AN ARTIFICIAL INTELLIGENCE APPLICATION IN THE LAW: CCLIPS, A COMPUTER PROGRAM THAT PROCESSES LEGAL INFORMATION

BY CARY G. DEBESONET † AND GEORGE R. CROSS ‡

INTRODUCTION

Technology has made a significant impact on the legal profession. Most lawyers know, for example, that commercial systems can retrieve cases and statutes from large databases by searching for key words supplied by the user.¹ In addition, activities such as client interviews, document preparation, and case file management can now be done by computer.

The legal profession cannot afford to ignore the potential that technology has to offer. Manifestations of technology will be finding their way into the practice of law in the not-too-distant future. Researchers are in the process of creating systems to perform conceptual retrieval tasks and to assist lawyers in arguing their cases. The hope is that these systems will be able to find relevant cases and statutes and suggest legal strategies based on the retrieved materials. To accomplish these goals, these systems will have to possess human-like reasoning capabilities.²

The general goal of artificial intelligence ("AI") research is to enable computers to interact with the world as ordinary humans do.³ Humans perceive, reason, communicate, and engage in other intelligent activities

© 1987 High Technology Law Journal

† Professor of Law, Southern University; Director, Law and AI Project, Louisiana State Law Institute, Paul M. Herbert Law Center. CSNET: debesson@lsu.
‡ Assistant Professor of Computer Science, Department of Computer Science, Washington State University. CSNET: cross@wsu.

The authors would like to express their appreciation to Barry Sullivan and Sean Larouque-Doherty for their assistance. The authors would also like to thank the administration and staffs of the Louisiana State Law Institute's "Principle Sponsor" and the Center of Civil Law Studies at the Louisiana State University, Paul M. Herbert Law Center, for their support of the research described in this Article.

to interact with their environments. AI researchers are attempting to create enhanced simulations of these human activities. Researchers have already done quite a bit of work in the areas of pattern and speech recognition and speech production. One can expect to be able to feed information into a computer by oral means and have it respond in a like manner in the near future. Other researchers are modeling computer processes after human cognitive processes such as remembering and daydreaming. Even the cherished reasoning ability of human beings is being automated through diverse methods. In addition, in a field known as "automatic programming," computers are being taught how to program themselves.

Being cognizant of the goals being pursued by AI researchers, it is quite natural for lawyers to speculate about how the new technology might affect the legal profession. This Article describes an AI application in law to give members of the legal profession a view of the AI field and a better perspective on how they might participate in it.

This Article discusses The Civil Code Legal Information Processing System ("CCLIPS"). CCLIPS is designed to retrieve relevant cases and statutes from a highly integrated and efficient data base containing, among other things, portions of the Civil Code of Louisiana. In the near future, CCLIPS will digest facts presented to it, retrieve relevant cases and statutes, and determine legal effects that follow from the facts.

Section I of this Article introduces the concepts of knowledge representation and natural language parsing along with some other basic principles of AI. Section II describes the operations of CCLIPS, including the problems it faces. Section III describes the notational conventions used in CCLIPS to represent variables. Section IV describes the representation of facts and law, including the ways in which causality, goals, states of arousal, mental acts, and temporal schemes are represented in CCLIPS. Section V discusses some applications of AI techniques in the legal domain, and Section VI predicts the future of CCLIPS. The Appendices present technical information to support the

4. See id. at 23-28.
5. See generally id. at 15-36. For a description of this "cognitive modeling process," see infra text accompanying notes 69-71.
6. See, e.g., P. JACKSON, INTRODUCTION TO EXPERT SYSTEMS 210 (1986).
7. CCLIPS is in an intermediate stage of development in which an original prototype is being enhanced to accommodate a new version of the programming language in which it is written. The inferencing powers of CCLIPS are also being enhanced to allow it to perform reasoning and other intelligent operations described in this Article. In addition, this Article describes some of the AI techniques used in CCLIPS to handle causality. Other aspects of CCLIPS, such as the way it handles property inheritance, quantification, denotative reference, and inferencing, will be described in future articles.
I. PRINCIPLES OF ARTIFICIAL INTELLIGENCE

The aim of this section is to give the reader a general understanding of some important AI principles. Subsection A describes how knowledge can be represented for computers. Subsection B describes problems associated with automated language understanding and suggests ways to avoid some of these problems.

A. Knowledge Representation

Computers must have knowledge available to them to be able to engage in activities similar to human intelligence. A good portion of AI research and development is devoted to making knowledge available to the computer in a form it can understand. The field of "knowledge representation" deals with the problem of representing knowledge for the computer. In AI terms, a knowledge representation is a set of conventions describing something, whether it be an object, state, event, rule, procedure, or abstraction.

One of the most popular representation schemes used in AI is based on the notion that information can be packaged into conceptual units, sometimes called "frames." Frames contain slot-like components that bear role descriptions about the described object. The units can be cast into hierarchies to capture relationships between units. For example, a frame for a particular kind of owner would be placed under a frame for owners in general. A frame for owner might contain slots for the following: (1) the name of the owner; and (2) a description of the owned object.

There are many types of representation schemes that follow the general pattern described above. One example, "semantic nets," uses "nodes" (knowledge) as conceptual units. In this scheme the relations between units are represented by "arcs" that connect the units. The IS-A or A-KIND-OF arc, is often used to represent the hierarchical nature of a given relationship. The representation

```
OWNER
IS-A
OWNER OF LAND
```

indicates that an owner of land is a kind of owner because of the presence of the IS-A arc connecting the nodes. By arranging things in this way, it is easier to write programs to deal with matters such as property
inheritance by which subtypes inherit the properties of general types. Some software packages have property inheritance mechanisms built into them.

Once a knowledge structure has been developed to receive information, it can be "instantiated," that is, filled in with concrete information. When the same structure receives different contents on multiple occasions, each different instantiation may be considered an "instance" of whatever the structure represents. A program designed to operate on a knowledge structure independent of its contents can go about the business of manipulating knowledge without receiving new instructions. The knowledge structures function like data abstractions based on the particular needs of the user.

Manipulating knowledge structures to derive results in accord with what one might wish to infer is called "inferencing." Programming languages which have certain inferencing capabilities already built into them are called "logic programming languages." These languages do not require the programmer to write certain types of inferencing programs.

Computers have long been able to perform analytical operations on numbers, symbols, and structures. For example, if a computer is given the following rules in its vernacular, (Rule 1) if A, then B, and (Rule 2) if B, then C, it can deduce C from A. In this AI technique, called "forward chaining," the computer matches the antecedent of Rule 1, A, to obtain B. The computer would then take B and match it to the antecedent of Rule 2, yielding C as the consequent of that rule. Conversely, through a process of "backward chaining," the computer can determine which conditions or antecedents would have to be satisfied to reach result C. It would first look at the immediate antecedent of C in Rule 2 and find B. Thereafter, it would find B as a consequent of Rule 1. It would thus come up with A, the antecedent of Rule 1, as a condition that leads to result C.

A computer can carry out forward and backward chaining quickly and accurately. It can handle a sizable number of rules whose antecedents can vary in both number and complexity.

Forward and backward chaining can also be applied to sentence-level expressions. The symbols A, B, and C can be replaced with sentences without diminishing the deductive capabilities of the system. Suppose, for example, that the following substitutions were to be made:

\[
A = \text{(person is 21 years of age or older)}; \\
B = \text{(person is a major)}; \text{ and} \\
C = \text{(person has contractual capacity)}. \\
\]
The result would read:

Rule 1, if (person is 21 years of age or older)
then (person is a major).
Rule 2, if (person is a major)
then (person has contractual capacity).

A program that could deduce C when given A in the original example might have little difficulty in deducing "person has contractual capacity" when given that "person is 21 years of age or older". The problem is to make the computer treat expressions like "person is 21 years of age or older" as a unit like A. The parentheses enclosing the expressions make it easy for the computer to do this when a programming language like LISt Processor ("LISP") is used.\(^8\)

LISP is one of the computer languages that has been developed to accommodate the many ways knowledge can be represented for computer processing.\(^9\) It can be used to implement a general frame-based scheme like the one described above and has been used extensively in CCLIPS. LISP has a special syntax that must be followed to write programs. It contains a collection of primitive functions that perform basic operations on data. These functions can be combined into more elaborate functions to perform more intelligent operations. By using the tools available in LISP, knowledge can be represented in diverse patterns of knowledge structures, sometimes called "formalisms." Also, programs can be written to manipulate those patterns and their components in intelligent ways. The patterns can be tailored to meet specific needs and can range in sophistication from the very simple to the very complex.

AI researchers have developed a varied assortment of knowledge structures over the years to accommodate a wide range of information, including information about human activities such as planning and goal achievement. Although a system might be able to make the deduction described above when given the expression "person is 21 years of age or older," it might not be able to reach the same result when given "person

---

9. The models presented in this Article are written in a form that, with slight modification, can be processed directly in a LISP environment. The versions described are designed to operate in the FRANZ LISP environment of a VAX 11/780 running UNIX. Over the years systems have been developed, such as LOOPS and FLAVORS, that add objects to LISP. COMMON LOOPS is one of the most promising packages now becoming available that merges the strengths of LISP and object-oriented programming. See Bobrow, Kahn, Kiczales, Masinter & Stefik, Common Loops: Merging Lift and Object-Oriented Programs, 1986 OOPSLA PROCEEDINGS (to be published).
is 21 years old or older.” The difficulty is caused because the two expressions are not identical; one uses the word “old” instead of the words “of age.” This constitutes a significant limitation because the two expressions have the same meaning and should yield the same result. Unfortunately, many systems do not have the ability to recognize that similar, but not identical, expressions can have the same meaning. Systems that do have this ability possess some “natural language” understanding capabilities.

B. The Natural Language Barrier

The term “natural language” is used in AI to refer to ordinary language, such as that found in everyday written and oral discourse. Most expert systems that have been built in recent years have a very limited ability to understand this language. These systems are limited in their ability to determine when different representations have the same meaning in expressions like sentences, clauses, or phrases that must be uniformly represented.

AI researchers have achieved only limited success in enabling computers to understand an ordinary language such as English. Researchers refer to the difficulties involved in this endeavor as the “natural language barrier.” Initially, some researchers were very optimistic about being able to achieve results quickly in this area. However, as time went on, it became apparent that the task was going to be much more difficult than was first imagined.

Natural language text contains many ambiguities. Some of those ambiguities can be resolved if the computer is able to classify the words it is processing. A word like hand, for example, is used both as a noun (to mean laborer, part of an arm, or part of a clock) and as a verb (as in the request, “hand me the book”). There are not as many verb senses of the word as there are noun senses. Therefore, if a computer knows that the word is used as a verb in a particular context, the computer will more easily deduce its meaning than if the word is used as a noun.

Natural language “parsing” is a process by which natural language text is analyzed for the purpose of classifying and otherwise treating its component parts. One might, for example, wish to determine what parts of speech appear in a text to understand how the words are used within the whole text.

There are many parsing techniques that can be employed. Some are syntactically-oriented in that they are based on the syntactic conventions of language, whereas others are semantically-oriented in that they are rooted in previously acquired knowledge of the meaning of the text or of the meaning of language in general. The application described in this Article combines these approaches.
CCLIPS avoids the natural language barrier by combining syntactically-oriented and semantically-oriented parsing in a language called Atomically Normalized Form ("ANF"). Instead of using natural language to represent legal information, ANF employs sets of formalisms, that is, structures or models that outline the essential features of concepts. Rules and factual information are written into the formalisms and are processed by programs that convert them into a highly integrated and efficient database.

Other researchers have tried to avoid the problems of natural language by creating systems that do not require natural language understanding capabilities. The prospects of overcoming the barrier in the near future appear to be so dim that a trend toward the creation of what are known as "expert systems" has emerged. An expert system is an automated system designed to give advice that an expert might give in a particular area. 10 Although it might seem that such a system would be more difficult to create than others, it turns out that expert behavior can be simulated to a useful extent without having extensive natural language capabilities.

Commercial tools are available to build expert systems and some researchers predict that law offices will soon build their own expert systems tailored to the legal vernacular. However, until more powerful natural language understanding technologies are available, the developers of AI systems will have to compromise their goals somewhat and adopt strategies that will avoid confrontations with the natural language barrier.

II. DESCRIPTION OF OPERATIONS

This Section describes some of the difficulties CCLIPS will have to overcome to be as intelligent as its developers hope it to be. Subsection A describes some of the text processing problems that other systems are facing and the attempts the developers of CCLIPS are making to avoid these problems. Subsection B gives some examples of the kind of descriptions CCLIPS will process in order to point out the problems the system now faces. Subsection C discusses the strategies being employed to enable CCLIPS to acquire a deep understanding of the information it reads.

A. Problems That CCLIPS Will Face

The original goal of CCLIPS was the creation of a conceptual retrieval system\(^{11}\) that would incorporate some of the more promising artificial intelligence techniques that have emerged recently. Some of those techniques are described in this Article to give the reader a useful perspective from which to view CCLIPS.

By the time the development of CCLIPS was underway, it was already obvious that an advanced theory of knowledge representation would be needed to build a conceptual retrieval system to operate in the legal domain. The TAXMAN system\(^ {12} \) demonstrated that deep conceptual models would have to be used to perform conceptual retrieval tasks effectively. In the AI realm, the use of abstract constructs to represent conceptual content became popular in some circles in the late 1970's when object primitives,\(^ {13} \) scripts, plans, goals,\(^ {14} \) themes,\(^ {15} \) and the like began to be employed. Knowledge application, knowledge structure instantiation, and the use of multiple knowledge source domains emerged as the most prominent AI techniques in the text processing realm. A gradual recognition of how syntactically-oriented parsing\(^ {16} \) compliments semantically-oriented parsing\(^ {17} \) seems to have paved the way for significant advancement in automatic natural language parsing. This in turn has contributed much to the success achieved in the text processing realm.\(^ {18} \)

\(^ {13} \) See Lehnert, Representing Physical Objects in Memory, in Philosophical Perspectives in Artificial Intelligence 81 (1979); Lehnert & Burstein, The Role of Object Primitives in Natural Language Processing, 1979 Proc. Sixth Int'l Joint Conf. on Artificial Intelligence 522.
\(^ {17} \) Wilks, An Intelligent Analyser and Understannder of English, 18 Comm. ACM 264 (1975).
\(^ {18} \) For an overview of the current state of automatic natural language parsing, see K. Jones & Y. Wilks, Automatic Natural Language Parsing 7 (1983).
BORIS is a unified text processing system which incorporates memory search, instantiation, and inferencing into a unified parsing process that proceeds on a word-by-word basis. BORIS attempts to acquire an “in-depth understanding” of the narratives it reads by constructing a complete representation of the text, including all physical events, mental states, and causal connections. CCLIPS is being developed to employ similar techniques in performing more limited tasks. In this Article, the techniques used in BORIS are taken to be exemplars of modern AI strategy in the text processing realm and thus are described and referred to often.

Researchers have maintained for some time now that an in-depth understanding of a narrative requires more than knowledge of fact and causality. The point or significance of the narrative must also be grasped. Acquisition of this level of understanding depends upon the ability to recognize thematic patterns in a narrative. The process by which BORIS acquires this level of understanding is quite sophisticated. Thematic patterns cannot be recognized unless a number of processes interact properly. Consequently, BORIS has an impressive arsenal of AI techniques at its disposal, including indexing by content and split-level

19. BORIS is a text processing system designed to read and understand narrative text. It is a highly integrated system in which memory search, instantiation, and inference tasks occur ancillary to a single, unified, word-by-word parsing process. BORIS constructs a complete representation of a narrative, processing all physical events and mental states, along with causal connections. M. DYER, IN-DEPTH UNDERSTANDING: A COMPUTER MODEL OF INTEGRATED PROCESSING FOR NARRATIVE COMPREHENSION 16-17 (1983).

20. Id. at 143. The integration of parsing with other processes has been popular in some AI circles since the creation of FRUMP. Instead of answering questions about a narrative or story, FRUMP displays its understanding of the story by summarizing it. FRUMP contains “sketchy” scripts that direct parsing strategies to extract data to form a summary or the “gist” of a story. FRUMP ignores words in text that do not trigger its program. Id. at 15; DeJong, Prediction and Substantiation: Two Processes that Comprise Understanding, 1979 PROC. SIXTH INT’L JOINT CONF. ON ARTIFICIAL INTELLIGENCE 217.

21. Understanding a narrative “in-depth” requires more than simply extracting the facts of a narrative and inferring causal connections between them. An in-depth understanding recognizes what was important about a narrative, what “episodes” were significant, and indexes the “episodes” in memory for future use in similar situations. M. DYER, supra note 19, at 16.

22. Artificial intelligence programs which model cognitive processes are usually very complicated. Due to their complexity, the only way to learn how a theoretical construct will work within a large processing framework is to construct a model and run it on a computer. Once the model is implemented and run on a computer, the successes and failures of the construct become clear. This serves as a valuable research tool. For CCLIPS, BORIS is such a tool. M. DYER, supra note 19, at 355-56.

23. M. DYER, supra note 19, at 10, 165, 299.

representation consisting of cognitive—and arousal—level representation.25

BORIS attempts to detect goals, goal failures, expectations, expectation failures, plans, and plan failures that are implied or given in the text it reads. It uses this information to activate rules, inferences, and other abstract constructions which, in turn, are processed and used as part of the unified parsing process to create a deep episodic memory that can be searched using questions and answers.

Thus, BORIS faces the problem of having to determine connections between events when those connections are not explicitly mentioned in the text. For example, in processing a narrative called DIVORCE-I, BORIS encountered a paragraph that described a situation in which a character named Paul was facing a divorce. The report indicated that Paul’s wife, Sarah, not only wanted a divorce, but also the family car, house, children, and alimony. The paragraph went on to state:

Paul wanted the divorce, but he didn’t want to see Sarah walk off with everything he had. His salary from the state school system was very small.

BORIS had to interrelate all the information in the paragraph, including the italicized statement in this quotation. A human reader could recognize the connection between a small salary, alimony, and attorney fees. However, for BORIS, establishing this connection was no trivial task.

