: Studies in Cognitive Science, ed. Bobrow,D.G.,Collins,A.M., Academic Press, New York, pp 151-184.

Ot Kuipers,B. 1977. Modeling spatial knowledge, IJCAI-77, pp 292-298.

Ot Kuipers,B. 1979. On representing commonsense knowledge. In: Associative Networks: Representation and Use of Knowledge by Computers, ed. Findler,N.V., Academic Press, New York.

Ot Levesque,H.J. 1983. The logic of incomplete knowledge bases. In: Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages, ed. Brodie,M.,Mylopoulos,J., Schmidt,J., Springer-Verlag, New York.

LO Lewis,H.R. 1978. Renaming a set of clauses as a horn set, Journal of the ACM, vol. 25, no. 1., Jan 1978, pp 134-135.

LO Linsky,L.(ed) 1971. Reference and Modality, Oxford Univ. Press, London.

Ti Long,W.J.,Russ,T.A. 1983. A control structure for time dependent reasoning, IJCAI-83, pp 230-232.

TP Loveland,D.W. 1978. Automated Theorem Proving: A Logical Basis, North-Holland, New York.

Ti Malik,J.,Binford,T.O. 1983. Reasoning in time and space, IJCAI-83, pp 343-345.

Ot Martin,W.A. 1979. Descriptions and the specialization of concepts. In: Artificial Intelligence: An MIT Prespective, vol. 1, ed. Winston,P.H., Brown,R.H., The MIT Press, Cambridge, MA, pp 375-419.

Ti McArthur 1976. Tense Logic, Reidl Publish. Comp.

LO McCarthy,J. 1958. Programs with Common Sense. In: Mechanization of Thought Processes, Proc. Symp. Nat. Phys. Lab., vol I, pp 77-84, London: Her Majesty's Stationary Office, Also in: Semantic Information Processing, ed. Minsky,M.,1968, MIT Press, Cambridge, MA.

GC McCarthy,J. 1977. Epistemological problems in artificial intelligence, IJCAI-77, pp 1038-1044.

LO McCarthy,J. 1979. First-order theories of individual concepts and propositions. In: Expert Systems in the Micro-electronic Age, ed. Michie,D., Edinburgh University Press, Edinburgh, pp 270-287.

LO McCarthy,J. 1980. Circumscription- A form of non-monotonic reasoning, Artificial Intelligence 13, pp 27-39.

GC McCarthy,J.,Hayes,P.J. 1969. Some philosophical problems from the standpoint of artificial intelligence. In: Machine Intelligence 4, ed. Michie,D.,Melzer,B., Edinburgh Univ. Press, Edinburgh, pp 463-502.

Ti McDermott,D. 1982. A temporal logic for reasoning about processes and plans, Cognitive Science 6(2), pp 101-155.

EX Michalski,R.S., Baskin,A.B. 1983. Integrating multiple knowledge representations and learning capabilities in an expert system: The ADVISE system, IJCAI-83, pp 256-258.

EX Minker,J. 1983. Issues in developing expert systems, Proc. of Logic Programming Workshop '83, Portugal, pp 204-215.

GC Minsky,M.(ed) 1968. Semantic Information Processing, The MIT Press, Cambridge, MA.

FR Minsky,M. 1975. A framework for representing knowledge. In: The Psychology of Computer Vision, ed. Winston,P.H., McGraw-Hill, New York, pp 211-277.

LP Mizoguchi 1983. PROLOG based expert system, New Generation Computing Journal 1, pp 99-104.

Ot Moore,R.C. 1975. Reasoning from incomplete knowledge in a procedural deduction system. AI-TR-347, AI Lab. MIT, Cambridge, Mass.

ML Moore,R.C. 1977. Reasoning about knowledge and action, IJCAI-77, pp 223-227.

LO Moore,R.C. 1982. The role of logic in knowledge representation and commonsense reasoning, AAAI-82, pp 428-433.

GC Mylopoulos,J.,Levesque,H.J. 1983. An overview of knowledge representation. In: Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages, ed. Brodie,M.,Mylopoulos,J., Schmidt,J., Springer-Verlag, New York.

SN Mylopoulos,J.,Shibahara,T.,Tsotsos,J.K. 1983. Building knowledge-based systems: the PSN experience, IEEE Computer 16(10), Oct.

LP Naish,L. 1982. An introduction to MU-Prolog, Technical Report 8212, Dept. of Computer Science, Univ. of Melbourne.

GC Newell,A.,Simon,H.A. 1972. Human Problem Solving, Prentice-Hall.

GC Nilsson,N.J. 1980. Principles of Artificial Intelligence, Tioga Publishing Co., Palo Alto, CA.

Ot Norman,D.A. 1983. Design principles for human-computer interfaces, To be published in: Proc. CHI Conf. on Human Factors in Computer Systems, Boston, Dec.

Ot Palmer,S.E. 1978. Fundamental aspects of cognitive representation. In: Cognition and Categorization, ed. Rosch,E.,Lloyd,B.B., Lawrence Erlbaum, Hillsdale, NJ.

TP Pelletier,J.F. 1982. Completely non-clausel, completely heuristically driven automatic theorem proving, Technical Report TR 82-7, Dept. of Computing Science, Univ. of Alberta, Edmonton.

FZ Prade,H. 1983. Tutorial on fuzzy sets and systems, IFAC Conf., Marseille.

Ti Prior,A.N. 1957. Time and Modality. Oxford Univ. Press, London.

Ti Prior,A.N. 1967. Past, Present, and Future. Oxford Univ. Press, London.

Ti Prior,A.N. 1968. Papers on Time and Tense. Oxford Univ Press, London.

SN Quillian,J.R. 1968. Semantic Memory. In: Semantic Information Processing, ed. Minsky,M., The MIT Press, Cambridge, MA, pp 227-270.

Ot Quinlan,J.R. 1983. Consistency and plausible reasoning, IJCAI-83, pp 137-144.

EX Reboh,R. 1983. Extracting useful advice from conflicting expertise, IJCAI-83, pp 145-150.

GC Reiter,R. 1978a. On Reasoning by Default, Proc. TINLAP-2, Univ. of Illinois at Urbana-Champaign.

DB Reiter,R. 1978b. On closed wold data bases. In: Logic & Data Bases, ed. Gallaire,H.,Minker,J., New York, pp 55-76.

LO Reiter,R. 1980. A logic for default reasoning, Artificial Intelligence 13(1,2), pp 81-132.

LO Reiter,R. 1981. On interacting defaults, IJCAI-81, pp 270-276.

LP Reiter,R. 1983. Towards a logical reconstruction of relational database theory. In: On Conceptual Modelling Perspectives from Artificial Intelligence, Databases & Programming Languages, ed. Brodie, Mylouplos, Schmidt, Springer Verlag.

Ti Rescher,N.,Urguhart,A. 1971. Temporal Logik, Springer-Verlag, Berlin.

Ot Rich,C. 1980. Knowledge representation in languages and predicate calculus: how to have your cake and eat it too, Proc. AAAI-82, pp 193-196.

LO Robinson,J.A. 1965. A machine oriented logic based on the resolution principle, J. ACM 12, pp 23-41.

SN Rumelhart,D.E.,Norman,D.A. 1973. Active semantic networks as a model of human memory, IJCAI-73, pp 450-457.

LO Sandwall,E. 1970. Representing natural language information in predicate calculus. In: Machine Intelligence 5, ed. Melzer,B.,Michie,D., Edinburgh Univ. Press, Edinburgh.

Ot Schank,R.C. 1972 Conceptual dependency: A theory of natural language understanding, Cognitive Psychology 3 (4), pp 552-631.

SN Schank,R.C. 1975. Conceptual Information Processing, North Holland, Amsterdam.

Ot Schank,R.C.,Abelson,R.P. 1977. Scripts, Plans, Goals, and Understanding, Lawerence Erlbaum, Hillsdale, NJ.

FZ Schefe,P. 1980. On foundations of reasoning with uncertain facts and vague concepts, Int. J. Man-Machine Studies 12, pp 35-62.

SN Schmolze,J.G.,Lipkis,T.A. 1983. Classification in the KL-ONE knowledge representation system, IJCAI-83.

SN Schubert,L.K. 1976. Extending the expressive power of semantic networks, Artificial Intelligence 7(2), pp 163-198.

SN Schubert,L.K. 1979. Problems with parts, IJCAI-79, pp 778-784.

SN Schubert,L.K.,Goebel,R.G.,Cercone,N.J. 1979. The structure and organization of a semantic net for comprehension and inference. In: Associative Networks: Representation and Use of Knowledge by Computers, ed. Findler,N.V., Academic Press, New York, pp 121-175.

SN Schubert,L.K.,Papalaskaris,M.A.,Taugher,J. 1983. Determining type, color, and time relationships, IEEE Computer, Oct. 1983.