Neither the original nor the new version of CCLIPS26 face problems of this complexity at the input level. The domain of CCLIPS is the Civil Code of Louisiana ("Civil Code"). The Civil Code is divided into books, titles, chapters, sections, and articles, all of which theoretically constitute an analytic whole. As CCLIPS reads input, it enjoys the luxury of already having available many explicitly represented connections between divisions. The import of the material processed by CCLIPS is easier to detect than the themes of ordinary narratives because the subject matter of a given code provision is often revealed by its structural position. Statutes seem to be instantiations of general formalisms that represent typical patterns, such as patterns for defining rights and for describing how rights are created, extinguished, and modified.27

26. Currently, the developers of ANF are establishing a version of ANF indistinguishable in form from ordinary English. This new version will maintain all the precision of the current version of ANF. Considerable progress has been made in this endeavor. The examples presented in this Article often contain descriptions that appear to be written in ordinary English but which are written in this new version of ANF. In those instances, the descriptions are not identified as ANF representations because the scope of this Article does not include a description of this new version of ANF.
Therefore, it is usually easy to determine the point of a given segment of statutory information.

Despite the analytical quality of statutory material, CCLIPS at some point will have to face the kind of problems that BORIS now faces. CCLIPS must eventually be able to process information on at least two levels. First, it must be able to acquire an in-depth understanding of statutory material written in ANF. Second, CCLIPS must eventually be able to read and acquire an in-depth understanding of facts presented to it in ordinary English, a capability far more difficult to achieve.

It is ludicrous to think that attorneys would be willing to learn a language like ANF in order to be able to communicate with computers. That is why techniques similar to the ones used in BORIS have been under development in the CCLIPS project. Presently, the facts presented to CCLIPS are encoded in ANF. The hope is that, once ANF is developed fully, the interface between it and a selected subset of ordinary English will not be too difficult to establish. Users could then communicate with the system in English by presenting narrative facts.

The surface-level ANF representations in this Article, that is, the ANF counterparts of natural language descriptions, are actually knowledge structures or expansions. The connector HAS, for example, is used in various state descriptions, yet that connector does not flag the differences that distinguish one description from another. ANF offers numerous subtypes of the general connector HAS that can be used to flag the nature of the connection involved (e.g. HAS-POSSESS for possession and HAS-ATT for "having an attitude"). CCLIPS creates semantic depth by parsing top-level representations into lexicographical formalisms that are semantically richer than the surface-level representations.

The prototype of CCLIPS was set up to read and understand statutory material written in a simple version of ANF. The prototype was not capable of acquiring the kind of in-depth understanding of ANF representations that a system like BORIS acquires when it reads narratives. CCLIPS acquired an understanding of ANF sufficient to perform some conceptual retrieval tasks, but it did not generate an "in-depth" representation of the text. Part of the problem was rooted in ANF itself. As originally developed, ANF was not sufficiently flexible to handle the conceptual niceties that pervade the statutory realm.

Over the last two years substantial progress has been made toward giving ANF the representational power it needs to be an effective legal retrieval and reasoning system. The present goal of CCLIPS is to acquire an in-depth understanding of the ANF text. CCLIPS is being redesigned to proceed in a "bottom-up" manner to construct a deep

representational scheme that can be searched in the process of responding to presented facts. Once CCLIPS is able to read and understand ANF text, it will be able to access the resulting representational scheme held in memory when parsing facts written in English, or in a slightly formalized subset of English.

Any law student knows that a narrative description of facts can raise numerous legal issues and that the narration itself can be quite lengthy and complex. Nevertheless, the problems that CCLIPS will face in attempting to understand narratives may be less troublesome in some respects than those faced by a system like BORIS. In many cases, CCLIPS will be able to proceed in a simple "top-down" manner to analyze facts presented to it because legal rules are but abstractions that await instantiation by relevant facts. BORIS accesses knowledge structures in memory when parsing new text and uses them to perform special tasks, such as disambiguation tasks. What is already in memory can thus be used to analyze new text, and this technique should be especially effective in the legal domain because of the top-down nature of legal rules.

Although processing problems may be reduced because of the top-down nature of statutory material, this advantage may not be as great as one might imagine. One of the current dissatisfactions with key-word retrieval is the difficulty of choosing appropriate key words to be used in memory search. In many respects a similar problem would exist even if lawyers were allowed to describe factual situations to the system in ordinary English because relevant facts might be omitted. It would be a mistake to assume that the researcher already knows how to state facts in a way that will enable the system to recognize all the relevant instantiations of the rules. A researcher may have no idea what rules are applicable to the facts and hence may not be able to tailor the facts to fit the rules. For this reason, CCLIPS must use AI techniques to convert the facts into a form that will fit the representational scheme held in memory.

B. Examples of Fact Descriptions that CCLIPS Will Process

CCLIPS is being designed to receive descriptions of facts written in ANF and to determine whether those facts are covered by the rules it has in memory. The most immediate goal is to enable CCLIPS to find rules relevant to the facts by using conceptual retrieval techniques.

29. Id.
30. M. DYER, supra note 19, at 180.
Once it can do that effectively, an attempt will be made to implement some of the reasoning methods described later in this Article.

Perhaps the best way to understand how CCLIPS will operate is to consider some examples of the kind of material it is being designed to process. These examples are designed to illustrate the need for versatile expressive power in a representation language and to raise some points and issues that will be discussed later in more detail.

The first example deals with improvements made by a lessee to property under lease. It will be referred to as “EXAMPLE-1” throughout this Article. It is offered here to point out the need for an ability to represent event and state descriptions. EXAMPLE-1 reads:

The lessee improved the property under lease, which pleased the lessor.

This sentence, when taken as a factual assertion, describes both an event, that is, the improvement of the property, and a human emotional state, that is, the pleased state of the lessor. It also describes a causal relationship. On its face, it directly implies that the event of improvement caused the emotional state of the lessor. Thus, the following knowledge components can readily be distinguished:

1) EVENT ----> [lessee improved leased property];
2) CAUSAL RELATION CONNECTOR ----> [caused];
3) STATE ----> [lesser is pleased].

For CCLIPS to understand this example, it must recognize the presence and nature of the knowledge components outlined above. Since it does not have general natural language understanding capabilities, it must be given some means by which to make these determinations. ANF is designed to provide CCLIPS with much of the information it needs. It has special syntactical conventions for describing each type of knowledge component. CCLIPS detects descriptions of states, events, and relations by recognizing the special syntax used to represent them.33

Because the syntax of ANF is flexible, complex descriptions of facts can be conveniently represented. Thus, CCLIPS can digest facts and relate their knowledge components by recognizing friendly syntactical structures and by employing built-in combinative techniques. The

32. Note that the text implies the causal connection.
33. The following Sections of this Article give detailed descriptions of some of the more important aspects of this syntax. Structures used to represent emotional states are described in Section IV, infra. Structures used to represent events and causal relations are sometimes discussed individually, but at other times they are discussed together as part of a description of an inclusive process.
second example offered in this Section is such a complex description. Throughout this Article it is referred to as “EXAMPLE-2.” It reads:

John loved Mary and knew that she wanted to buy a lot he owned. He made her happy by telling her that he would consider selling it to her. He knew that Mary wanted to build a store on the lot and believed she would have the right to do so if she owned the lot. Mary's desire and belief served as the primary motivations for her purchasing the lot. She decided to buy it after John orally communicated an offer of sale to her. After buying the lot, she drove to the lot in a car and discovered a posted zoning regulation that would prohibit the construction of a store on the lot. Mary hired two lawyers who wrote a memo for her about her rights.

For CCLIPS to understand these facts well enough to perform conceptual retrieval and reasoning operations, it must recognize the differences between the events and relations described in the text. The love relationship described in the first sentence differs significantly from the act of communication referred to in the second sentence. Each of these knowledge components is treated differently in ANF.

Several other knowledge components in this example are worth mentioning. One of these is Mary's belief that caused her to enter into a contract of purchase—a belief causality. Another is the instance of decision making in which Mary decides to purchase the lot. The instance of making an offer, the instance of transportation by car, and the instance of preparing a memorandum are also knowledge components worthy of mention. All these acts and relations can be represented in a way that will enable CCLIPS to recognize them when they are encountered in parsing.

One of the most difficult tasks that the current version of CCLIPS will have to perform is to recognize legal issues presented by facts. EXAMPLE-2 presents a number of legal issues, one of which is whether Mary could rescind her contract with John because she consented to a contract tainted with error. Her error rested on the false belief that she would have the right to build the store on the lot if she owned it. Section V, infra, presents an analysis of this problem and describes a set of ANF models that can be used to assist CCLIPS in making the appropriate decisions.

C. Understanding Text

This Section describes techniques being used to enable CCLIPS to understand the information it reads. Subsection 1 describes some techniques under development elsewhere that could be used in CCLIPS. Subsections 2 through 4 describe the original versions of ANF and
CCLIPS. Subsection 5 describes how those versions are being enhanced to produce more intelligent results.

1. Techniques that Could Help CCLIPS Understand Information

A goal that is becoming more attractive as the expressive power of ANF increases is that of providing existing statutory material with formally structured descriptions that could be processed in lieu of the statutory text. The technique of processing formal descriptions of text instead of, or in addition to, the text itself has already been employed usefully in the legal domain and elsewhere. The Legal Information Retrieval System ("LIRS"), for example, uses representations of cases, statutes, and official comments to perform intelligent retrieval tasks. The TAXMAN project, and the LEGOL project, as well as the work of Meldman, each considered the general problem of designing a language in which legal rules and legal concepts might be easily expressed. Sergot has explored the possibility of using logic programming languages to represent areas of the law, and Gordon has described some advantages of using an object-oriented approach to programming languages in a knowledge representation system. It seems that interest in both the formalization and normalization of the law is still very much alive.


35. C. HAFNER, supra note 34, at 3, 61, 85.


41. See Allen & Engholm, Normalized Legal Drafting and the Query Method, 29 J. LEGAL EDUC. 380 (1978); Allen, Two Modes of Representing Sets of Legal Norms: Normalization and an Arithmetic Model, 16 PROC. THIRD INT'L CONG. ON THEME: L'INFORMATICA GIURIDICA E LE COMUNITA NAZIONALI ED INTERNAZIONALI (SESSION III) 1-61 (1983); Allen & Saxon, Computer Aided Normalizing and Unpacking: Some Interesting Machine-Processable Transformations of Legal Rules, in COMPUTING POWER AND LEGAL REASONING 495 (C. Walter ed. 1985); deBessonet, supra note 27, at 46-54; Gray, Statutes Enacted in
Large commercial retrieval systems operating in the legal domain, however, have not adopted a formal approach to representation.\textsuperscript{42} The major constraint on the use of a formal approach to represent legal information is the difficulty of encoding such a huge mass of material.\textsuperscript{43} Some researchers have attempted to avoid that difficulty by concentrating their efforts on the creation of expert systems to deal with particular legal areas or tasks (including document preparation and construction of computer-based thesauri) or on the modeling of narrow areas of law. These areas include zoning laws,\textsuperscript{44} automated legal drafting,\textsuperscript{45} taxation of stock redemptions,\textsuperscript{46} analysis of judicial decisions,\textsuperscript{47} product liability law,\textsuperscript{48} transfer tax planning,\textsuperscript{49} automation of law offices,\textsuperscript{50} general legal decision-making,\textsuperscript{51} general taxation questions,\textsuperscript{52} sentencing decisions,\textsuperscript{53} contract law questions on examinations,\textsuperscript{54} bankruptcy,\textsuperscript{55} thesaurus


\textsuperscript{43} Maggs & deBessonet, \textit{Automated Logical Analysis of Systems of Legal Rules}, 12 \textit{Jurimetrics} J. 158 (1972).


\textsuperscript{52} Kovach, \textit{Application of Computer-Assisted Analysis Techniques to Taxation}, 15 \textit{Akron L. Rev.} 713 (1982).


construction, toxic substance law, and federal taxation questions.

Others have sought to improve strategies for retrieving information from cases and statutes written in natural language.

AI research suggests that it might be feasible to write descriptive headings for Civil Code titles, chapters, sections, and articles to serve as indices for conceptual retrieval. The development of ANF is almost at the point at which it could be used effectively for this purpose. Techniques similar to ones used in LIRS, LEGOL, TAXMAN, CYRUS, and BORIS could be used in CCLIPS to perform conceptual retrieval tasks. If the formalized descriptions were sufficiently representative of the statutory content, they could function as cross-contextual abstractions. In other words, they could function like Thematic Abstraction Units ("TAUs") in BORIS.

In BORIS, plan and expectation failures trigger TAU's, that is, abstractions that function like adages in describing the import of narratives. TAU's are used in BORIS as cross-contextual indexing mechanisms because the same TAU may describe the content of more than one narrative. TAU's are triggered when BORIS encounters words that flag plan and expectation failures. A description of an expectation failure,
for example, may be accompanied by a description of an emotional state or state of arousal. States of arousal are represented by the knowledge structure AFFECT. A word like “angry” might trigger an AFFECT which, in turn, might trigger a TAU that entails that AFFECT.

In the statutory realm, techniques similar to the TAUs in BORIS can be used to trigger abstract descriptions of statutory antecedents. For example, if the description of an accident mentioned that a person “failed to replace a light bulb,” the word “failed” could be construed as an operator that would activate knowledge structures associated with the concepts of duty and legal causality. Those structures could be processed and used to analyze the text to see if the accident was caused by lack of adequate lighting and, if so, whether there existed a duty to provide adequate lighting that was breached by the failure to replace the light bulb. If a breach of duty were found, the system could then activate a set of statutory provisions indexed by the breach-of-duty formalism.

In CYRUS (Computerized Yale Retrieval and Updating System) Episodic Memory Organization Packets (“EMOPs”) are used to organize events. Descriptions of statutory events could be organized in EMOP structures according to their differences and could be accessed on that basis. TAU-like abstractions and EMOPs could help CCLIPS determine what legal effects follow from facts presented to it. Abstractions could be drawn from the facts and then matched against the TAU-like abstractions that would be attached to the statutory components. A perfect match would flag the applicable statutory component. Statutory relevance could then be indicated by locating events through EMOPs.

Another potentially useful technique for CCLIPS is deforming descriptions that can be used to produce a modified state description network. This process deforms high level descriptions into lower level descriptions and then maps them into one another based on shared properties. The deforming descriptions are then mapped into one another based on shared properties. A recently developed program called HYPO modifies descriptions of hypothetical cases to reach goals described in legal argumentation.

62. C. Hafner, supra note 34, at 65; McCarty, supra note 11, at 267-76.
63. See J.L. Kolodner, Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model 194-98 (1984); Kolodner, supra note 24, at 228.
64. Compare McCarty, supra note 36, at 868-70; McCarty & Sridharan, supra note 36, at 246.
strengthen the analogy between them and target cases in the data base. Similar techniques could be employed in CCLIPS because the system will be called upon to deform facts to match antecedents in the data base. In such cases, the facts presented could be viewed as exemplars or hypotheticals that are deformed to achieve retrieval goals.

The CCLIPS project also hopes to benefit from research conducted in the deontic realm of permissions and obligations. The role of deontic operators in ordinary legal discourse has been the topic of much recent discussion. Because CCLIPS expands high-level representations into basic representations, its developers are particularly interested in the work being done in this area. Once a basic set of permissions and obligations is established, CCLIPS can provide a foundation that will accommodate expansions of high-level representations of legal relations.

The most useful techniques for automating the legal domain in the long run will result from research in the area of cognitive modeling. Cognitive modeling is an experimental computational approach intended to provide insight into human cognitive skills. Long-term research goals for cognitive modeling are directed toward the production of process models for the representation of everyday thoughts and the understanding of everyday situations. This includes the creation of models for the ordinary processes of comprehension, story understanding, question answering, reasoning, planning, argumentation, and debate. Since legal concepts and processes are primarily rooted in ordinary concepts and processes, progress made toward reaching the aforementioned goals would complement the work being done in the legal realm.

68. McCarty, supra note 11, at 271-72; McCarty & Sridharan, supra note 36, at 250.
69. Dyer & Flowers, Toward Automating Legal Expertise, in COMPUTING POWER AND LEGAL REASONING 65 (C. Walter ed. 1985) (description of current research concerned with the design and implementation of computer programs capable of modeling various aspects of the cognitive processes of legal experts).
70. Id. at 65.
72. deBessonnet, supra note 27, at 47-63.
The following Subsections describe the original versions of ANF and CCLIPS and the ways in which those versions are being improved. The techniques discussed above may prove quite useful in furthering efforts to improve ANF and CCLIPS.

2. Original ANF

The original version of ANF was used in the prototype of CCLIPS. The expression of information in ANF involved converting the information into a set of basic (atomic) statements configured strictly in accordance with a set of syntactical rules. Thus, the resulting atomic statement consisted of a statement expressed in a basic syntactical form. Some of the forms used are:

1. \(<\text{SUBJECT}> \text { IS } <\text{SO} & \text{SO}>;\)
2. \(<\text{SUBJECT}> \text { HAS } <\text{SO} & \text{SO}>;\)
3. \(<\text{SUBJECT}> \text { CAUSES: } <\text{SUBJECT}> \text { IS } <\text{SO} & \text{SO}> \text { AND/OR } <\text{SUBJECT}> \text { HAS } <\text{SO} & \text{SO}> \text { AND/OR } <\text{SUBJECT}> <\text{VERB}> <\text{SO} & \text{SO}>;\)
4. \(<\text{SUBJECT}> <\text{VERB}> <\text{SO} & \text{SO}>;\)
5. \(<\text{THERE IS } <\text{SO} & \text{SO}>;\)
6. \(<\text{SUBJECT}> \text { INTERFERES-WITH } <\text{SO} & \text{SO}>;\)
7. \(<\text{SUBJECT}> \text { MAY } <\text{VERB}> <\text{SO} & \text{SO}>;\) and
8. \(<\text{SUBJECT}> \text { MUST } <\text{VERB}> <\text{SO} & \text{SO}>.\)

The most basic statement-form used in ANF was the following four-slot unit:

\(<\text{subject}> <\text{verb}> <\text{object}> <\text{indirect object}>.\)

This type of statement was produced by instantiating the subject and verb slots of the four-slot unit. Each of the four-slots could be instantiated with an atom (single word) or with a complex piece of information. Complex instantiations were referred to as clusters and were associated with particular slots. Each cluster had a root term, consisting of

\[\text{73. The context of the expression SO & SO, once instantiated, would consist of whomever or whatever is to be taken as the subject. Id. at 55.}\]
\[\text{74. These slots were instantiated with the contents of atomic statements created by a manual normalization process. Id. at 47-63; deBessonet, Hintze, & Waller, supra note 60, at 9-12.}\]
the corresponding primary subject, verb, object, or indirect object of the statement that was being parsed into the slots.

The statement:

A lessee improved the property under lease could be parsed into the basic four-slot unit to produce:

\[ ((\text{LESSEE IMPROVED LEASED-PROPERTY}) \text{ AT } T_1) \]

The expression "AT T_1" in this representation is a temporal description. All verbs were initially represented in the present tense along with temporal indicators that could be used to determine tense. In this example, "T_1" would have been associated with the described act of improvement so that, with reference to any "T_x" after T_1, the verb "IMPROVE" would have represented past action.

A cluster resulted when a root term was modified by one or more terms or expressions. If the original expression had read, "A lessee completely improved the property under lease," the expression would have contained a verb cluster. The resulting ANF expression would have read:

\[ ((\text{LESSEE (IMPROVE (COMPLETELY)) LEASED-PROPERTY}) \text{ AT } T_1). \]

The subexpression "(IMPROVE (COMPLETELY))" is a verb cluster.

It was also possible to place verb modifiers at the end of a sentence. For example, the statement above could have been expressed as follows:

\[ (((\text{LESSEE IMPROVE LEASED-PROPERTY}) \text{ (COMPLETELY)}) \text{ AT } T_1). \]

Negation in ANF was represented by the word "NOTCASE."

\[ (\text{NOTCASE } ((\text{LESSEE IMPROVE LEASED-PROPERTY}) \text{ AT } T_1)). \]

Complex sentences could also be represented in ANF. In general, a complex sentence refers to one or more embedded states or events. An event description, for example, could occupy the subject position in an ANF expression. The statement "A lessee improved the property under

\[ \text{NOTCASE means "it is not the case that."} \]
lease, which pleased the lessor," could have been represented in ANF as:

$$((\text{LESSEE IMPROVE LEASED-PROPERTY}) \text{ PLEASE LESSOR}).$$

The subject position in this statement is occupied by an event-description, that is, the improvement of the leased property by the lessee. Thus, with the addition of appropriate temporal information, the ANF representation would read:

$$(((\text{LESSEE IMPROVE LEASED-PROPERTY}) \text{ AT } T_1))$$

$$\text{ PLEASE LESSOR) AT } T_2)$$

$$(T_1 \leq T_2))$$

This indicates that the lessor became pleased either at the same time the event occurred or at some time after the event occurred. The following simple temporal scheme was employed:

$$(T_x = T_y) = (T_x \text{ equal to } T_y);$$
$$(T_x < T_y) = (T_x \text{ before } T_y);$$
$$(T_x > T_y) = (T_x \text{ after } T_y);$$
$$(T_x \leq T_y) = (T_x \text{ before or equal to } T_y); \text{ and}$$
$$(T_x \geq T_y) = (T_x \text{ after or equal to } T_y).$$

In addition to representing equality, the symbol "=" is used to represent identity. Thus, the expression:

$$\text{BATON ROUGE} = \text{CAPITAL OF LOUISIANA}$$

would have identified Baton Rouge as the capital of Louisiana.

The original version of ANF was designed for the Civil Code rules, which were represented in ANF formalisms. For example, the text of Article 536 of the Civil Code of Louisiana Code reads:

**Article 536. Consumable things.** Consumable things are those that cannot be used without being expended or consumed, or without their substance being changed, such as money, harvested agricultural products, stocks of merchandise, foodstuffs, and beverages.\(^{76}\)
Its "if . . . then" ANF counterpart reads:

\[
\begin{align*}
\text{(ART 536)} \\
& ((\text{if } ((\text{there is thing-x}) \text{ (thing-x is or AKO)} \\
& \quad \text{ (thing (consumable)))))
\text{then } ((\text{if } ((\text{something-x-n use thing-x})(\text{normally})))
\text{then } ((\text{something-x-n (burden (consume or expend))}
\text{thing-x}))
\text{or}
\text{\quad (if } ((\text{thing-x has substance-x})(\text{something-x-n use}
\text{thing-x}) \text{ (normally))}
\text{then } (((\text{something-x-n use thing-x}) \text{ (normally)) change}
\text{substance-x))))
\text{if } ((\text{there is thing-x})(\text{something-x-n use thing-x}) \text{ (normally)})
\text{then } ((\text{thing-x (is or AKO) (thing (consumable))}))
\text{(if } ((\text{there is thing-x}) \text{ (thing-x has substance-x)})
\text{(((something-x-n use thing-x)(normally)) change}
\text{substance-x))}
\text{then } (((\text{thing-x (is or AKO)(thing (consumable))}))
\text{(((thing (consumable)) (example-of))}
\text{= (money or ((products (agriculture)) (harvested))}
\text{or foodstuff}
\text{or beverage}
\text{or (stock (of merchandise))}
\text{or (perhaps (merchandise)).}
\end{align*}
\]

3. *Inadequacies of Original ANF*

The lack of a sophisticated temporal representation scheme was the principal inadequacy of Original ANF. Quite often sequences of events will activate legal rules. The antecedent of the rule may be viewed as a narrative description of the factual situation that will activate the rule. Narrative components of legal rules are usually stated concisely. Since each antecedent of this kind implicitly or explicitly describes a sequence of events, it follows that episodic units can be distinguished within each antecedent.

This characteristic of some antecedents increases the importance of temporal representation in ANF. The original version of ANF was somewhat weak in its ability to represent temporal relations.\textsuperscript{77} It did not

\textsuperscript{77} The power of ANF to represent temporality has been increased significantly over the last two years. For a brief overview of the temporal representation scheme used in
differentiate between points and intervals, allowing only general temporal flags to be attached to events. Notions of sequentiality could be represented, but notions such as simultaneity were difficult to represent.

4. Original CCLIPS

The foregoing discussions point out that CCLIPS must be able to detect the nature of knowledge types to decide how they should be processed. Once CCLIPS determines the type of knowledge it is dealing with, it knows where to find appropriate formalisms to build new knowledge representations. CCLIPS does this by instantiating the formalisms with the information given and by manipulating the results to perform intelligent operations. This Section describes this process as implemented in the original version of CCLIPS using causality as a model.

The original goal of the CCLIPS project was to build a system that could perform conceptual retrieval\(^7^8\) from a data base of Civil Code information.\(^7^9\) The data base of the CCLIPS prototype consisted of processed ANF representations of Civil Code articles. ANF expressions were first parsed into formalisms that bore the general structure given below:

\[
\text{(<Subject Slot>}
\text{<Object Slot>}
\text{<Indirect Object Slot>}
\text{<Verb Slot>}
\text{<Semantic Slot>}
\text{<Index Slot>)}.\]

The “Subject,” “Object,” “Indirect Object,” and “Verb” slots of this structure received the corresponding root components of atomic ANF statements. The “Semantic Slot” was filled with semantic information about the verb in the “Verb Slot.” This semantic information was derived from a verb lexicon. The information in the verb lexicon was processed to produce a set of semantic inferences. These semantic inferences were encoded into semantic slots for access by memory search. The “Index Slot” was filled with information used to map the content of the other slots into appropriate articles of the Civil Code.

\(^{78}\) Conceptual retrieval is discussed in McCarty, supra note 11, at 265.
\(^{79}\) The CCLIPS prototype processed statutory information encoded in the original version of ANF.
Complex ANF statements that contained embedded descriptions of one or more states or events were parsed into a different formalism in which the relations among components were defined less precisely than was done for atomic statements. The formalism bore a structure similar to the one diagramed below:

```prolog
(<Entity Slot-1>
 <Entity Slot-2>
 <Entity Slot-3>
 <Verb Slot>
 <Semantic Slot>
 <Index Slot>).
```

Note that the "Entity" slots in this formalism do not describe the nature of the relations between the entities or expressions that fill the slots. The relations between the contents of the slots were defined in the programs that processed the formalism. The content of "Entity Slot-1" was related to the elements of "Entity Slot-2" and "Entity Slot-3" through the elements contained in the "Verb Slot."

For example, the ANF expression:

```
(LESSEE IMPROVE LEASED-PROPERTY) PLEASE LESSOR
```

would have been represented in the database (except for indexing and semantic information) as:

```prolog
((LESSEE ;Entity Slot-1
 LESSEE LEASED-PROPERTY)
 (LEASED-PROPERTY ;Entity Slot-2
 LESSOR LESSOR)
 (NIL ;Entity Slot-3
 NIL NIL)
 (IMPROVE ;Verb-Slot
 UPSET PLEASE)
```

Several things should be noticed about this data structure. As just indicated, it relates the contents of "Entity Slot-1" to the contents of "Entity Slot-2" and "Entity Slot-3" through the contents of the "Verb-Slot." The original statement, which is complex, has been transformed into a set of noncomplex relations, each of which can be represented by
an atomic statement. The three atomic structures embedded in the instantiated formalism are:

(1) (LESSEE IMPROVE LEASED-PROPERTY);
(2) (LESSEE PLEASE LESSOR); and
(3) (LEASED-PROPERTY UPSET LESSOR).

As can be seen by comparing these atomic statements with the original input,

((LESSEE IMPROVED LEASED-PROPERTY) PLEASE LESSOR),

some of the relations produced by the process described above are not necessarily implied by the original input. For example, the process would have related "LEASED-PROPERTY" to the "LESSOR" through the verb "PLEASE," yet the inference "LEASED-PROPERTY PLEASE LESSOR" does not seem to be directly and necessarily implied by the original input. Most people, however, would recognize that there is some kind of relation between the leased property and the lessor that has something to do with the emotional state created in the lessor. The prototype of CCLIPS would not have attempted to classify the nature of that connection.

5. New Strategies for CCLIPS

It is important to note that because the original version of CCLIPS did not attempt to classify the nature of relations between components of ANF statements, it acquired a somewhat shallow understanding of the information it read. The new strategies used in CCLIPS attempt to classify connections of this sort by generating a penumbral set of causal inferences to surround each causal statement. For instance, PLEASE in the example above would carry an indication of the nature of the relation between the LESSOR and the LEASED-PROPERTY.

The new strategies designed for deeper understanding of text enable CCLIPS to understand the nature of relations between knowledge components. For example, the meaning of a statement that describes an event is defined in part by the relations evidenced by the causal structure of the statement. Therefore, to understand the meaning of the statement, CCLIPS must understand the causal relations described or otherwise implied by the statement.

CCLIPS understands causal relations by knowing and recognizing a statement's causal structure. The process of understanding consists of
instantiating and expanding formalisms that match the causal structure of the statement. 80

III. NOTATIONAL CONVENTIONS

This Section discusses the notation used for the complex syntactical examples discussed in Section II. Since the notation used in the original version of ANF differs from the current version, the differences between the two forms are noted.

A. Notation Used in Original ANF to Represent Variables

In the original version of ANF, any expression that functioned as a variable was flagged as such by an uppercase letter appended by a hyphen to the end of the expression. In the expression LESSEE-X, LESSEE would have functioned as a variable and could have been instantiated by any constant or variable that met the criteria for a lessee. The same uppercase letter could be used to flag different variables. To indicate that a variable could be instantiated by one or more constants or variables, the letter N was appended by a hyphen to the uppercase letter that flagged the expression as a variable. The expression “LESSEE-X-N,” for example, could have been instantiated by one or more constants or variables that satisfied the criteria for a lessee.

B. Notation Used in Current Version of ANF to Distinguish Types

In the current version of ANF, the following notational scheme has been adopted to distinguish types from individuals. The scheme bears some similarity to conceptual graph notation 81 which maps function type concepts onto a set T, whose elements are called type labels. 82 Type labels represent generic terms; they specify some attribute of the things or acts referred to. A generic term applies equally well to whatever has the specified attribute. 83

In ANF, the generic term “lessee” would be represented by the type label LESSEE. Type labels in ANF are written in uppercase character strings. Subscripted, smallcase letters attached to type labels function as generic markers in ANF. The expression LESSEE_x thus refers to an

80. See Appendix D for a more detailed description of causal inference generation and other types of causal connections such as the connection between the LESSOR and the LEASED-PROPERTY in this example.
82. Id. at 49-83.
83. Id. at 84-87.
unspecified individual of type LESSEE and may be read "some lessee x" or "a lessee x." The subscripted, smallcase letters also serve a cross-referencing function. If different smallcase letters are attached to identical type labels, the expressions refer to different, unspecified individuals unless the contrary is indicated by an explicit representation of equivalence. Representations of equivalence override conflicting representations.  

Multiple occurrences of the same expression, including type label and subscript, refer to the same individual. When the same subscript is attached to two or more different type labels, they refer to the same individual, which in such a case would conform to both type labels. The expressions LESSEE_x and MAN_x refer to the same individual, who is both a lessee and a man. By default, a type label that has no subscript implicitly carries a generic marker. The expression LESSEE thus refers to some unspecified individual of type LESSEE.  

If in a given context the same type label appears without subscript on multiple occasions, the label implicitly carries the same generic marker and thus refers to the same unspecified individual. When two or more different type labels have no subscripts, they are presumed to carry different generic markers unless the contrary is indicated by an explicit representation of equivalence. Finally, a relation of conformity is assumed to exist between a marker and its label. Thus, the expression LESSEE_x implies that the unspecified individual x conforms to the type LESSEE.

C. Notation Used in Current Version of ANF to Distinguish Individuals

Subscripted numbers attached to type labels function as surrogates or individual markers. They refer to particular individuals. Thus, the expression LESSEE_{151} refers to a particular individual of type LESSEE.

For notational convenience, proper names are expressed in the same form as type labels but refer to particular individuals. Any place a proper name appears in the notation CCLIPS assumes that the proper name uniquely identifies the individual to which it refers. Just as the

84. For example, the expressions LESSEE_x and LESSEE_y refer to different, unspecified individuals unless it is given explicitly that LESSEE_x = LESSEE_y.
85. J. Sowa, supra note 81, at 84-87.
86. The expressions LESSEE and LESSOR thus refer to different unspecified individuals because they are presumed to carry different generic markers.
88. See J. Sowa, supra note 81, at 84.
expression LESSEE may be used to refer to a particular individual named John who is a lessee, the expression JOHN may be used to refer to that same individual. In such a case the expression JOHN is not a type label although it is expressed in uppercase letters. This Article uses popular names so that the reader will not confuse names with type labels.

D. Notation Used in Current Version of ANF to Represent Variables

When a type label is preceded by the "?" symbol, the expression functions as a variable and may be instantiated by any constant or variable that conforms to the type label. Variables are represented this way in the ANF formalisms, whereas generic marker notation is used in the ANF descriptions of "facts." The expression ?LESSEE is a variable that may be instantiated by any constant or variable that conforms to the type LESSEE. When different uppercase letters are attached to identical type labels of variables, the same constant or variable cannot be used to instantiate the variables unless there is an explicit indication of equivalence. The expressions ?LESSEE-X and ?LESSEE-Y may not be instantiated with the same constant or variable. When the same uppercase letter is attached to identical type labels, the instantiation of one such variable by a given constant or variable requires the uniform instantiation of the others by the same constant or variable.

Because generic terms are like variables in that they represent unspecified individuals of a given type, CCLIPS treats generic-marker notation as variable notation when performing certain matching operations. By doing so, it is able to match one ANF representation with another through instantiation. When an expression bears a generic marker and is used in a representation of a rule, CCLIPS treats the expression as a variable. CCLIPS is thus able to match descriptions of facts with antecedents of rules. For example, through instantiation, the fact-description

LESSEE \_y IMPROVES LEASED-PROPERTY \_z

can be made to match the antecedent of the following rule:

IF LESSEE \_w IMPROVES LEASED-PROPERTY \_x
    THEN LESSEE \_w HAS RIGHT \_y.

The match is achieved by treating the expressions LESSEE \_w and LEASED-PROPERTY \_x of the antecedent of the rule as variables, each of
which may be instantiated by any constant or variable that conforms to the variable’s type label. Thus, the expression LESSEE\textsubscript{y} of the fact description matches the expression LESSEEW of the rule because the unspecified individual “y” referred to by the expression LESSEE\textsubscript{y} is of type LESSEE, which is precisely the condition that must be fulfilled for a valid instantiation of LESSEEW to occur. The expression LESSEEW is treated as if it were the variable ?LESSEE, which may be instantiated by any variable or constant that conforms to the type label LESSEE. The “y” of the expression LESSEE\textsubscript{y} of the fact description is treated as a variable that conforms to the type label LESSEE and thus may be used to instantiate the generated variable ?LESSEE.

E. Special Notation Adopted for Readability

For the convenience of the reader, the numerous examples presented in the following Sections are encoded in a version of ANF that is more readable than the one actually processed by CCLIPS. In the examples, generic or individual markers are sometimes attached to expressions as subscripts. In the version processed by CCLIPS, the markers are not imploded to the ends of expressions, which would necessitate numerous exploding operations during processing, but are set apart from them and are processed separately. The connectors “and” and “or” used in the notation should be understood in their normal senses, not as LISP operators.

IV. REPRESENTATION OF FACTS AND LAW IN ANF

CCLIPS must be able to analyze fact descriptions of the sort encountered by lawyers in practice in order to operate effectively in the legal domain. Since lawyers analyze world situations in search of legal effects, CCLIPS must be able to do the same. CCLIPS thus requires a rich assortment of available techniques to understand reports about ordinary facts and the relations between them, both of which are primary components of real-world knowledge. This Section discusses the representations of facts and law in ANF, using causality as a model.

89. The morphology of a word can be obtained by an exploding operation. Since the morphology of a word depends upon the spelling of the word, it is necessary to break up a word into letters. This exploding operation takes a word and returns a list of letters. For example, the command “explode boys” produces “b o y s.” Next, suppose you want the dictionary definition of “b o y s,” or “b o y.” This can be done by exploding the letters “b o y” to “boy” in order to ask a dictionary for the definition. See E. CHARNIAK & D. MCDERMOTT, INTRODUCTION TO ARTIFICIAL INTELLIGENCE 187-88 (1985).
A. Causality

Causality is one of the most important real-world phenomena that CCLIPS will have to understand to operate principles of causality that are defined or otherwise recognized by law—to be able to give sound legal advice. The notions of "proximate cause" and "remote cause" are widely known in the legal community.