LP Sergot,M. 1983. Query-the-user facility for logic programming. In: Integrated Interactive Computing Systems, ed. Degano,P.,Sandewall,E., North Holland, Amsterdam, pp 27-41.

Ot Shafer,G. 1976. A Mathematical Theory of Evidence, Princeton University Press, Princeton.

LP Shapiro,E.Y. 1981. An algorithm that infers theories from facts, IJCAI-81, pp 446-451.

LP Shapiro,E.Y. 1983a. Logic programming with uncertainties: a tool for implementing rule-based systems, IJCAI-83, pp 529-532.

LP Shapiro,E.Y. 1983b. Methodology of logic programming, Proc. of Logic Programming Workshop '83, Portugal.

Ot Shapiro,E.Y.,Takeuchi,A. 1983. Object oriented programming in concurrent Prolog, New Generation Computing J. 1, pp 25-48.

LO Shapiro,S.C. 1979a. Numerical quantifiers and their use in reasoning with negative information, IJCAI-79, pp 791-796.

SN Shapiro,S.C. 1979b. The SNePS semantic network processing system. In: Associative Networks: Representation and Use of Knowledge by Computers, ed. Findler,N.V., Academic Press, New York, pp 179-203.

EX Shortliffe,E.H. 1976. Computer-Based Medical Concultations: MYCIN, American Elsevier, New York.

SN Simmons,R.F. 1973. Semantic networks: their computation and use for understanding english sentences. In: Computer Models of Thought and Language, ed. Schank,R.C.,Colby,K.M., W.H. Freeman and Co., San Francisco, pp 63-113.

Ot Sloman,A. 1971. Interactions between philosophy and AI: the role of intuition and non-logical reasoning in intelligence, Artificial Intelligence 2, pp 209-225.

ML Smith,B. 1982. Reflection and semantics in a procedural language, MIT/LCS/TR-272, MIT, Cambridge, Mass.

FR Sridharan,N.S. 1978. AIMDS User Manual- Version 2, Rutgers Univ. Computer Science Tech. Report CBM-TR-89, Rutgers Univ., New Brunswick, NJ.

Ot Stallman,R.M.,Sussman,G.J. 1977. Forward reasoning and depencency-directed backtracking in a system for computer-aided circuit analysis, Artificial Intelligence 9, pp 135-196.

FR Stefik,M. 1979. An examination of a frame-structured representation system, IJCAI-79, pp 845-852.

ML Stefik,M. 1981. Planning and meta-planning, Artificial Intelligence 16(2), pp 141-170.

Ot Stoyan,H.W.(ed) 1983. Tagung Objektorientierte Software- und Hardware- architekturen, Berlin, ACM German Chapter 15.

PR Sussman,G.J.,Steele,G.L. 1980. CONSTRAINTS: A language for expressing almost-hierarchical descriptions, Artificial Intelligence 14(1), pp 1-40.

FR Szolovits,P.,Hawkinson,L.,Martin,W.A. 1977. An overview of OWL, a language for knowledge representation, MIT/LCS/TM-86, MIT Lab. for Computer Science.

FZ Tribus,M. 1979. Comments on fuzzy sets, fuzzy algebra, and fuzzy statistics, Proc. IEEE vol 67, p 1168.

Ti Vilain,M.B. 1982. A system for reasoning about time, AAAI-82, pp 197-201.

Ot Wahlster,W. 1981. Natuerlichsprachliche Argumentation in Dialogsysteme, Springer-Verlag, Berlin.

PR Waterman,D.A.,Hayes-Roth,F. 1978. Pattern-Directed Inference Systems, Academic Press, New York.

Ti Wesson,R.B. 1977. Planning in the world of the air traffic controller, IJCAI-77, pp 473-479.

ML Weyhrauch,R. 1980. Prolegomena to a theory of mechanized formal reasoning, Artificial Intelligence 13, pp 133-170.

PR Winograd,T. 1972. Understanding Natural Language, Academic Press, New York.

GC Winograd,T. 1975. Frame representations and the declarative/procedural controversy. In: Representation and Understanding: Studies in Cognitive Science, ed. Bobrow,D.G.,Collins,A.M., Academic Press, New York, pp 185-210.

GC Winston,P.H.(ed) 1975. The Psychology of Computer Vision, McGraw-Hill, New York.

SN Woods,W.A. 1975. What's in a link. In: Representation and Understanding, ed. Bobrow,D.G.,Collins,A., Academic Press, New York, pp 35-82.