Because of the importance of causal notions in law, ANF has been designed as an action-oriented language so that these notions can be represented in a form that CCLIPS can handle with convenience. ANF uses a primitive model of causality to represent the cause-effect relationships of the real world, and it uses this same model to represent other notions of causality, such as legal causality.

1. Representing Causality

The representation of causality bears heavily upon the intelligence of CCLIPS because CCLIPS uses knowledge of causal structures not only to recognize causal relations in parsing, but also to build an in-depth understanding of the information it reads. Representations of events written in ANF describe the essential causal features of those events in a way that enables CCLIPS to recognize their presence; however, the mere recognition of those features does not constitute a deep understanding of them. CCLIPS therefore must build an in-depth understanding by using the knowledge of causality available to it. It accesses causal formalisms as part of an instantiation and expansion process and generates inferences from those expansions using a built-in causal logic. Without its knowledge of causal structure, CCLIPS would not be able to perform these operations.

When thinking of ordinary cause-effect relations, one normally conceives of something causing a result or state of affairs. The "something" that brings about the result is sometimes called the "agent," and the result itself can be referred to as the "end-state." If a lessee causes property under lease to be improved, the lessee may be thought of as the agent that brought about the end-state, that is, the improved state of the property.

---

90. Although one may normally think of the agent in a causal relation as something that acts consciously to bring about the result, this Article uses "agent" in a broader sense to include anything that is responsible, in some way, for bringing about an end-state. This conception allows ANF to accommodate a wide range of phenomena by using a simple causal formalism, even though those events do not entail agents that have acted consciously. An example of such an agent would be a painting that would cause someone to be delighted at the mere sight of it.
The syntax for representing causal events contains sundry symbols for representing participants and causal relations. A primitive causal statement in ANF consists of a unary subject linked to a unary object by a causal link. More complex causal statements consist of multiple or complex subjects or objects joined by one or more causal links. The most basic causal formalism is represented in ANF as:

[(?A CAUSE ?B)].

In this representation "?A" occupies the "agent" or "subject" position in the relation, and "?B" occupies the "end-state," "product," or "object" position. The expressions "?A" and "?B" function as variables and may be instantiated with representations of entities, states, or events.91

a. Entities: Persons and Things

Persons and things are represented in a causal formalism as a single unit—an entity—or as a set of units—a cluster. A reference to two persons, say John and Mary, constitutes reference to a cluster consisting of John and Mary.

b. States

States are represented in state-description formalisms. The basic formalism used to represent states is:

[(<Subject> <Verb> <So & So> (TOWARD <So & So>))]

where the "<Verb>" is either HAS or IS and the directional expression TOWARD <So & So> is optional. Each of the following expressions represents this formalism:

(LESSOR IS PLEASED);
(LESSOR HAS PATIENCE);
(LESSOR IS TALL); and
(LESSOR IS ANGRY TOWARD LESSEE).

---

91. See Appendix B for types of items that can be instantiated into causal formalisms.
c. Events

Event descriptions in ANF denote action and imply the existence of at least one causal link. Events define the causal relationship between an agent and an end-state and are taken to be any relation between entities, states, or subevents that contains or implies a causal link. Each of the following expressions represents an event:

(LESSEE CAUSED (LESSOR IS PLEASED));
(JOHN CAUSED (MARY IS HAPPY)); and
((JOHN CAUSED (MARY IS HAPPY)) CAUSED (JACK IS GRATEFUL)).

Event descriptions include any statement that contains an action verb. These action verbs can be converted into a statement that contains a causal link. For example, the statement:

(LESSEE PLEASED LESSOR)

can be expanded into the following statement that contains the causal link CAUSED:

(LESSEE CAUSED (LESSOR IS PLEASED)).

Voluntary personal acts can also be described as an event. These personal acts are represented by referring to the same entity, state, or event both in the subject position and in the END-STATE position of a causal statement.\(^\text{92}\) The representation of

(JOHN MOVED)

would be

(JOHN CAUSED (JOHN HAS MOVED)).

d. Beliefs

Both physical and mental causation can be represented conveniently in ANF. Mental activities are represented by statements that have the following general structure:

(\textit{<Subject>}' <\textit{Mental Act}>' (\textit{<Product of Mental Act>}'
).

The ANF statements:

\begin{align*}
\text{(LESSEE BELIEVES (\textit{<So \& So>'})^{93}} \\
\text{(LESSEE INTENDS (\textit{<So \& So>'})}
\end{align*}

are partial instantiations of this structure. Also note that mental activities can be interpreted as the causes of some physical acts.\textsuperscript{94}

2. \textit{Causal Links}

The nature of the relationship between an agent and an end-state varies with the circumstances. In the law, for example, there are some relationships in which the agents are considered to be mere remote causes of the end-states. Any lawyer knows that the relations in those cases differ significantly from ones in which the agents are found to be proximate causes of the end-states.\textsuperscript{95} The differences, of course, are partly defined by the legal consequences that result from each type of relationship.

Because causal relationships differ, the causal representation language used in CCLIPS must be able to capture differences in a way that will enable the system to detect them and to go about its business of trying to understand them. The representational devices used in ANF to represent these differences are called "causal links." They consist of special symbols designed for use in particular causal situations. CCLIPS is being designed to recognize thousands of types of causal connections. This Section discusses six factors used to distinguish causal connections and defines and otherwise describes some of the more important ones.

a. Proximate and Remote Cause

ANF distinguishes proximate cause (PROX-CAUSE) and remote cause (REM-CAUSE) by noting the position of the agent in relation to the end-state. In a causal representation, the position that is closest to

\textsuperscript{93} The "So \& So" segments of these expressions represent mental possessions, such as beliefs and intentions.

\textsuperscript{94} R. Schank & C. Riesbeck, \textit{supra} note 92, at 28-29.

\textsuperscript{95} Proximate cause is the limit courts place on an actor's responsibility for the consequences of his actions. Theoretically, the consequences of an act can be traced backwards in time to the "dawn of human events" and forwards in time to eternity. Legal cause is limited to causes so closely connected to the result that the law is justified in imposing liability. W. Prosser & P. Keeton, \textit{Handbook of the Law of Torts} 264 (5th ed. 1985).
the end-state is the proximate cause. For example, in the ANF expression:

\[
((\text{MATCH CAUSES FIRE})(\text{FIRE CAUSES SMOKE}))
\]

the proximate cause of the smoke is the fire. The match is the proximate cause of the fire but only the remote cause of the smoke.

b. Contributing Cause

The contributing cause (CONTRIB-CAUSE) link in ANF represents the nature of the participation of the agent in the causal act as either a solitary or joint agent. Whenever a particular frame of reference is not the sole cause of the end-state, the agent is taken to be a mere contributing cause of the end-state. If two persons each put one quart of water into a bucket, each person contributed to cause (in ANF, CONTRIB-CAUSED) the bucket to contain two quarts.

c. Enabling Cause

ANF defines the relation of one causal act to another is defined in terms of how the acts bring about the end-state. This is represented by the ENABLING CAUSE (ENAB-CAUSE or ENC) link. When something is used by another to bring about an end-state, the causal relation may be described as one of enablement. In EXAMPLE-1, if the lessee used tools to improve the property, the tools would be the enabling cause (ENAB-CAUSE) of the end-state: “lessor is pleased.” The tools enabled the lessee to improve the property under the lease.

d. Create Cause

The nature of the end-state as either a “created product” or “other results” is represented by the CREATE-CAUSE link. This identifies the relation between a “creator” and a “product.” For example, to represent the idea that a lawyer produced a memorandum, the following ANF statement might be used:

\[
(\text{LAWYER CREATE-CAUSE MEMO}).
\]

96. R. Schank & C. Riesbeck, supra note 92, at 28-29.
97. The relation of enablement is broken down into two categories when a system or machine performs a function that brings about an end-state. First, activation (ENAB-ACTIVATE-CAUSE) occurs when a person turns on a switch that starts a machine and the machine performs a task. The person enabled the machine to perform the task by activating it. Second, activation and operation (ENAB-OPERATE-CAUSE) occur when a person activates a machine and operates it to bring about an end-state. In this case, there is a stronger participation by the acting party in the causal events.
CREATE-CAUSE is a special link that activates task-oriented processes in CCLIPS. These processes attempt to define the scope of activity that created the product. The processes activate "angels" that examine the statement to find the point at which some part of the end product has been created.

### e. Possible Cause

The strength or weakness of the causal connection between the agent and the end-state is defined in terms of the degree of certainty that such a causal relation exists. This is represented by using the POSSIBLE-CAUSE (POSS-CAUSE) or MAYBE-CAUSE links. POSS-CAUSE and MAYBE-CAUSE are employed to show that a given causal link may represent one or more special causal relations that bring about an end-state. These links flag the possibility that there exists a causal connection between the agent and the end-state. When such a causal link description is generated as an inference, the inference is referred to as a "penumbral inference." These inferences are generated in accordance with special rules designed to produce inferences that do not follow by deduction but which are nevertheless useful.

POSS-CAUSE and MAYBE-CAUSE links are the weakest causal links used in ANF; the weakness is rooted in the lack of certainty that a

---

98. "Angels" in CCLIPS serve the same purposes that "demons" serve in some other systems. Demons are delayed action knowledge processors. They wait until their test conditions are satisfied to execute their actions. Activated demons remain alive until: (1) the test condition is completed; (2) the test condition has been completed by another demon; or (3) the test condition no longer has any chance of being satisfied. M. Dyer, supra note 19, at 165-66.

99. The scope of the creation process can be accessed as an independent knowledge structure. Such a knowledge structure may be used to understand relations among the entities referred to in the text. For example, if the creation of a part of a thing is a prerequisite to the creation of the whole thing and it is known that the crucial part never came into existence, it may be inferred that the thing never came into existence.

100. For example, the sentence "a lessee put a mailbox on the leased property" could be represented by:

\[(\text{LESSEE CAUSED (LEASED-PROPERTY HAS MAILBOX))}\]

The above statement, in addition to implying that the lessee caused a state relation to exist between the leased property and the mailbox, could imply that the lessee also created the mailbox. Because the creation of a state relation may or may not entail the creation of one or more constituents of the relation, CCLIPS faces special problems when it attempts to process state descriptions. To solve this problem, CCLIPS generates inferences that are possibly valid to serve as representations of the special relations by accessing a special causal logic. Thus, to cover the possibility that a given causal link might represent only part of the creation process, a statement representing that part would be generated. Inferences of this kind ultimately may be represented by employing the POSS-CAUSE and MAYBE-CAUSE links and are referred to as "penumbral inferences." See Appendix D.
causal connection exists between the agent and the end-state. Whenever a segment of a representation that occupies the subject or agent position in a causal description is related to the end-state of that description, either a POSS-CAUSE or a MAYBE-CAUSE link is employed to represent the relation.

f. Principle Cause or Subsidiary Cause

The extent to which an agent contributed to bring about the end-state is represented by the PRINCIPLE-CAUSE (PRIN-CAUSE) or SUBSIDIARY-CAUSE (SUBS-CAUSE) link. The PRIN-CAUSE link implies that the agent was a substantial factor that brought about the end-state. The SUBS-CAUSE link implies that the agent was an influential, but not a substantial, factor that brought about the end-state. Thus, given that something is the principal cause of a particular end-state, all other causes would be subsidiary.101

3. Causal Link Descriptors

Causal links can be associated with other information that can be used to further clarify causal relationships. Causal links, along with descriptors, can serve as nodes to which information can be attached.

ANF uses verbs and causal links as one method of explaining the meaning of an act that produces an end-state. This method breaks the act into parts or subacts. ANF employs the symbol CONSISTS OF (“CO”) to represent whole-part and act-subact relationships. For example,

(C CO (A AND B))

indicates that “C” consists of a set of acts that constitute “A” and “B.”

Another method of explaining or describing causal links is to attach one or more clauses to them that describe how an end-state occurs. The expression BY MEANS OF (“BMO”) is one expression used in ANF to introduce clauses of this kind.102 BMO clauses may be attached to ordinary verbs and causal links. For example, the expression “lessee pleased lessor,” can be represented in ANF by:

(LESSEE CAUSED (LESSOR IS PLEASED)).

101. The classifications of the causal links recognized in ANF overlap to some extent. For example, an agent that is in a proximate cause position to an end-state may share that position with another agent so that the causal link qualifies both as a PROX-CAUSE link and as a CONTRIB-CAUSE link.

102. See Appendix C.
This representation indicates that the lessee was the cause of the emotional state of the lessor. It does not, however, indicate what the lessee did to please the lessor (if anything). If it were known that the lessor was pleased because the lessee had improved the leased property, this could be added to the ANF representation by using a BMO description:

\[ ((\text{LESSEE CAUSED BMO}) \ (\text{LESSEE (CAUSED BMO (\text{LESSEE IMPROVE LEASED-PROPERTY})) (LESSOR IS PLEASED)}))].\]

If what pleased the lessor is a combination of events, for example, the cleaning and painting of the leased property, these events could be placed in a statement as follows:

\[ ((E_1 = (\text{LESSEE CLEANED LEASED-PROPERTY})) \ (E_2 = (\text{LESSEE PAINT LEASED PROPERTY}))).\]

Then the whole idea could be represented as:

\[ ((\text{LESSEE (CAUSED BMO CAE) (LESSOR IS PLEASED)}) \ (\text{CAE = (E}_1 \ \text{AND E}_2))).\]

where CAE represents a cluster of acts, E1 and E2, that pleased the lessor.

ANF also provides for abbreviated descriptors for the means by which end-states are brought about. As noted above, BMO clauses may be instantiated with statements. In addition, the BMO clauses may be instantiated with verbs or other expressions that describe, in abbreviated form, the acts that were performed to bring about the end-state. In such a case, the expression VBMO is used to introduce the clause. For example,

\[ (\text{LESSEE (IMPROVED VBMO PAINTING) LEASED-PROPERTY}) = (\text{LESSEE (IMPROVES BMO (LESSEE PAINT LEASED-PROPERTY)) LEASED-PROPERTY})].\]

Another abbreviated descriptor is the expression IN ACCORDANCE WITH ("IAW"). This descriptor is used to introduce clauses that refer to rules, standards, or criteria that somehow define, govern, or qualify the term modified by the clause. The ANF statement

\[ (\text{COURT-PROCEEDING (PROCEEDED IAW RULES-OF-COURT))}]
represents the idea that a court proceeded in accordance with some rules (here represented by the expression RULES-OF-COURT).

IAW may also modify qualifiers. The statement

\[(\text{CAR IS (GOOD IAW STANDARDS)})\]

represents the idea that the car is good based on the standards referred to by the expression STANDARDS.

4. Applying Causal Representations

Causal representations are used by CCLIPS to make crucial decisions about the nature of knowledge types it encounters in parsing. CCLIPS builds deep models of knowledge types by generating successive causal representations in order to understand causal phenomena.\(^{103}\) The best way to understand the importance of the roles that causal structures play in the CCLIPS environment is to consider how causal structure bears upon meaning.

The causal structure of a statement is an integral part of its meaning. The English expression “a lessee improved the property under lease” may be represented in ANF as follows:

\[
((\text{LESSEE CAUSED (LEASED-PROPERTY IS IMPROVED)})
\text{ CAUSED (LESSOR IS PLEASED)}).
\]

The structure of the original English expression allows for the following interpretations:

(1) The improvement of the property may have been the crucial factor that caused the lessor to be pleased; or

(2) The fact that the lessee was involved in the event may have been the crucial factor that caused the lessor’s emotional state; or

(3) Both the improvement of the property and the lessee’s involvement may have been crucial factors that brought about the end-state.

Possibility (3) seems to best capture the meaning of the original expression. The ANF version of the original expression represents possibility (3). This is evidenced by the fact that

\[
(\text{LESSEE CAUSED (PROPERTY IS IMPROVED)})
\]

\(^{103}\) Section V, \textit{infra}, describes how these techniques can be usefully employed in the legal domain.
is placed in the "agent" position in relation to the end-state. The causal structure of possibility (3) can be represented as follows:

$$((?A \text{ CAUSE}2 ?B) \text{ CAUSE}1 (?C \text{ IS } ?D)).$$

Possibility (1) above could be represented in ANF as follows:

$$((\text{LESSEE CAUSED (LEASED-PROPERTY IS IMPROVED)}) \text{ AND } ((\text{LEASED-PROPERTY IS IMPROVED}) \text{ CAUSED (LESSOR IS PLEASED))).$$

This representation could be instantiated into the following causal structure:

$$((?A \text{ CAUSE } ?B) \text{ AND } (?B \text{ CAUSE } (?C \text{ IS } ?D))).$$

This causal structure differs from the one given for possibility (3).

The structural differences distinguish the two possibilities. The causal structure of possibility (3):

$$((?A \text{ CAUSE}2 ?B) \text{ CAUSE}1 (?C \text{ IS } ?D)),$$

when considered out of context, does not syntactically or semantically rule out the possibility that the CAUSE1 link represents one or more special causal relations. This is true because the link CAUSE may qualify as one or more of the special causal relations described above. To cover the possibility that the link CAUSE1 represents an act of creation, CCLIPS can replace the link CAUSE1 with the link CREATE-CAUSE1. This generation process may be carried out recursively. Thus, once CCLIPS generates:

$$((?A \text{ CAUSE}2 ?B) \text{ CREATE-CAUSE } (?C \text{ IS } ?D))$$

as a causal structure, it could then generate other possible inferences, such as,

$$((?A \text{ CAUSE}2 ?B) \text{ CREATE-CAUSE } ?D).$$

104 For convenient reference, numbers have been attached to the causal links in this diagram.
105 See Appendix D for a detailed description of the theory behind the generation process.
5. Abstractions Show Causal Relation

Often, detailed causal information cannot be translated into ANF because it is too complicated or too incomplete for CCLIPS to perform conceptual retrieval. For example, the idea of a car transporting a person can be represented by:

\[(\text{CAR (TRANSPORTS BMO ?SOME-ACTS) PERSON}).\]

In instantiating the variable ?SOME-ACTS, one must consider that an explanation of how a car is able to transport a person is very complex and entails an explanation of how a car engine operates. If an appropriate knowledge structure were not available, a representation of the operation of an engine would have to be created, which would mean that the causal chains involved would have to be represented explicitly. This is too burdensome a task to be feasible. One alternative would be to terminate the expansion process at this point. However, this would make the representational scheme too shallow. Another possibility would be to create an alternative representation to substitute for the detailed one.

When one wishes to avoid having to describe causal acts in detail, the causal acts may be described abstractly.\(^{106}\) In the expression:

\[(\text{(LAWYER1 WROTE PART-MEMO-A) (LAWYER2 WROTE PART-MEMO-B)})\]

the cluster of acts that resulted in the end-states, MEMO parts "A" and "B," could be represented by CAE.\(^{107}\) The actors that participated to produce the MEMOS, LAWYER1 and LAWYER2, could be represented by CE.\(^{108}\) Then, the expression PARTICIPATES-IN can be used in ANF to represent participation by one or more actors in one or more events. The entire scheme above can be abstractly represented as:

\[(\text{CE PARTICIPATES-IN CAE}).\]

Instead of terminating the expansion process at the point at which one faces the task of creating a detailed representation of the chain, an abstract description of the chain can be created by using representational

\(^{106}\) M. Dyer, supra note 19, at 29.

\(^{107}\) See Appendix B.

\(^{108}\) M. Dyer, supra note 19, at 29.
devices. These types of descriptions may be instantiated into BMO slots in lieu of causal chain descriptions.

B. Goals, States of Arousal, and Mental Acts

CCLIPS uses a complex array of knowledge structures to build deep representations of emotional states and interpersonal relationships. The following Subsections describe how states and relationships can be represented in ANF.

1. Descriptions of SOUL

For purposes of describing psychological aspects of human beings in ANF, the inner being of a person is represented by SPIRIT and SOUL. Only the aspects of SOUL are described here since SOUL descriptions are sufficient for ordinary cognitive and arousal level representation. The SOUL of a person consists of mental possessions ("MPs"), emotions ("EMOTs"), attitudes ("ATTs"), desires ("DESs"), and feelings ("FEELGs"). The following primitive statements refer to these aspects of SOUL:

(?PERSON HAS ?MP);
(?PERSON HAS ?EMOT);
(?PERSON HAS ?ATT);
(?PERSON HAS ?DES); and
(?PERSON HAS ?FEELG).

SOUL also comprises mental acts such as thinking, believing, deciding, and setting goals. Subsection 2, infra, describes how a text processing system can use descriptions of psychological phenomena to understand the text it reads. Subsections 3 through 7, infra, describe how these aspects of SOUL are represented for CCLIPS. Examples of their use in the legal domain can be found in Section V, infra.

2. Descriptions of Psychological Phenomena in Text Processing

Like CCLIPS, some text processing systems perform functions using psychological factors. This Subsection briefly explains how BORIS builds an in-depth representational scheme of psychological phenomena in order to show how CCLIPS employs similar techniques when parsing psychological statements expressed in ANF.
In BORIS,\textsuperscript{109} emotional reactions are represented in a knowledge structure called an AFFECT, which has six basic components:

1. \textbf{STATE} (describes the emotional state as being either positive (POS) or negative (NEG));
2. \textbf{CHAR} (identifies who experiences the emotion);
3. \textbf{G-SITU} (refers to the goal situation that gave rise to the emotional state);
4. \textbf{TOWARD} (optional) (identifies and object where the emotion is directed toward that object);
5. \textbf{SCALE} (optional) (describes the intensity of the emotion);
6. \textbf{E-MODE} (optional) (describes expectations that characters have about future or on-going outcomes).

In the AFFECT lexicon of BORIS, information about AFFECTs is cross-indexed with descriptions of goal situations. Complex AFFECTs can be decomposed to more basic AFFECTs. Examples of the kind of information that appears in the lexicon of BORIS are presented below.\textsuperscript{110}

\begin{itemize}
  \item \textbf{Lexical items}: happy, joyous
  \item \textbf{Affect Info}:
    \begin{itemize}
      \item (AFFECT
      \item \quad \text{STATE (POS)}
      \item \quad \text{CHAR x}
      \item \quad \text{G-SITU (a)}
    \end{itemize}
  \item \textbf{Goal Situation (a)}: goal of x achieved
  \item \textbf{Lexical items}: grateful, thankful
  \item \textbf{Affect Info}:
    \begin{itemize}
      \item (AFFECT
      \item \quad \text{STATE (POS)}
      \item \quad \text{CHAR x}
      \item \quad \text{G-SITU (b)}
      \item \quad \text{TOWARD y)
    \end{itemize}
  \item \textbf{Goal Situation (b)}: y caused goal situation (a) to occur.
\end{itemize}

If BORIS were to encounter the word "grateful" in parsing, the AFFECT information associated with that word would be activated. Thus, goal situation (b), "y caused goal situation (a) to occur," would be activated. In turn, this would activate the goal situation (a) because goal situation (b) explicitly refers to goal situation (a).

BORIS interrelates emotional states, goal situations, interpersonal themes ("IPTs"), and thematic abstraction units ("TAUs") to produce

\textsuperscript{109} \textit{Id.} at 105-06.
\textsuperscript{110} \textit{Id.} at 107.
knowledge structures that can be accessed to answer questions about the emotional reactions of characters in the narratives it reads. The interaction of these AFFECTs, GOALs, TAUs, and IPTs enables BORIS to build an in-depth representational scheme. CCLIPS employs similar techniques when parsing statements expressed in ANF. Beliefs, intentions, goals, and emotional states are represented in a way that allows CCLIPS to recognize the connections between constituent expressions. The following Subsections describe how mental possessions, emotions, desires, and feelings are represented in a way that enables CCLIPS to carry on its operations.

3. Representation of Mental Possessions

Mental possessions are defined in CCLIPS as objects of mental acts. For example, something that a person believes may be classified as a mental possession ("MP"). The following simple formalism is used to represent MPs:

\[(\text{subject} \text{ HAS } \text{MP}).\]

Statements that represent mental acts imply MP statements; thus, the act of believing referred to above would imply the following instantiation of the MP formalism:

\[(\text{PERSON HAS BELIEF}).\]

Subsection 6, infra, describes how the mental acts themselves are represented.

4. Representation of Emotions, Attitudes, and Desires

The general forms used to represent attitudes are:

(1) \[(\text{subject} \text{ HAS } \text{attitude} \text{ TOWARD } \text{object}).\]
(2) \[(\text{subject} \text{ IS } \text{attitude} \text{ TOWARD } \text{object}).\]

ANF allows for the nature of the attitude or emotion to be described. The following are five different multi-form representations of "John loves Mary":

111. Id. at 127-28.
(1) (JOHN LOVES MARY);  
(2) (JOHN HAS ((ATT, POS) = LOVE) TOWARD MARY);  
(3) ((JOHN HAS (ATT, POS) TOWARD MARY) (ATT = LOVE));  
(4) (JOHN HAS ((EMOT, POS) = LOVE) TOWARD MARY);  
(5) (JOHN HAS ((FEELG, POS) = LOVE) TOWARD MARY).\(^{112}\)

Descriptions (2) and (3) treat “love” as an attitude, description (4) treats it as an emotion, and description (5) treats it as a feeling. The intensity of emotions, feelings, and attitudes may be indicated by attaching an intensifier to the description of the emotion, feeling, or attitude. To represent the idea that John loves Mary deeply, the intensifier DEEP could be used to represent the depth of the love. The following representation, which treats love as an emotion, makes use of the intensifier DEEP:

$$((\text{JOHN HAS (EMOT, POS) TOWARD MARY}) \quad \text{(EMOT, POS) = LOVE}) \quad \text{(EMOT, IS DEEP)})),$$

CCLIPS has a lexicon of ordered intensifiers. “True love” and “deep love,” for example, are more intense than mere “love.” At the most basic level, intensifiers are encoded in symbols that represent the degree to which the referenced intensity is above or below the norm. A similar technique is employed in BORIS.\(^{113}\)

The general formalism used to represent desire is:

$$(<\text{subject}> \text{ HAS } <\text{desire}> \text{ TOWARD } <\text{object}>)).$$

The portion of the formalism within brackets is optional. Acts of desire are represented in the same way as are other mental acts.

5. **Representation of Feelings**

Feelings, attitudes, and emotions can be described as being positive (“POS”), negative (“NEG”), or neutral (“NEUT”), or as involving pleasure (“PLEAS”) or pain (“PAIN”). The following ANF statements represent feelings:

\(^{112}\) Emotion can be described as positive (“POS”) or negative (“NEG”). See Subsection 5, *infra*.

\(^{113}\) M. Dyer, *supra* note 19, at 117.
(?PERSON HAS (?FEELG POS));
(?PERSON HAS (?FEELG NEG));
(?PERSON HAS (?FEELG NEUT));
(?PERSON HAS (?FEELG PLEAS)); and
(?PERSON HAS (?FEELG PAIN)).

Manifestations of SOUL are somewhat loosely related in ANF. Feelings, emotions, and attitudes overlap to some extent. A feeling of pain, for example, may be intimately connected with an emotional state or attitude. It is possible for a person to feel pain yet have a positive attitude about feeling that pain. On the other hand, a person who feels pain may, as a result, experience negative attitudes and emotions. ANF allows multi-form representation in this area.

In ANF, a statement about feelings does not necessarily contain the directional expression TOWARD because sometimes feelings are not directed toward anything but are merely experienced. Thus, the statement:

(JOHN HAS (FEELG POS))

is a legitimate ANF expression even though it does not contain the directional expression TOWARD.

6. Representation of Mental Acts

Mental possessions ("MPs") are the objects of mental activities such as thinking and believing. Mental activities are described in ANF by statements in the form of:

(<subject> <mental-act> <object>).

The following ANF statements are instantiations of this structure:

(1) (JOHN BELIEVES (?MP));
(2) (JOHN INTENDS (?MP));
(3) (JOHN KNOWS (?MP));
(4) (JOHN CONSIDERS (?MP));
(5) (JOHN HOPES (?MP)); and
(6) (JOHN THINKS (?MP)).

The expression (?MP) in each of these statements represents a mental possession. For example, the (?MP) of statement (1) above represents a belief.
7. Intentions, Goals, and Decision Making

The actor’s intent is often necessary to determine the legal effect of a given act.\footnote{114} In the law of torts, for example, intentional acts may produce effects that differ from those produced by acts of negligence.\footnote{115} An actor who intentionally inflicts injury on another may face criminal charges in addition to the liability for damages that may be assessed against him in a civil action brought by the victim.

One of the most important legal distinctions drawn in the area of intention is the one between objective and subjective intent. Many legal effects are activated not by what an actor actually intends but by what the actor’s manifestations imply when viewed objectively.\footnote{116} In the contractual realm, for example, to determine whether an offer to enter into a bargain has been made, it must be determined whether the actor manifested a willingness to enter into a bargain and whether that manifestation was made so as to justify another person in understanding that his assent to the bargain was invited.\footnote{117}

ANF is flexible in its ability to represent mental possessions and acts. The natural language description,

John intends to go to the store

could be represented in ANF in a number of ways, among them:

\begin{enumerate}
\item (JOHN INTENDS (JOHN GOES-TO STORE\textsubscript{1}));
\item ((JOHN HAS INTENTION\textsubscript{2}) \hspace{1em} (INTENTION\textsubscript{2} = (JOHN GOES-TO STORE\textsubscript{1})));
\item (JOHN HAS (INTENTION\textsubscript{2} = JOHN GOES-TO STORE\textsubscript{1})).
\end{enumerate}

As in BORIS, intentions imply goals. Thus, to say that

(JOHN HAS INTENTION\textsubscript{2})

is to say:

\begin{itemize}
\item \footnote{114} See, e.g., W. Prosser \& P. Keeton, supra, note 95, at 33-39.
\item \footnote{115} Id. at 33-36.
\item \footnote{116} See A. Corbin, Corbin on Contracts §§105, 106, 538, 539 (1952).
\item \footnote{117} Restatement (Second) of Contracts § 24 (1979). For an overview of the problems one is likely to encounter when attempting to model the law of offer and acceptance, see Gardner, supra note 54, at 114-18.
\end{itemize}
(JOHN HAS GOAL$_3$).

A goal-achievement representation in ANF has three basic parts: (1) a description of a goal; (2) a description of an attempt to achieve the goal; and (3) an indication of success. The idea that John achieved the goal of being inside a house could be represented as follows:

(((JOHN HAS GOAL$_1$) (GOAL$_1$ = (JOHN IS (IN HOUSE$_2$)))) (JOHN ATTEMPTS GOAL$_1$) = ATTEMPT$_3$) (ATTEMPT$_3$ SUCCEEDS)).

A more detailed way of representing the same idea is:

(((JOHN INTENDS (JOHN CAUSES (JOHN IS (IN HOUSE$_2$)))) CAUSE (ATTEMPT$_3$)) (ATTEMPT$_3$ = (JOHN ATTEMPTS (JOHN CAUSES (JOHN IS (IN HOUSE$_2$)))))) (ATTEMPT$_3$ CAUSES (JOHN IS (IN HOUSE$_2$)))).

This represents the idea that John had an intention that caused him to attempt to carry out the intention and that his attempt was successful. The causal relation between the intention and the attempt can be explained by mental causality.

Actually, there is a missing link in the causal sequence described in this example. Between having a goal and actually attempting to achieve the goal is the act of deciding to engage in the attempt. The following is a representation of an act by which John wills an attempt to achieve GOAL$_1$:

(((JOHN HAS GOAL$_1$) (JOHN WILLS-ATTEMPT-AT GOAL$_1$) (JOHN ATTEMPTS GOAL$_1$) = ATTEMPT$_3$) (ATTEMPT$_3$ SUCCEEDS)).

To produce a more explicit representation of the idea, representations of the causal connections between the acts should be added. For example, there is a causal connection between John's willing the attempt and the actual attempt. The connection may be represented as follows:

118. See R. SCHANK & C. RIESBECK, supra note 94, at 28-29.
((JOHN WILLS-ATTEMPT-AT GOAL₁)
CAUSES (JOHN ATTEMPTS GOAL₁)).

A model of decision-making can be added to the goal achievement model. ANF decision-making is a process by which a person considers alternatives and decides on a goal or intention. The general formalism used for decision-making is:

((?SUBJECT CONSIDERS <alternative-list>)
(?SUBJECT DECIDES-UPON <alternative(s)>)).

John's decision to go to a movie instead of to a store could be represented as follows:

(((JOHN CONSIDERS
 (JOHN ATTENDS MOVIE₁) (JOHN GO-TO STORE₂))
 (JOHN DECIDES (JOHN ATTENDS MOVIE₁))).

C. Temporal Representation

The importance of the ability of a legally oriented language to represent time was recognized in the LEGOL project.¹¹⁹ Over the last decade, useful techniques for handling temporality have been developed by AI researchers.¹²⁰ Temporal calculi can be used to determine what events can take place within, before, or after any given time reference.¹²¹ In the natural language processing realm, the technique of using reference events¹²² and reference intervals¹²³ has proved to be a useful one. Networks of events¹²⁴ are established to formulate the basis for

¹²¹. Findler & Chen, supra note 120, at 165.
¹²². Hirschman, supra note 120, at 158.
¹²³. Allen, supra note 120, at 224.
¹²⁴. Hirschman, supra note 120, at 156-66.
incorporating new events into the overall temporal scheme. A reference interval\textsuperscript{125} may be defined for any given event to include all possible events to which the given event need be related. This technique seems particularly suitable for use in CCLIPS because CCLIPS uses reference intervals to define the scope of certain processes, such as the process of creation. Reference intervals may be changed incrementally\textsuperscript{126} to maintain an ongoing sense of "now" and to lessen the task of updating since only those events that occur within a given reference interval at any given point need be updated.\textsuperscript{127}

A system engaged in inferencing and matching can successfully employ temporal descriptions. For example, mutually exclusive temporal descriptions can disassociate incoming lexical items from other information in memory. Conversely, mutually exclusive conditions cannot be allowed to overlap in time.\textsuperscript{128} This kind of information can be used to maintain consistency in the knowledge base\textsuperscript{129} and to notify a user when an inconsistent fact is introduced.\textsuperscript{130}

For CCLIPS to be able to employ the aforementioned techniques and to acquire an in depth understanding of the texts it reads, it will have to be able to recognize and understand the following:

1. adverbial time expressions (e.g., "last week," or "yesterday");
2. verb tenses;
3. temporal connectives (e.g., "precedes," or "the result of");\textsuperscript{131}
4. changes in state (e.g., "Mary became well");
5. narrative time progressions (e.g., whether events in a narrative are concurrent or progressive);\textsuperscript{132} and
6. imprecise references (e.g., "a while ago," or "recently").

Accordingly, ANF has been designed to allow explicit representation of temporal information at the surface level. The following subsection describes a set of primitive temporal concepts that can be combined to form more complex notions of time. The concept of "yesterday," for example, is a complex notion because one must understand the concepts of "day," "now," "sequence," "before" (and therefore "after" also), and

\textsuperscript{125} Allen, supra note 120, at 223.
\textsuperscript{126} Id. at 224-25.
\textsuperscript{127} Id. at 222.
\textsuperscript{128} Long, supra note 120, at 254.
\textsuperscript{129} Allen, supra note 120, at 223; Kahn & Gorry, supra note 120, at 98-100; Malik & Binford, supra note 120, at 344.
\textsuperscript{130} Kahn & Gorry, supra note 120, at 98-100.
\textsuperscript{131} Bruce, A Model for Temporal References and Its Application in a Question-Answering Program, 3 Artificial Intelligence 1, 2-4 (1972).
\textsuperscript{132} Hirschman, supra note 120, at 164-66.
“consecutiveness” to be able to conceptualize the day that is sequentially before and consecutive to the current (“now”) day.

1. Temporal Features of ANF

CCLIPS automatically assigns temporal descriptions\textsuperscript{133} to entities and events when those descriptions are not given explicitly. In ANF, the expression TR is used to represent time references, including references to one or more points, intervals or chains. The expression:

OVER TR

when attached to an event description, signifies that the event occurred over the time reference TR. If the event consisted of subevents, the expression OVER TR implies that all the subevents occurred within the TR, but it does not imply that one or more of those subevents were in progress at each and every point within the TR. The expression:

FOR TR

indicates that the event to which the expression is attached was occurring at every point within the time reference TR.