FZ Zadeh,L.A. 1975. Fuzzy logic and approximate reasoning, Synthese 30, pp 407-428.

FZ Zadeh,L.A. 1977. A theory of approximate reasoning, Memo no. UCB/ERL 0177/58, Electronics Research Lab., Univ. of California, Berkeley.

FZ Zadeh,L.A. 1978. PRUF - A meaning representation language for natural languages, Int. J. Man-Machine Studies 10, pp 395-460.

FZ Zadeh,L.A. 1980. Fuzzy sets vs. probability, Proc. IEEE 68(3), p 421.

FZ Zadeh,L.A. 1983. Commonsense knowledge representation based on fuzzy logic, IEEE Computer vol 16(10), pp 61-65.

FORSCHUNGSGRUPPE KÜNSTLICHE INTELLIGENZ (INTELLEKTIK)

Artificial Intelligence Research Group

Institut für Informatik, TUM, Postfach 202420
D-8000 München 2,
Germany
Tel.: +49-89-2105-8163
Telex: tumue d 05-22854
EUROKOM: Wolfgang Bibel TU München
CS-NET : tum@germany.csnet

Papers/Reports 1980 – 1983 (updated in July 1985)

ATP-1-I-80 Bibel, W.:
A theoretical basis for the systematic proof method.
Mathematical Foundations of Computer Science, Proceedings of the 9th Symposium MFCS'80, Rydzyna, Poland, September 1980, z(P. Dembinski, ed.), LNCS **88**, Springer, Berlin 1980, pp. 154-167.

ATP-2-II-80 Bibel, W.:
Syntax-directed, semantics-supported program synthesis.
Artificial Intelligence **14** (1980), pp. 243-261.

ATP-3-IV-80 Bibel, W.:
A strong completeness result for the connection graph proof procedure.

ATP-4-VI-80 Bibel, W., Hörnig, K.M.:
LOPS – a system based on a strategical approach to program synthesis. (cf. ATP-9-IX-81)

ATP-5-XI-80 Hörnig, K.M.:
Generating small models of first-order axioms.
GWAI-81, German Workshop on Artificial Intelligence, Bad Honnef, Germany, January 1981, (J. Siekmann, ed.), Informatik-Fachberichte **47**, Springer, Berlin 1981, pp. 248-255.

ATP-6-XII-80 Bibel, W.:
The complete theoretical basis for the systematic proof method.

ATP-7-I-81 Bibel, W.:
Matings in matrices.
GWAI-81, German Workshop on Artificial Intelligence, Bad Honnef, Germany, January 1981, (J. Siekmann, ed.), Informatik-Fachberichte **47**, Springer, Berlin 1981, pp. 171-187.
and C.ACM 26, November 1983, pp. 844-852.

ATP-8-IX-81 Bibel, W.:
Logical program synthesis.
International Conference on Fifth Generation Computer Systems, Tokio, Japan, Oktober 1981, (T. Moto-oka, ed.), North-Holland, Amsterdam 1982, pp. 226-237.

ATP-9-IX-81 Bibel, W., Hörnig, K.M.:
LOPS – a system based on a strategical approach to program synthesis.
(Extended version of ATP-4-VI-80)
In: Automatic program construction techniques, (A. Biermann, G. Guiho, Y. Kodratoff, eds.), MacMillan, New York 1984, pp. 69-89.

ATP-10-XII-81 Bibel, W.:
Computationally improved versions of Herbrands theorem.
Herbrand Symposium, Marseille, France, July 1981, (J. Stern, ed.), Studies in Logic **107**, North-Holland, Amsterdam 1982, pp. 11-28.

ATP-11-XII-81 Hörnig, K.M.:
Can logical program synthesis cope with real life problems?

ATP-12-XII-81 Müller, A.:
An implementation of a theorem prover based on the connection method.

ATP-13-XII-81 Hörnig, K.M., Bibel, W.:
Improvements of a tautology testing algorithm.
6th Conference on Automated Deduction, New York, USA, June 1982, (D. Loveland, ed.), LNCS **138**, Springer, Berlin 1982, pp. 326-341.

ATP-14-II-82 Hörnig, K.M.:
Aspects of automatic program construction.
European Meeting on Cybernetics and Systems Research, Wien, Austria, April 1982, (R. Trappl, ed.), North-Holland, Amsterdam 1982, pp. 909-915.