Because events occur over time, CCLIPS generates an OVER TR description for every event unless another temporal description is given explicitly. Thus, the expression:

(PERSON ATE DINNER)

would be expanded to:

((PERSON ATE DINNER) OVER ?TR).

The expression ?TR stands for the time interval over which the eating took place, assuming of course that it took the person more than an instant to eat dinner.

\textsuperscript{133} ANF allows one to represent the following temporal concepts and relations at the surface level: points; intervals; chains; starting points of intervals; intersection of intervals; consecutive points, intervals, or chains; equal of points, intervals, or chains; disjointed points, intervals, or chains; and relations of inclusion and exclusion.
The letter T is used in ANF to refer to a point in time, whereas the letter I is used to refer to an interval of time. The expressions ST-PT and END-PT are used to represent the starting points and ending points of TRs. Relations between TRs are represented by temporal connectors. Some of the temporal connectors used in ANF are given below.

<table>
<thead>
<tr>
<th>Connector</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>before</td>
</tr>
<tr>
<td>A</td>
<td>after</td>
</tr>
<tr>
<td>C</td>
<td>consecutive</td>
</tr>
<tr>
<td>NC</td>
<td>not consecutive</td>
</tr>
<tr>
<td>IT</td>
<td>intersects/overlaps</td>
</tr>
<tr>
<td>BC</td>
<td>before and consecutive</td>
</tr>
<tr>
<td>AC</td>
<td>after and consecutive</td>
</tr>
<tr>
<td>BNC</td>
<td>before and not consecutive</td>
</tr>
<tr>
<td>ANC</td>
<td>after and not consecutive</td>
</tr>
</tbody>
</table>

In this notation, to say that two TRs are "consecutive" is to say that there are no intervening points between them and that they have no point(s) in common. As mentioned previously, the temporal description "OVER TR" is used to indicate that the event to which it pertains may not have been in progress at each and every point within the referenced TR. The representation:

\[(\text{JOHN INGESTED DINNER}_1) \text{ OVER TR}_{10}\]

indicates that at one or more points or intervals during "TR_{10}" it is valid to say that "JOHN" was ingesting dinner. It follows that "TR_{10}" contains at least one point or interval about which it can validly be said that "JOHN" was ingesting an increment of "DINNER_1" at that point or for that interval. Thus, the representation above is equivalent to the following:

\[(\text{(((JOHN INGESTED DINNER-INCREMENT}_2) \text{ FOR TR}_{8}) \ldots \ldots \ldots ((\text{JOHN INGESTED DINNER-INCREMENT}_n) \text{ FOR TR}_n)) \text{ (TR}_8 \text{ B TR}_n) \text{ (TR}_{10} \text{ INCLUDES (TR}_8 \text{ THROUGH TR}_n))}\]
(DINNER₁ CO
(DINNER-INCREMENT₂ THROUGH DINNER-INCREMENTₙ)))

The original representation has been transformed into a representation of subacts of ingestion. Each subact is represented to have taken place for an entire subinterval of time. The subscript "n" is used in ANF to represent the last of a possible series of acts or TRs.

Temporal descriptors were not included in most of the examples in the preceding sections of this article. For illustrative purposes, temporal descriptors are included in the following example. The natural language description for a person who walks to a store can be represented in ANF as:

((NOTCASE ((PERSON WAS (AT STORE)) AT Tₚ))
(PERSON (CAUSED BMO ((PERSON WALKED) OVER TRₚ))
 ((PERSON WAS (AT STORE)) (AT Tₜ FOR TRₜ))))
(Tₚ BNC Tₜ)
(Tₛ = ST-PT-TRₚ)
(END-PT-TRₚ BC TRₜ)
(ST-PT-TRₜ = Tₛ).

In this context, the expression AT flags the location of something unless it is immediately followed by a TR, in which case it introduces the TR that describes the time when something occurred. The representation above indicates that the PERSON was not at the store at a particular point in time (Tₛ), but that over an interval of time (TRₚ), the PERSON performed one or more acts of walking that caused the person to be at the store at point Tₜ. Given that the person was at the store at point Tₜ, it follows that the person was there for some amount of time [here represented by FOR TRₜ], whether that amount be equal to an interval or to an instant (in which case Tₜ would be equal to TRₜ). The point Tₛ, which represents a "reference point" at which the person was not at the store, is the starting point of the interval TRₚ (assuming of course that it took more than one instant to walk to the store). The end point of that interval of walking is described as being before and consecutive to the point at which John was actually at the store. This idea is represented by the expression:

(END-PT-TRₚ BC TRₜ)

where BC stands for "before and consecutive." The expression represents the idea that the end point of TRₚ is before and consecutive to TRₜ.
CCLIPS' temporal descriptions in the example above relate state and event descriptions using an interval calculus. By substituting new temporal descriptions, CCLIPS can incorporate alternative state and event descriptions to build an in-depth temporal scheme.

V. APPLICATION OF AI TECHNIQUES IN THE LEGAL DOMAIN

This Section describes how the foregoing techniques can be employed usefully in the legal domain. Subsection A discusses how the representational techniques described in this article can be used to build models of legal concepts. It also emphasizes that CCLIPS must interact with the user to solve difficult problems of text understanding. Subsection B discusses how conceptual models built from ANF expressions can be used to detect legal issues.

A. A. Building and Using ANF Models of Legal Concepts

ANF has many special features designed to make knowledge representation in the legal realm convenient. By Means Of ("BMO") clauses are used for causal explanation and expansion; the expression Consists Of ("CO") is used to identify the constituents of a thing or event. In Accordance With ("IAW") clauses define, govern, or qualify the terms they modify.

A model of a valid offer in contract law can be constructed using an IAW clause along with other representational devices. A test known as the "reasonable person test" is often used to determine whether a given manifestation is an offer.\textsuperscript{134} This test consists of determining whether a reasonable person would interpret the manifestation as an offer.\textsuperscript{135} To indicate in ANF that an act has been performed as a reasonable person would have performed it, the expression:

\begin{center}
IAW STAND-REAS-OBJ
\end{center}

is attached to the verb that represents the act. This expression indicates that the act was performed in accordance with the standards that determine reasonableness based on objective criteria. The expression:

\begin{center}
(MARY (BELIEVED IAW STAND-REAS-OBJ) (?BELIEF))
\end{center}

\textsuperscript{134} See A. CORBIN, supra note 116, § 11.
\textsuperscript{135} Id.
indicates that a reasonable person acting in place of Mary would have believed ?BELIEF. The following ANF expressions represent a situation in which an offer is made:

\[
((\text{PERSON}_x \text{ COMMUNICATES INFORMATION TO PERSON}_y) \\
\text{CAUSES } (\text{PERSON}_y (\text{BELIEVES IAW STAND-REAS-OBJ}) \\
\text{INFORMATION IS OFFER})).
\]

These expressions indicate that \text{PERSON}_y believed that the communication from \text{PERSON}_x was an offer.\textsuperscript{136}

The legal effects of a given act may depend upon the intent of the actor. Sometimes that intent must be determined objectively. The objective intent of a person is the intent implied by his actions.\textsuperscript{137} The ANF knowledge structure for \text{OBJECTIVE-INTENTION} is:

\[
\text{IF } ((?A \text{ MANIFESTS } ?\text{MANIFESTATION TO } ?B) \\
\text{CAUSES } (?B \text{ UNDERSTANDS IAW STAND-REAS-OBJ}) \\
(?\text{MANIFESTATION EVIDENCES } ?\text{INTENTION})) \\
\text{THEN } (?\text{MANIFESTATION (EVIDENCES (OBJECTIVELY))} \\
?\text{INTENTION})
\]

This simply means that if the manifestation by "?A" causes "?B," acting reasonably, to infer that "?A" has a particular intention, then from an objective point of view that intention is manifested.

An ANF legal concept model of the process of making an offer could be constructed from the following three representations:

\[
(1) (?)\text{INTENTION} = (?A \text{ INTENDS } (?A \text{ MAKES} \\
\text{BMO } ?A \text{ COMMUNICATES } ?\text{INFORMATION TO } ?B) \\
?\text{OFFER TO } ?B));
\]

\textsuperscript{136.} It is beyond the scope of this Article to discuss all the conceptual problems that could arise in the course of determining whether a given manifestation is an offer. Suppose that \text{PERSON}_y did not believe that the communication was an offer but that a reasonable person would have believed it to be such. How should the communication be classified in such a case? This raises the legal question whether a manifestation is an offer if the person who perceives the manifestation does not understand it to be an offer. Another question that could arise in some situations is whether a communication that otherwise qualifies as an offer should be held to be such if the person to whom the communication is directed never receives the communication. To avoid addressing problems of this sort, the discussions and examples in this Section cover only cases in which the communication is received and is interpreted by the recipient as an offer.

\textsuperscript{137.} See A. \textsc{Corbin}, \textit{supra} note 116, §§ 105, 106, 538, 539.
Representation (1) above defines \( \text{INTENTION} \) as an intent to make an offer. Representation (2) indicates that \( \text{A} \) acted on his intention by communicating \( \text{INFORMATION} \) to \( \text{B} \). Representation (3) indicates that the communication caused \( \text{B} \), acting objectively and reasonably, to construe it as an offer. These representations cover the cases in which \( \text{A} \) intends to make an offer. Because of the way the objective theory of contracts operates, however, it is possible for someone to make an offer without actually intending to do so by creating manifestations construed by a reasonable person to be an offer. In such a case, the person is said to have intended to make an offer from the objective point of view. Thus, representation (3) includes an objective element—

\[
(\text{UNDERSTANDS IAW STAND-REAS-OBJ})
\]

Representations may be accessed as knowledge structures to determine whether a given manifestation is an offer. Representation 3 may be converted to rule-form to read as follows:

\[
(\text{IF (}} (\text{A COMMUNICATES } \text{INFORMATION TO } \text{B}) \quad \text{CAUSES (}} \text{B (UNDERSTANDS IAW STAND-REAS-OBJ)} \quad (\text{INFORMATION IS OFFER})))
\]

The OFFER knowledge structure could be activated to help CCLIPS decide whether ANF input refers to an offer. If CCLIPS were to parse ANF input that contained the verb communicate, or a verb conceptually dependent upon communicate, it would activate the following knowledge structure:

\[
(\text{A COMMUNICATES } \text{INFORMATION TO } \text{B}).
\]

This knowledge structure would then be used to activate the OFFER knowledge structure based on the fact that it matches the first line of the
antecedent of OFFER. CCLIPS would then instantiate OFFER with the content of the ANF input. It is able to do this because the ANF input conforms to the structure of the activated knowledge structures since those knowledge structures are represented in ANF.

For CCLIPS to decide whether input is an offer, it would have to instantiate the input into the entire antecedent of the OFFER-knowledge structure. If the input completely fills the OFFER antecedent, CCLIPS could determine whether the input implied that someone reasonably and objectively believed that a manifestation was an offer.

Determining whether input constitutes an offer is an extremely difficult operation for CCLIPS. Ultimately, CCLIPS would have to turn the difficult decision-making tasks over to the user. The reason for this perhaps can be best understood by example. The example chosen deals with the problems associated with determining whether a given newspaper advertisement for the sale of an item is an offer rather than of some other type of communication, such as a price quotation.

The test to determine whether a given newspaper advertisement for the sale of one or more items constitutes an offer is whether the advertisement is "clear, definite, explicit," leaving "nothing open for negotiation." If so, the advertisement is an offer. A court deciding whether a newspaper advertisement is an offer may consider:

| ITEM-IDENTITY | Identification of the items offered for sale, |
| QUANTITY      | A specification of the quantity of the items offered for sale, |
| QUALITY       | A specification of the quality of the items offered for sale, |
| PRICE         | A specification of the price of the items offered for sale, |
| TIME          | A specification of when the items are to be offered for sale, |
| WHERE         | A specification of where the items are to be offered for sale, |
| OFFEROR       | A specification of who is offering the items for sale, and |
| OFFEREE       | An identification of the offeree. |

139. See id. § 25; Lefkowitz v. Great Minneapolis Surplus Store, 251 Minn. 188, 86 N.W.2d 689 (1957).
The following language, reportedly contained in a newspaper advertisement by a particular department store, was construed by the court to meet the test of an offer:

Saturday 9 A.M.

1 Black Lapin Stole
Beautiful
worth
$139.50

$1.00
First Come
First Served.140

If CCLIPS were to have knowledge structures that incorporated the factors listed above, it could perhaps decide whether particular language manifested an offer. It would do this by associating the content of the knowledge structure with the content of the advertisement. First, CCLIPS would have to recognize that the communication was a commercial advertisement. Next, it would have to associate the information of the knowledge structure (listed below in the column on the left) with the information in the advertisement (listed below in the column on the right).

| ITEM-IDENTITY: | Stole |
| QUANTITY:      | 1    |
| QUALITY:       | Black Lapin |
| PRICE:         | $1.00 |
| OFFEREE:       | First Come, First Served |

This top-down method of analysis would work fine as long as expressions like "first come, first served" could be recognized by the system. The problem with this approach is that even if the system were to find all the factors listed above in an advertisement, that in and of itself would not justify the conclusion that the advertisement is an offer. Those factors merely guide courts in making decisions about whether advertisements are offers. There are so many context-dependent factors used by courts that it would be infeasible even to attempt to list them all. Each case has its own particulars and presents its own difficulties.

For example, the problem of interpretation presented by the advertisement under consideration would have been more difficult had the

140. Lefkowitz, 251 Minn. at 189, 86 N.W.2d at 690.
advertisement read differently. The advertisement could have contained
the following additional language that might have changed its entire im-
port:

THIS IS NOT AN OFFER.

How this would affect the import of the advertisement would to some
extent depend on whether the language was conspicuous.

A slightly more difficult problem would have been presented had
the advertisement begun with the heading:

PRICE QUOTATION.

This is a typical problem that arises in the area of document interpret-
tion. The problem presents itself when the title of an instrument does
not accurately describe the contents of the body of the instrument.
What should control the import of the instrument in such a case—the ti-
tle or the body of the instrument? Generally, the title is considered to be
a mere factor to be weighed along with others to determine the nature
of the instrument.141

Different legal transactions may have factors in common. A lease,
for example, is similar to a sale in that one party promises some perfor-
mance in consideration for certain rights to real property.142 An instru-
ment labeled LEASE AGREEMENT could prove to be a sale instead of a
lease if the contents of the instrument indicate that the ownership of the
object passed from one party to the other for consideration. Similarly, if
the newspaper advertisement example had the heading PRICE QUOTA-
TION, the court would have to determine whether that was sufficient to
identify the advertisement as a mere quotation of prices rather than an
offer. In such a case, the content of the advertisement could be con-
structed to satisfy the criteria for an offer, yet the presence of the heading
PRICE QUOTATION adds weight to the possibility that the advertise-
ment is a mere price quotation.

As reflected in the OFFER knowledge structure, the basic test to
determine whether a communication is an offer depends on whether the
recipient, acting as a reasonable person, would construe the communi-
cation as such. A determination would have to be made about what effect
the heading PRICE QUOTATION would have on a reasonable person
reading the advertisement. Would the presence of that heading lead

142. Id. §§ 32-37.
that person to conclude that the advertisement is not an offer? As an alternative to answering such a question, CCLIPS would pose the question to the user.

B. Using ANF Legal Concept Models to Detect Legal Issues

Under the Articles of the Civil Code of Louisiana a simple error voids a contract if the error bears upon the "principal cause" of the contract. Essentially, the "cause" of a contract is the motive(s) for entering into the contract. Since "cause" is defined in terms of motive, the error must bear upon the principal motive for contracting. The essence of error under the Civil Code is mistaken belief; thus, an error as to the principal motive results from the motive being rooted in a mistaken belief. The PRIN-CAUSE link can be employed to build the following simple model of the principal cause or motive for contracting:

```
((?PERSON-X HAS ?MOTIVE)
 PRIN-CAUSE (?PERSON-X ENTER-INTO ?CONTRACT WITH
  ?PERSON-Z)).
```

The "principal cause" is that consideration without which the contract would not have been made. The PRIN-CAUSE link aptly accommodates this notion because it activates the following information:

```
(IF (NOTCASE (?PERSON-X HAS ?MOTIVE))
 THEN (NOTCASE (?PERSON-X ENTER-INTO ?CONTRACT WITH ?PERSON-Z))).
```

This means that if ?PERSON-X does not have ?MOTIVE, then ?PERSON-X does not contract with ?PERSON-Z. This follows because the PRIN-CAUSE link signifies that the ?MOTIVE is a substantial factor that brings about the act of contracting. The aspects of motivation that are relevant for the Louisiana law of error are represented below in ANF formalisms:

143. In the following discussions, motives are assumed to result from processes prompted by beliefs and desires. References to the processes that produce the motives have been omitted for simplicity.
145. Id. art. 1896.
146. Id. art. 1825, 1896.
147. Id. art. 1821.
148. Id. art. 1825.
(??PERSON DESIRES (?OBJECT-OF-DESIRE))
AND (?PERSON BELIEVES (?BELIEF)))
(CAUSE ENAB-CAUSE) (?PERSON HAS ?MOTIVE)).

These formalisms represent the idea that motivation is rooted in belief and desire.

This principal motive model is not adequate to model the law of error. The Civil Code implies that no error in the motive can invalidate a contract unless the other party knew or should have known that the motive was the principal motive for contracting. The expression (OBJECTIVELY) can be associated with the PRIN-CAUSE link so that together they imply that the PRIN-CAUSE connection would be detected by a reasonable person who would perceive the act involved. The expression (?A (PRIN-CAUSE (OBJECTIVELY)) ?B) thus implies that ?A is the principal cause of ?B from the objective point of view.

The legal concept model for error can now be constructed. The components of the model are presented below in segmented form.

(1) (((?PERSON-X DESIRES (?OBJECT-OF-DESIRE))
(?PERSON-X BELIEVES (?BELIEF)))
(CAUSES ENAB-CAUSES) (?PERSON-X HAS ?MOTIVE))
(2) (NOTCASE (?BELIEF IS TRUE))
(3.1) (((?PERSON-X HAS ?MOTIVE)
PRIN-CAUSE
(?PERSON-X ENTER-INTO ?CONTRACT WITH
?PERSON-Z))
and
(3.2.1) (?PERSON-Z KNOWS ((?PERSON-X HAS
?MOTIVE)
PRIN-CAUSE
(?PERSON-X ENTER-INTO ?CONTRACT WITH
?PERSON-Z)))

or

(3.2.2) ((?PERSON-X HAS ?MOTIVE)
(PRIN-CAUSE (OBJECTIVELY)) (?PERSON-X ENTER-INTO
?CONTRACT WITH ?PERSON-Z))).

The conjunction of segments (1), (2), and (3) constitutes the model for error. Segment (2) flags the point at which error in belief occurs.

149. Id. art. 1826 (objective test).
because it points out that the belief described in segment (1) is false. Segment (3.1) represents the idea that the ?MOTIVE is a principal motive from a subjective point of view. Segment (3.2.1) enables that motive to qualify legally as the principal cause because it indicates that ?PERSON-Z is aware of its status as such. Segment (3.2.2) takes care of the case in which ?PERSON-Z should have known that ?MOTIVE was the principal motive to contract. The causal link description in segment (3.2.2) contains the expression (OBJECTIVELY) and thus functions in CCLIPS to charge ?PERSON-Z with the knowledge that a reasonable person would have acquired in that contractual situation.

This legal concept model may be accessed as a knowledge structure to help detect legal issues posed by fact descriptions such as the following:

Mary wants to build a store on a lot owned by John and believes she would have the right to do so if she owned the lot. This serves as the primary motivation for her purchase of the lot from John.

This could be represented in ANF as:

\[
((\text{JOHN OWNS } \text{LOT}_2) \\
((\text{MARY DESIRES } (\text{MARY CONSTRUCTS } \text{STORE}_1 \text{ ON } \text{LOT}_2)) \\
(\text{MARY BELIEVES} \\
\quad (\text{IF } \text{MARY OWNS } \text{LOT}_2, \\
\quad \text{THEN } \text{MARY MAY CONSTRUCT } \text{STORE}_1 \text{ ON } \text{LOT}_2))) \\
\quad (\text{CAUSE ENAB-CAUSE} (\text{MARY HAS MOTIVE}_3)) \\
\quad ((\text{MARY HAS MOTIVE}_3) \\
\quad \text{PRIN-CAUSE } (\text{MARY ENTER-INTO CONTRACT}_4 \text{ WITH } \text{JOHN})) \\
\quad (\text{CONTRACT}_4 = (\text{MARY PURCHASE } \text{LOT}_2 \text{ FROM } \text{JOHN}))).
\]

If the facts include a zoning law prohibition against commercial construction, Mary’s belief would be false. If it was also a fact that John was aware of Mary’s belief and desire at the time of contracting (MOTIVE$_3$) the model on error could be used to detect the legal issues determining whether Mary could rescind the contract on account of error. The entire ANF fact description could be matched against the legal concept model (through instantiation).

The facts would match segments (1) and (2) of the model but would fail to fit segment (3). The facts indicate that John knew that Mary had MOTIVE$_3$, but segment (3) requires that John be apprised (directly or indirectly based on objective standards) of the PRIN-CAUSE
link of Mary’s motive to contract. The facts do not indicate explicitly whether John knew or should have known of that PRIN-CAUSE connection; therefore, the facts do not match segment (3). Thus, the crucial issue is whether John, as a reasonable person, should have known that Mary would not have entered into the contract had she known the true facts. To resolve that issue, it would have to be determined whether John reasonably could have believed that Mary would have purchased the lot anyway for some alternative reason(s).

**CONCLUSION: THE FUTURE OF CCLIPS**

The version of CCLIPS described in this Article is being designed to perform conceptual retrieval operations on the provisions of the Civil Code of Louisiana. Ultimately, the provisions of the Code will have to be represented in ANF, at least to some extent, if this goal is to be achieved. The Code contains 3556 articles, which is enough information to justify the use of conceptual retrieval techniques since it is difficult to keep particulars, details, and cross references in mind.

Detailed representation of the Code and the official comments in ANF would be difficult to achieve. Comments to articles are not considered primary sources of law; however, they are very important because they contain information that defines and otherwise relates the conceptual contents of the articles. Admittedly, portions of those comments would have to be represented in ANF for CCLIPS to perform conceptual retrieval of the sort envisioned by the authors, all of which points to a trade off between breadth and depth.

The solution seems to be to use ANF to describe the essentials of the Code rather than its details. Brief descriptions of the conceptual contents of articles and comments could be written in ANF and processed instead of, or in addition to, the Code itself. This approach would produce wider coverage but would sacrifice depth. Although the approach has its limitations, the authors believe it would produce useful results when used in conjunction with ordinary key-word retrieval techniques applied to the entire text. At the very least, the conceptual retrieval could reduce inefficient key-word results. This approach seems well suited to the version of CCLIPS that is described in this Article.

The long-term goals of the CCLIPS project are to enhance the system to accommodate an English-like version of ANF. The authors believe that once a more flexible version of ANF is available, it will become more feasible to consider writing both the statutory text and the comments in ANF. Success appears to be contingent not only on the availability of the more expressive version of ANF, but also upon the availability of a multiplicity of conceptual models of legal phenomena and real-world phenomena. The authors are currently working on models to
serve the system, some of which have been described in the foregoing pages.

The efforts of the authors to provide CCLIPS with an adequate collection of conceptual models hopefully will be complemented by research being conducted elsewhere. Interesting work is being done at the Center for the Study of Language and Information to create a common sense knowledge base that could serve to ground a number of different types of AI systems. Intelligent legal systems would benefit from having such a knowledge base because the knowledge would enhance a system's understanding of factual situations. Important work is also being done to model the deontic realm to enable systems like CCLIPS to handle problems posed by deontic operators such as "shall," "may," and "must," which frequently cause problems in the area of statutory interpretation. The progress made in the development of analogical and other reasoning methods is relevant to CCLIPS because its developers have set ambitious goals in this area. Advances in techniques by which conceptual memory is organized for retrieval will be particularly appreciated because of the sizable amount of conceptual information that CCLIPS must store in memory.

If CCLIPS reaches the level of development envisioned by the authors, an attempt will be made to empower it to process natural language input through interaction with the user. The goal would be to develop ANF so that natural language input could be digested without difficulty to accord with the basic conventions of ANF. Hopefully, a sufficient amount of conformity with those conventions could be produced to enable the system to clear up deviations through slight or moderate interaction with the user. The usefulness of the system, of course, would in large measure depend on how often the system would have to ask the user to clear up what it does not understand.

For some applications, quite a bit of interaction could be tolerated. Statutory drafting is an example of such an application. If the system were capable of acquiring an in-depth understanding of the information entered into it, it would be in a position to advise the drafter of inconsistencies and other deficiencies that might result from, or be included in, the information entered. Drafters are accustomed to taking considerable time to write even small pieces of legislation in attempts to come up with language that captures the legislators' intent. The process of drafting can become quite tedious; it is not unusual for numerous changes to be made in a proposed draft before a final draft is adopted. CCLIPS

---

could be useful in this situation despite its likely need to call for interaction.

In the future, CCLIPS will have many uses in the legal realm besides those associated with statutory drafting. Expert systems could be built on top of it, and its methods seem applicable to many other areas of the law. Some of the same techniques that work on statutes could be made to work on cases. There is, however, a need for systems to perform conceptual retrieval both inside and outside the legal profession. Any advances CCLIPS makes in this area would have immediate impact on many disciplines.

Artificial Intelligence applications such as CCLIPS have the potential to alter the nature of the legal profession. The more analytically integrated the domain, the better a system like CCLIPS will perform. Once AI systems begin to operate effectively in the legal domain, legal professionals may become motivated to be more analytical in their approaches to law.

Overall, the prospects for the future success of CCLIPS appear to be good at this stage of its development. Reports on improvements in the system will be published when appropriate. The authors invite comments from readers.\textsuperscript{151}

\textsuperscript{151} Correspondence should be addressed directly to Professor Cary G. deBessonet, Southern University, Louisiana State Law Institute, Paul M. Herbert Law Center, Room 382, University Station, Baton Rouge, Louisiana 70803-1016.
APPENDIX A: CAUSAL FORMALISMS

Every action verb defined in CCLIPS has an associated causal formalism that is activated when the verb is used.\textsuperscript{a1} For example, if the expression:

A LESSEE HELPED A LESSOR

was read, the causal formalism associated with "help" would be activated. An important segment of this formalism is presented below:

\[
(?A \text{ HELP } ?B) <==>
\]
\[
((?A \text{ CAUSE } (?B \text{ HAS } (?SOUL-ASPECT -> POS))))
\]
\[
\text{ OR }
\]
\[
((?B \text{ HAS } ?BURDEN) \text{ AT } ?TIME-1)
\]
\[
(?A \text{ CAUSE } ((\text{NOTCASE } (?B \text{ HAS } ?BURDEN)) \text{ AT } ?TIME-2))
\]
\[
\text{ OR }
\]
\[
(((\text{NOTCASE } (?B \text{ HAS } ?ENHANCEMENT)) \text{ AT } ?TIME-1)
\]
\[
(A \text{ CAUSE } ((?B \text{ HAS } ?ENHANCEMENT) \text{ AT } ?TIME-2)) ... ).
\]

This is only part of the formalism for "help." Only three senses of the verb "help" are described. The formalism, for example, does not provide for a case in which a negative aspect of the SOUL of "?B" is transformed into a non-negative aspect. In such a case "?B" would be "helped" by whatever caused the transformation.

In this formalism, the symbol "<==>" represents reverse implication and the expression "POS" represents positiveness. The symbols "?A" and "?B" are variables. The information to the right of the symbol "<==>" represents an expansion of the information to the left of the symbol. The expansion itself is also subject to expansion. The terms "BURDEN" and "ENHANCEMENT," for example, may be expanded to produce a new expansion. The entire scheme constitutes an abstraction/expansion hierarchy.\textsuperscript{a2} Higher-level representations are related to lower-level representations through "abstraction-expansion" relationships. In the formalism above, (\(?A \text{ HELP } ?B\)) is related to the stated expansion in this way. The information to the right of the symbol "<==>" represents the following three senses of the verb "help":

\textsuperscript{a1} See generally M. Dyer, supra note 19.

\textsuperscript{a2} The concept of an abstraction-expansion hierarchy is described in some detail in McCarty, supra note 11, at 272.
(1) (?A CAUSE (?B HAS (?SOUL-ASPECT -> POS)))

This represents a causal act in which "?A" causes "?B" to have some positive soul-aspect, such as a positive attribute or emotional state.

(2) (((?B HAS ?BURDEN) AT ?TIME-1) (?A CAUSE ((NOTCASE (?B HAS ?BURDEN)) AT ?TIME-2)))

This represents the removal of a burden. "?B" has the burden at one point in time, and "?A" causes that state of affairs to change at another, subsequent point in time. Here, the temporal sequence is indicated by the sequentiality of the numbers attached to the temporal descriptors.

(3) (((NOTCASE (?B HAS ?ENHANCEMENT)) AT ?TIME-1) (?A CAUSE ((?B HAS ?ENHANCEMENT) AT ?TIME-2)))

This represents the acquisition of an enhancement. Thus, the entire formalism represents senses of the word "help" in which "?A" helps "?B" by causing either:

(1) ?B to have a positive soul-aspect; or
(2) ?B to be relieved of a burden; or
(3) ?B to be enhanced.

The abstraction "(?A HELP ?B)" follows from any of these three senses by implication. The reverse implication, of course, does not hold because an abstraction defined disjunctively does not necessarily imply any particular disjunct.

In CCLIPS, the activation of a formalism, such as the one given above for "help," triggers other processes similar to the operation of demons in BORIS.\(^3\) Demons in BORIS perform specific tasks and fire when certain trigger conditions are met. BORIS uses two general types of demons: those that deal with knowledge-structure activation and those that help build episodic memory. The words in the lexicon of BORIS have demons attached to them so that when a word is recognized in parsing, it is placed in a working memory along with its associated demons. Disambiguation demons attempt to select an appropriate conceptualization for the parsed word, by activating CD structures.

---

\(^3\) For a more detailed discussion of the operation of demons and related processes in BORIS, see M. Dyer, supra note 19.
CCLIPS operates in a similar manner. If, for example, the formalism for "help" were to be activated during parsing, other task-oriented processes would also be activated to determine what sense of "help" was being referred to by the parsed word. These subprocesses would search working memory to see if some data structure could be found to match all or part of the formalism for "help." If the activated subprocesses were to find that whatever was in the position of "?B" had been described as having a burden immediately prior to being helped, this would activate the sense of "help" that is defined in terms of a burden being removed. In this regard, it can be seen that removing or releasing a burden presupposes a previous state in which the burden existed. A representation of the previous state is included in the formalism itself as:

\[(?B \text{ HAS } ?\text{BURDEN}) \text{ AT } ?\text{TIME}-1)\].

CD-based systems use similar techniques. AFFECTs in BORIS, for example, have standard relationships with one another. For example, if a character feels "relieved," then this points to some earlier state in which the character may have felt "fearful." Similarly, in CCLIPS, the acquisition of a thing presupposes a state of not having the thing.

Once a causal formalism has been activated, CCLIPS generates a possible inference (see Appendix D) to accompany the formalism. The kind of inferences generated depends in some measure upon the type of causal agent involved. The inferences generated when the causal agent is an entity differ from those generated when the causal agent is an event.

The following formalism represents the idea that something ("?A") caused something else ("?B") to exist:

\((?A \text{ CAUSE } (?B \text{ EXISTS}))\).

This formalism would activate the following inferences:

1. \((?A \text{ CAUSE } (?B \text{ EXISTS}))\)
2. \((?B \text{ EXISTS})\)
3. \((?A \text{ CREATE-CAUSE } (?B \text{ EXISTS}))\)
4. \((?A \text{ PROX-CAUSE } (?B \text{ EXISTS}))\)

The meanings of the causal links that appear in these inferences are described in Section IV.A.2 of this Article. Inference 1 is the original statement. A representation always implies itself. Inferences 2, 3, and 4
follow almost immediately from inference 1. These inferences could be activated if CCLIPS were to encounter the original causal structure.

**APPENDIX B: SYMBOLS AND EXPRESSIONS USED TO REPRESENT AGENTS AND END-STATES**

1. **Unary Entity ("UE")**

A UE is an individual entity defined by set criteria. The definition may or may not be explicit. Whenever reference is made to a single individual, the reference is to a UE. If reference is made to more than one individual, the reference is not to a UE but to a cluster-entity. Reference to a person named "John" is reference to a UE, but reference to John and Mary is reference to a CE (cluster-entity).

2. **Cluster-entity ("CE")**

Cluster representation is used in ANF for many reasons. Besides providing a convenient and concise means of referring to groups, CEs constitute useful lexicographical components. The noun "team," for example, could be defined as a CE with specified criteria for membership. If the word "team" were to be encountered in parsing, CCLIPS would activate all the attendant CE information.

A CE consists of a group of entities. The individuality of each member of the group is determined by particular criteria, and the group, in turn, is defined by more inclusive criteria. CEs may contain subclusters ("SUBCEs"). If one wished to refer to a group consisting of two male and two female persons, say Tom, Bob, Jill, and Mary, a CE representation could be used for that purpose. The expression "CE₁," for example, could be used to represent the group. SUBCEs could be then distinguished within CE₁. Tom and Bob could be identified as members of SUBCE₂, and Jill and Mary could be identified as members of SUBCE₃. Each of these SUBCEs would carry its own criteria for membership.

CEs are categorized under CE-ABSTRACTIONS that are activated any time an appropriately indexed CE is parsed. CE-ABSTRACTIONS can be very useful in processing statements. For example, a CE-ABSTRACTION would be useful to define how many members of a CE must participate in a given act before the act will be taken as an act of the CE. In some situations, a CE-ABSTRACTION that requires a quorum for CE action will be accessed to assist in determining whether action by members constitutes action of the CE itself. Sometimes one may wish to ignore individual members and to refer to the cluster as a block.
3. CE Taken as a Block ("CEB")

The primary reason for drawing a distinction between CEs and CEBs is that the inferences they activate differ. Individual members are ignored when a CEB representation is encountered. Thus, given the statement:

\[(CEB_1 \text{ HELPED JOHN}),\]

detailed inferences about the relation of each member of \(CEB_1\) to \(JOHN\) would not be activated as would have been the case if

\[(CE_1 \text{ HELPED JOHN})\]

had been given instead. Actually, a CEB is a special kind of CE. For example, the parsing of the statement:

\[(CE_1 \text{ HELPED JOHN})\]

would activate a number of knowledge structures associated with \(CE_1\), including its criteria for membership and the criteria to be used to determine when action by the CE implies action by its members. If some or all of this information had not been explicitly defined for \(CE_1\), the CE would expand into alternatives, one of which would be the block-unity sense of this CE. Assuming that \(CE_1\) had been defined to include members Tom, Bob, Jill, and Mary, a number of inferences would be generated for the statement:

\[(CE_1 \text{ HELPED JOHN}),\]

including:

1. \(CE_1\) helped John
2. \(CEB_1\) helped John
3. Tom helped John
4. Bob helped John
5. Jill helped John
6. Mary helped John
7. \(\{\text{Tom,Bob}\}\) helped John

\[\ldots\]
8. \(\{\text{Tom,Bob,Jill,Mary}\}\) helped John.

Inference (2) represents "CE\(_1\)" taken as a block.
It should be noted that this set of inferences may activate other processes within the system. One such process is the "multiple subjects classifier." The subjects of inferences 1 through 6 given above are singular, but the subjects of inferences 7 through 17 are multiple. Statements with multiple subjects or objects activate processes that attempt to determine the nature of the relations between the subjects or objects. In the legal domain, the nature of the relations must be determined because legal results may be dictated by the nature of the relations between subjects or objects.

4. Clusters of Acts ("CAVs," "CAVBs," "CAEs," and "CAEBs")

In addition to allowing explicit representation of groups of entities, ANF allows representation of groups of acts. The symbol "CAV" is used for this purpose. A particular act consisting of a set of subacts may be represented by a CAV. The inclusive act of studying, for example, may consist of subacts of reading and outlining material. The expression "CAV," for example, could be used to refer to subacts associated with the verb "study." The idea that a lessee improved the property under lease could be represented as follows:

\[
(\text{LESSEE CAV}_y \text{ LEASED-PROPERTY})
\]
\[
(\text{CAV}_y \text{ CO } (\text{LESSEE CLEANED LEASED-PROPERTY})
\]
\[
(\text{LESSEE PAINTED LEASED-PROPERTY}))).
\]

The symbol "CO" used in this representation means "consistency of."