ATP-15-II-82 Bibel, W.:
Deduktionsverfahren.
Künstliche Intelligenz, Frühjahrsschule, Teisendorf, Germany, Maerz 1982, (W. Bibel, J. Siekmann, eds.), Informatik-Fachberichte **59**, Springer, Berlin 1982, pp. 99-140.

ATP-16-VIII-82 Bibel, W.:
On the role of deduction in intellectics.
2nd International Conference on Artificial Intelligence and Information Control Systems of Robots, Smolenice, CSSR, Oktober 1982, (I. Plander, ed.), Slowakische Akademie der Wissenschaften, Bratislava 1982, pp. 11-16.

ATP-17-II-83 Eder, E.:
Properties of substitutions and unifications.
GWAI-83, 7th German Workshop on Artificial Intelligence, Dassel/Solling, Germany, September 1983, (B. Neumann, ed.), Informatik-Fachberichte **76**, Springer, Berlin 1983, pp. 197-206.
and : J. Symbolic Computation 1 (1985), pp. 31-46.

ATP-18-II-83 Bibel, W., Eder, E., Fronhöfer, B.:
Towards an advanced implementation of the connection method.
IJCAI-83, 8th International Joint Conference on Artificial Intelligence, Karlsruhe, Germany, August 83, (A. Bundy, ed.), Kaufmann Inc., Los Altos 1983, pp. 920-922.

ATP-19-V-83 Bibel, W.:
Knowledge representation from a deductive point of view.
Artificial Intelligence, Proceedings of the IFAC Symposium, Leningrad, USSR, 4-6 October 1983, (V.M. Ponomaryov, ed.), Pergamon Press, Oxford 1983, pp. 37-48.

ATP-20-IX-83 Fronhöfer, B.:
On refinements of the connection method.
in: Algebra, Combinatorics and Logic in Computer Science, (J.Demetrovics, G.Katona, A.Salomaa, eds.), Colloquia Mathematica Soc. J. Bolyai, Vol.42, North-Holland, Amsterdam 1985. (to appear)

ATP-21-IX-83 Bibel, W.:
First-order reasoning about knowledge and belief.
Third International Conference on Artificial Intelligence and Information-Control Systems of Robots, Smolenice, CSSR, June 1984, (I. Plander, ed.), North Holland, Amsterdam 1984, pp. 9-16.

ATP-22-XII-83 Bibel, W., Bruynooghe, M., Freksa, C., Haddawy, P., Moreau, M.:
An overview of formalisms, issues, languages, and systems for the representation of knowledge.

FORSCHUNGSGRUPPE KÜNSTLICHE INTELLIGENZ (INTELLEKTIK)

Artificial Intelligence Research Group

Institut für Informatik, TUM, Postfach 202420
D-8000 München 2,
Germany
Tel.: +49-89-2105-8163
Telex: tumue d 05-22854
EUROKOM: Wolfgang Bibel TU München
CS-NET : tum@germany.csnet

Papers/Reports since 1984 (compiled July 1985)

ATP-23-I-84 Caferra, R., Eder, E., Fronhöfer, B., Bibel, W.:
Extension of PROLOG through matrix reduction.
ECAI-84, 6th European Conference on Artificial Intelligence, Pisa, Italy, September 1984, (T. O'Shea, ed.), North-Holland, Amsterdam 1984, pp. 101-104.

ATP-24-III-84 Bibel, W.:
Automatische Inferenz.
in: Artificial Intelligence - Eine Einführung, (Retti et al., eds.), Teubner, Stuttgart 1984, pp. 145-167.

ATP-25-III-84 Furbach, U., Dirlich, G., Freksa, C.:
Towards a theory of knowledge representation systems.
AIMSA'84, Artificial Intelligence - Methodology Systems Applications, Varna, Bulgaria, September 1984, (W. Bibel, B. Petkoff, eds.), North-Holland, Amsterdam 1985, pp. 77-84.

ATP-26-III-84 Fronhöfer, B.:
Heuristics for recursion improvement.
ECAI-84, 6th European Conference on Artificial Intelligence, Pisa, Italy, September 1984, (T. O'Shea, ed.), North-Holland, Amsterdam 1984, pp. 577-580.

ATP-27-IV-84 Bibel, W.:
Software - Eine Studie aus der Sicht der Künstlichen Intelligenz.

ATP-28-IV-84 Bibel, W.:
Inferenzmethoden.
Künstliche Intelligenz, Frühjahrsschule, Dassel/Solling, Germany, March 1984, (C. Habel, ed.), Informatik-Fachberichte, Springer, Berlin 1985, pp. 1-47.

ATP-29-V-84 Haddawy, P., Freksa, C.:
A description of knowledge representation features for a design support system.

ATP-30-VII-84 Bibel, W.:
Memoria in Machina.
in: Gedächtnis und Menschsein, (Schubert, ed.), EOS, St. Utilien 1984.

ATP-31-VII-84 Fronhöfer, B., Caferra, R., Jacquet, P.:
Improvement of recursive programs from a logic programming point of view.
GWAI-84, 8th German Workshop on Artificial Intelligence, Wingst, Germany, October 1984, (J. Laubsch, ed.), Informatik-Fachberichte, Springer, Berlin 1984, pp. 229-237.

ATP-32-VII-84 Brena, R., Caferra, R., Fronhöfer, B., Gresse, C., Jacquet, P., Potet, M.:
Program synthesis through problem splitting: a method for subproblem characterization.
AIMSA'84, Artificial Intelligence - Methodology Systems Applications, Varna, Bulgaria, September 1984, (W. Bibel, B. Petkoff, eds.), North-Holland, Amsterdam 1985, pp. 113-120.
and Computers and Artificial Intelligence 5, 1985 (to appear).

ATP-33-VII-84 Eder, E.:
An implementation of a theorem prover based on the connection method.
AIMSA'84, Artificial Intelligence - Methodology Systems Applications, Varna, Bulgaria, September 1984, (W. Bibel, B. Petkoff, eds.), North-Holland, Amsterdam 1985, pp. 121-128.

ATP-34-X-84 Freksa, C., Furbach, U., Dirlich, G.:
Cognition and representation - an overview of knowledge representation issues in cognitive science.
GWAI-84, 8th German Workshop on Artificial Intelligence, Wingst, Germany, October 1984, (J. Laubsch, ed.), Informatik-Fachberichte, Springer, Berlin 1984, pp. 119-144.

ATP-35-X-84 Bibel, W.:
Artificial Intelligence in Europe.
AIMSA'84, Artificial Intelligence - Methodology Systems Applications, Varna, Bulgaria, September 1984, (W. Bibel, B. Petkoff, eds.), North-Holland, Amsterdam 1985, pp. 3-10.

ATP-36-XI-84 Bibel, W., Buchberger, B.:
Towards a Connection Machine for Logical Inference.
FGCS (Future Generations Computer Systems), North - Holland Publishing Company, The Netherlands, 1985

ATP-37-XII-84 Aspetsberger, K.:
Substitution Expressions: Extracting Solutions of Non-Horn Clause Proofs.
EUROCAL '85, Linz, Austria, April 1985, (B. Buchberger, ed.) (to appear)

ATP-38-II-85 Bibel, W.:
A Deductive Solution for Plan Generation.
Duke University, Department of Computer Science, Report CS-1985-7
Proceedings of the workshop on 'Knowledge Base Management Systems', (J. W. Schmidt, C. Thanos, eds.), Chania, Creece, June 1985

ATP-39-IV-85 Bibel, W.:
A Brief Survey of Inference Techniques.
Proceedings of the workshop on 'Knowledge Base Management Systems', (J.W. Schmidt, C. Thanos, eds.), Chania, Greece, June 1985

ATP-40-IV-85 Bibel, W.:
Predicative Programming Revisited.
MMSSSS '85, Berlin, GDR, April 1985, Akademie, Berlin und Springer, Berlin (to appear)

ATP-41-V-85 Bibel, W.:
Automated Inferencing.
Journal of Symbolic Computation, Vol.1, Fasc. 3 (to appear)

ATP-42-V-85 Fronhöfer, B.:
The LOPS-Approach: Towards New Syntheses of Algorithms.
ÖGAI-85, Vienna, Austria, September 1985 (to appear)

ATP-43-V-85 Aspetsberger, K.:
Towards parallel machines for Artificial Intelligence: Realization of the Alice Architecture by the L-components.
ÖGAI-85, Vienna, Austria, September 1985 (to appear)

ATP-44-VI-85 Freksa, C.:
Knowledge representation for interactive aircraft design.
Rüschlikon, April 1985, Switzerland (to appear)

ATP-45-VI-85 Haddawy, P.:
Knowledge Representation in Inductive Concept Learning Systems

ATP-46-VII-85 Bibel, W., Aspetsberger K.:
A bibliography on parallel inference machines.
Journal of Symbolic Computation, Vol.1, Fasc. 1, 1985, pp.115-118