If for some reason one wished to ignore the individual subacts, the expression "CAVB\(_y\)" could be used instead of "CAV\(_y\)" to establish block reference.

In ANF, one can define CAVs in several ways. In the example above, "CAV\(_y\)" is equated with descriptions of events. Another way to define CAVs is to focus solely upon descriptions of the acts involved. The following two definitions of "CAV\(_y\)" are examples of this kind of definition:

(1) (CAV\(_y\) CO ((CLEANED) (PAINTED))); and
(2) (CAV\(_y\) CO ((CLEANING) (PAINTING))).

On occasion, one may wish to refer to a set of acts that are not already grouped or otherwise defined. If one wished to refer to all the acts that caused the West to be won, for example, the expression "CAV\(_1\)" could be used for that purpose. Membership in CAV\(_1\) could be defined as follows:
CAV_1 == \{ACT_n | (ACT_n CONTRIB-CAUSE (WEST IS WON))\}

Thus, CAV_1 is defined to be the set of all ACT_n such that ACT_n contributed to the West being won.

When no criteria for membership in a CAV are set, the CAV refers to some set of unidentified acts, which in ANF would be represented by the expression "SOME-ACTS." For example, if no criteria for membership had been set for CAV_1, the expressions "CAV_1" and "SOME-ACTS_1" would be equivalent.

The act of studying (taken as a noun) can be represented in ANF by the expression CAE, which represents a cluster of acts that function in expression as an entity. The expression CAEB is used to represent block reference to such a cluster. Thus, the statement

IMPROVING THE PROPERTY UNDER LEASE MADE THE LESSEE TIRED,

could be represented in ANF as:

\[
((CAE_x CAUSED (LESSEE IS TIRED)) \\
(\text{CAE}_x INCLUDES ((\text{LESSEE CLEANED LEASED-PROPERTY}) \\
(\text{LESSEE PAINTED LEASED-PROPERTY})))).
\]

In this representation, the expression "INCLUDES" indicates that the listing is not necessarily exhaustive. If "CAE_x" were not defined, "CAE_x" would be equivalent to "SOME-ACTS_x." Thus, in such a case,

\[
\text{(CAE}_x \text{ CAUSED (LESSOR IS PLEASED))};
\]

is equivalent to:

\[
\text{(SOME-ACTS}_x \text{ CAUSED (LESSOR IS PLEASED))}.
\]

**APPENDIX C: ACTION FORMALISMS, BMOs AND PLANS**

BMO-clause representation can be employed when representing human interaction. Suppose one wished to represent the idea that John told Mary something that made her happy. The verb "tell" is defined in ANF as a type of oral communication. Some formalisms for oral communication are given below.
(1) (?SUBJECT (COMMUNICATES (BMO ?SUBJECT SPEAKS))
?INFORMATION TO ?RECEIVER)

(2) (?SUBJECT (CAUSES (BMO ?SUBJECT
COMMUNICATES
(BMO ?SUBJECT SPEAKS))))
(?RECEIVER HAS ?INFORMATION))

(3) ((?SUBJECT (CAUSES (BMO ?SUBJECT SPEAKS))
(?RECEIVER HAS ?COMMUNICATION))
(?COMMUNICATION = ?INFORMATION))

These formalisms have the following relationships to one another: Formalism 1 <-> Formalism 2 <-> Formalism 3. The reverse-directional arrows represent that the formalisms can be converted into one another in a straightforward manner. The formalisms are expansions of the following knowledge structure associated with the verb “tell”: 

(?SUBJECT TELLS ?INFORMATION TO ?RECEIVER).

Formalisms 1 and 2 define the verb “tell” in terms of the more primitive verb “communicate.” Formalism 3 uses the word “communication” instead of “communicate” and is designed to accommodate the noun that stands for an act of communicating. The idea expressed in EXAMPLE-2 that John made Mary happy by telling her he would consider selling her the lot could be represented by instantiating the past-tense version of the knowledge structure given above for “tell” and by combining the result with an instantiation of another general formalism used to represent states of arousal. The following formalism is one used to represent states of arousal:

(<subject> CAUSES (OBJECT IS <description of emotion>)).

The result of combining the two instantiations would read:

((JOHN TOLD INFORMATION\textsubscript{1} TO MARY)
(INFORMATION\textsubscript{1} CAUSED (MARY IS HAPPY))).\textsuperscript{c1}

\textsuperscript{c1. For the sake of simplicity, the content of John’s communication to Mary is represented by the symbol “INFORMATION\textsubscript{1}.”}
CCLIPS would expand this representation by parsing its contents into other associated knowledge structures, such as Formalisms 1, 2, and 3. CCLIPS would activate the formalisms through a conceptual dependency relationship that exists between the verbs "tell" and "communicate." To "tell" someone something is to "communicate" something to them. Since Mary is described as having a particular emotional state, a knowledge structure for emotional states would also be activated and instantiated. The result of the instantiation process would read:

\[
((\text{JOHN} \ (\text{CAUSES} \ (\text{BMO} \ \text{JOHN} \\
\quad (\text{COMMUNICATES} \ (\text{BMO} \ \text{JOHN} \ \text{SPEAKS}))))��)) \\
(\text{MARY} \ \text{HAS} \ \text{INFORMATION}_1) \\
((\text{MARY} \ \text{HAS} \ \text{INFORMATION}_1) \ \\
\text{CAUSES} \\
(\text{MARY} \ \text{HAS} \ ((\text{EMOT}_2 \ \text{POS}) = \text{HAPPY}))))
\]

This expanded version of the original ANF representation results from instantiating Formalism 2 and an appropriate knowledge structure for happiness.

BMO clauses are also used to define plans. The information in the BMO slot may be abstracted as the plan by which a particular goal is to be carried out. For example, in the representation:

\[
((\text{MARY} \ \text{HAS} \ \text{GOAL}_1) \\
(\text{GOAL}_1 = (\text{MARY} \ (\text{CAUSES} \ (\text{BMO} \ \text{MARY} \ \text{DRIVES} \ \text{CAR}))(\text{MARY} \ \text{IS} \\
\quad (\text{AT} \ \text{LOT}_2))))),
\]

the entailed BMO clause contains content that can be identified as the plan by which Mary intends to achieve her goal. In this example, she plans to achieve the goal by driving a car.

**APPENDIX D: INFERRENCE GENERATION**

CCLIPS generates inferences based upon the nature of the connections between events and upon the nature of the involvement of individuals in those events. When an individual does something that causes a state to exist or an event to occur, that individual has a particular relationship to each individual involved in the resulting state or event. The inferences that CCLIPS generates are based on those relationships.

Segments of the following examples are written in a special notation adopted solely to describe the theoretical foundation of the generation processes in CCLIPS. Unique numbers have been assigned to
causal links and verbs. The same has been done for state-description links (i.e. "HAS" and "IS"). Special connectors relate the subjects of the statements to the objects of those statements (thus flagging the nature of the relationships between those subjects and objects). These special connectors each bear the form:

\[(<E \text{ or } S> <\text{number}> <\text{s or o}>)\]

In this notation, 'E' stands for an event; 'S' stands for a state; the number in the number slot identifies the particular causal or state-description relationship; and an 's' or 'o' signifies a subject or an object of the state or event. Whenever a connector that bears this special form appears between a subject component and an object component (or an object and a subject) the E or S describes the nature of the relationship between the two components; the number identifies the particular relationship; and the s or o identifies the first component as either a subject or an object. The first component is always taken to be the primary component of the entire representation. For example, with reference to the state description "(?C HAS1 ?D)," the special representation "(?D S1o ?C)" carries the following meaning:

(a) "?D" is the primary component of this special representation. By stipulation, the presence of the special connector "S1o" makes the first component of the representation (i.e. "?D") function as the primary component.\(^d1\)

(b) "?D" is related to "?C" in a state description in which "?D" is the object. The letter "S" of the connector "S1o" identifies the nature of the description as being a state description. The letter "o" indicates that "?D" is the object in the referred to state description.

(c) The state description in which "?D" is related to "?C" is identified by a state-description link (in this case "HAS1") that bears the number 1. This is indicated by the presence of the number 1 in the connector "S1o."

The special notation described above is used in some of the examples that follow.

---

\(^d1\) For an example of why it is useful to treat the first component as the primary component of the entire representation, see the discussion that accompanies inference (7) of Example A, infra.
Example A

Inferences could be generated for the formalism

\[(?A \text{ CAUSE2 } ?B) \text{ CAUSE1 } (?C \text{ HAS1 } ?D)\]

based upon the following information, which consists of deductions and abstractions drawn from the formalism.

(1) \((?C \text{ HAS1 } ?D)\)

This follows from the formalism by deduction.

(2) \((?C \text{ S1s } ?D)\)

This is an abstract deduction drawn from the formalism. It is referred to as an "abstract" deduction because it consists of an abstract description of an inference that follows from the original formalism by deduction. In other words, inference (2) is an abstract description of "\((?C \text{ HAS1 } ?D)\)," which follows from the formalism by deduction. The "S" of "S1s" indicates that "?C" is the primary component of this inference and that "?C" is related to "?D" in a state description. The number 1 of "S1s" ties the state description to the link "HAS1," and the letter "s" indicates that "?C" occupies the subject position in the referred to state description.

Another example is:

(3) \((?D \text{ S1o } ?C)\)

which is an abstract deduction drawn from the formalism. The expression "S1" of "S1o" has the same meaning as defined in (2) above. The letter "o" of "S1o" indicates that "?D" is the primary component of this inference and that "?D" occupies the object position in the referred to state description.

(4) \((?A \text{ CAUSE2 } ?B)\)

This follows from the original formalism by deduction.

(5) \((?A \text{ E2s } ?B)\)
This is an abstract deduction drawn from the formalism. It indicates that "?A" and "?B" are related to one another in an event description whose causal link bears the number 2. As is usual for this notation, "?A," being in the subject position in this inference, is taken to be the primary component of the representation.

(6) (?B E2o ?A)

This is an abstract deduction drawn from the formalism. It has meaning similar to representation (5) above except that "?B" is taken to be the primary component. The letter "o" indicates that "?B" occupies the object position in the referred to event description.

(7) ((?A CAUSE2 ?B) CAUSE1 (?C S1s ?D))

This odd looking representation, which follows from the original formalism only in a penumbral sense, relates the causal event "(?A CAUSE2 ?B)" to "?C." This is accomplished by treating "?C" as the primary component of the representation "(?C S1s ?D)," an effect dictated by the presence of the special connector "S1s." To say that

((?A CAUSE2 ?B) CAUSE1 (?C S1s ?D))

is to say that "?C" was caused, since "?C" is taken to be the primary component of the end-state caused by "(?A CAUSE2 ?B)." This inference is, at best, penumbral because the original statement does not imply that "?C" itself was caused, but rather that a state of "?C" was caused. An inference of this sort is hereinafter referred to as a penumbral inference. The meaning of representation (7) above is that the event "(?A CAUSE2 ?B)" is related to "?C" in some way through "CAUSE1." If the system had this information in memory, it would be in a position to recognize that it had a near match for the input "((?A CAUSE ?B) CAUSE ?C)."

The following examples (8 through 14) are all penumbral inferences.

(8) ((?A CAUSE2 ?B) CAUSE1 (?D S1o ?C))

The meaning of this expression is similar to that of representation (7) above except that here "?D" is the primary component of the end-state "(?D S1o ?C)" and is identified as being the object of the state description indexed by the expression "S1" of "S1o."
(9) ((?A E2s ?B) MAYBE-CAUSE1 (?C HAS1 ?D))

The expression "E2" of "E2s" indicates that "?A" is related to "?B" in an event and that the event is identified by a link that bears the number 2. The "s" of "E2s" indicates that "?A" is in the subject position in the referenced event. The presence of the special connector "E2s" makes "?A" the primary component of the representation "(A E2s ?B)"; thus, inference (9) relates "?A" to the end-state "(?C HAS1 ?D)" through the link "MAYBE-CAUSE1," which carries the number of the link from which it is derived. In this example, the MAYBE-CAUSE link is derived from the link "CAUSE1" because the link "CAUSE1" relates the event in which "?A" participates (i.e. "(?A CAUSE2 ?B)"); to the end-state "(?C HAS1 ?D)." Although weaker than a direct causal link, the MAYBE-CAUSE link is stronger than some other causal links, such as the POSS-CAUSE link. The strength of the MAYBE-CAUSE link in this example is derived from the fact that "?A" occupies the position of agent in a segment of the causal structure and thus, in some sense, "causes" the results that are represented further down the causal structure.

(10) ((?A E2s ?B) POSS-CAUSE1 (?C S1s ?D))

This inference weakly relates "?A" to "?C" since "?A" is the primary component of the subject of the link "POSS-CAUSE1," and "?C" is the primary component of the object of that link. This entire inference represents the idea that it may be valid to infer that "?A," by participating in the event indexed by "E2," caused (in a weak penumbral sense) "?C." As is the case for the MAYBE-CAUSE link (see discussion of inference (9) above), the POSS-CAUSE link bears the number of the link from which it was derived, which in this example is the link "CAUSE1."

(11) ((?A E2s ?B) POSS-CAUSE1 (?D S1o ?C))

This means the same as inference (10) above except that "?D" is the primary component of the object of the link "POSS-CAUSE1."

(12) ((?B E2o ?A) MAYBE-CAUSE1 (?C HAS1 ?D))

This means the same as inference (9) above except that "?B" is the primary component of the subject of the link "MAYBE-CAUSE1."

(13) ((?B E2o ?A) POSS-CAUSE1 (?C S1s ?D))
This means the same as that of inference 10 above except that "?B" is the primary component of the subject of the link "POSS-CAUSE1."

(14) (?B E2o ?A) POSS-CAUSE1 (?D S1o ?C)

This means the same as that for inference 11 above except that "?B" is the primary component of the subject of the link "POSS-CAUSE1."

Example B

CCLIPS is capable of generating deductive and penumbral inferences for each causal structure it encounters. A description of how penumbral inferences are used appears below.

Given the structure:

((?A CAUSE (?B HAS ?C)) CAUSE (?E IS ?H))

CCLIPS could generate the following knowledge structures (penumbral inferences) for that structure:

(1) (?A POSS-CAUSE ?B);
(2) (?A POSS-CAUSE ?C);
(3) (?A POSS-CAUSE ?E);
(4) (?A POSS-CAUSE ?H);
(5) (?A MAYBE-CAUSE (?E IS ?H));
(6) (?B POSS-CAUSE ?E);
(7) (?B POSS-CAUSE ?H);
(8) (?B MAYBE-CAUSE (?E IS ?H));
(9) (?C POSS-CAUSE ?E);
(10) (?C POSS-CAUSE ?H);
(11) (?C MAYBE-CAUSE (?E IS ?H));
(12) ((?B HAS ?C) POSS-CAUSE ?E);
(13) ((?B HAS ?C) POSS-CAUSE ?H);
(14) ((?B HAS ?C) MAYBE-CAUSE (?E IS ?H));
(15) ((?A CAUSE (?B HAS ?C)) POSS-CAUSE ?E);

CCLIPS uses inference-knowledge structures such as these when it attempts to answer queries posed by the user. The following example illustrates the use of penumbral inferences. The natural language description:
a lessee improved the leased property, which pleased the lessor

may be represented in ANF as:

\[((\text{LESSEE CAUSED} (\text{LEASED-PROPERTY HAS IMPROVEMENTS})) \text{CAUSED} (\text{LESSOR IS PLEASED})).\]

This ANF version differs slightly from that previously given for EXAMPLE-1; however, the reader should not be troubled by this because the two representations are equivalent in CCLIPS.

It should be noted that this ANF representation is an instantiation of the following causal structure:

\[((?A \text{ CAUSE} (?B \text{ HAS} ?C) \text{ CAUSE} (?E \text{ IS} ?H)) d^2\]

This is the same structure that was used previously to show which penumbral inferences would be generated for such a structure. Penumbral inferences could be generated for the ANF representation above by parsing its components into the inference-knowledge structures associated with this causal structure.

Suppose the user wanted to know whether or not it was the improvements that made the lessor happy. How should the system respond to such a query? The query could be represented in ANF as:

\[(\text{IMPROVEMENTS CAUSE} (\text{LESSOR IS PLEASED})?\]

Technically read, neither the natural language description nor its ANF counterpart implies that the improvements were the sole cause of the lessor's pleased state. Each of those descriptions implies that an entire event pleased the lessor. The improvements were merely involved in that event. It is possible, for example, that the lessor would not have been pleased had the improvements been made by someone other than the lessee. For one to be able to deduce that the improvements were the cause of the lessor's emotional state, a different causal structure would have to be present, such as:

\[((?A \text{ CAUSE} (?B \text{ HAS} ?C))
((?B \text{ HAS} ?C) \text{ CAUSE} (?D \text{ IS} ?E)))

\text{d2. Note that differences in verb tense are ignored in this example.}\]
In other words, if the ANF version had read:

```
((LESSEE CAUSED (LEASED-PROPERTY HAS IMPROVEMENTS))
 ((LEASED-PROPERTY HAS IMPROVEMENTS)
  (CAUSED
   (LESSOR IS PLEASED))))
```

the appropriate causal structure would have been present, and CCLIPS could have deduced that the improvements caused the emotional state of the lessor. The original ANF representation, however, does not bear this structure; therefore, the conclusion could not be reached by deduction. However, CCLIPS could draw a valid penumbral inference that the improvements caused the lessor’s state. It would do this by instantiating inference-knowledge structure (11) in Example A, supra. The result would read:

```
(IMPROVEMENTS MAYBE-CAUSE (LESSOR IS PLEASED)).
```

By making minor adjustments in verb tense and by recognizing that the link “MAYBE-CAUSE” is a penumbral derivative of the link “CAUSE,” which is the present-tense version of the causal link of the query, CCLIPS could match the inference with the query and thus would be in a position to respond intelligently. Although it could not give a valid affirmative response, it could inform the user that an event involving improvements pleased the lessor, which is precisely the idea that serves as the foundation for the penumbral inference.