Cases and Rules: A Natural Synergy

Edwina L. Rissland

Department of Computer Science
University of Massachusetts
Amherst, MA, USA 01003
rissland@cs.umass.edu

Abstract

This paper discusses the use of case-based-reasoning (CBR) in concert with other reasoning paradigms, particularly rule-based reasoning (RBR), and how these ideas have been embodied in the hybrid system CABARET (CAse-BAsed REasoning Tool), built in the CBR Lab at the University of Massachusetts. CABARET combines case-based reasoning and rule-based reasoning in a dynamic way using an agenda-based controller to address the task of statutory interpretation.

1. Introduction

In many disciplines, such as law, medicine, mathematics, there is a natural and complementary relation between the rules and cases of the domain. Each enhances the reasoning done with the other, and in many cases, makes up for deficiencies in the other mode of reasoning. In this paper we discuss CABARET, a hybrid case-based and rule-based system. A fuller discussion can be found in [Rissland & Skalak, 1991]. The theory of statutory interpretation that underlies much of CABARET's approach is discussed in [Skalak & Rissland, 1992]. Another hybrid approach from the UMass CBR Lab is FRANK [Rissland et al., 1993]. Another blackboard-based CBR system is BROADWAY [Skalak, 1992].

There are many reasons for the synergistic relation between reasoning with cases and reasoning with rules:

- Cases help ameliorate deficiencies in rule-based reasoning, such as the absence of well-defined predicates and hard-and-fast rules.
- Rules can help index cases and otherwise focus case-based reasoning.
- Cases permit incremental learning that does not involve a major rewrite of the rules, for instance, to account for emergent exceptions.
- Rules can summarize truths and heuristics induced or gleaned from cases.
- Cases make a system more robust in the face of well-known problems, such as drift in the meaning of terms.
- Cases help reduce the knowledge acquisition bottleneck by allowing certain forms of episodic knowledge or experience to be captured in "snapshot" form without the need for reducing this knowledge to rules or models.

The architecture of CABARET allows rules and cases to complement and supplement each other in many of the ways just listed. It does so in an opportunistic and dynamic manner, which distinguishes it from other hybrid systems, where, for instance, CBR is only invoked when RBR reaches an impasse of some kind (e.g., Grebe [Branting, 1991] and Gardner's system [Gardner, 1987]). In its opportunism it is similar to other hybrid systems based on a blackboard architecture (e.g., Prolexs [Walker et al., 1991] and FRANK [Rissland et al., 1993]). CABARET is, in effect, a simple blackboard system, in which control of the system is accomplished through one central, control knowledge-source, a rule-based, agenda mechanism.

CABARET uses classic forward and backward chaining rule-based techniques—with one major modification—for its RBR component. (The RBR component requires mechanisms to recognize "near miss" situations, that is, situations, in which all but one rule antecedent has been satisfied.) CABARET uses a HYPO-style case-based reasoner for its CBR component [Ashley, 1990]. CABARET's CBR component uses standard HYPO mechanisms, such as indexing based on factors ("dimensions"), sorting of cases according to a most on-point ordering, and selection of "best" cases using a "claim lattice." CABARET uses a standard agenda-based controller, in which the creation and selection of tasks is governed by heuristic control rules.

The heuristic rules used to manipulate the agenda—post, assign priorities, and select items—embodies CABARET's approach to statutory interpretation. The control rules guide CABARET in its tasks of gathering and organizing information for

hybrid case-and-rule explanations and arguments. Its use of heuristic control rules is similar to those in Lenat's classic AM system [Davis & Lenat, 1982]. However, where Lenat's rules, in effect, embodied a theory of scientific and mathematical discovery, CABARET's rules embody a theory of statutory interpretation and argumentation.

Even though CABARET is instantiated in an area of United States statutory law—the so-called *home office deduction*, defined in Section 280A(c)(1) of the Internal Revenue Code—the architecture of CABARET is perfectly general: it only requires that there be rules and cases. In fact, one could employ another style of CBR than the one we have used or a different set of control rules that embody an approach or theory for a different reasoning task, for instance, advice-giving. Of course, the domain-specific aspects—cases, indices, rules, predicates, etc.—could also be changed for a different application domain.

2. Statutory Interpretation

2.1 Weak and Strong domains: the centrality of CBR

Many domains, such as the law, are "weak" domains in the sense that their rules and models do not or cannot predict outcomes with absolute certainty. Such domains require the use of both cases and rules in complementary and supplementary ways [Twining & Miers, 1982; Morris, 1937]. For instance, cases are needed to overcome impasses in reasoning—unresolved rule predicates, unsatisfied antecedents, etc.—and to doublecheck or "calibrate" the results of RBR—finding exceptions that have arisen in specific cases, checking outcomes on similar cases, etc.

Other domains, such as mathematics, are "strong" domains in the sense that their rules and models (e.g., theorems) are unassailable: if a theorem implies a certain conclusion, then it is indeed true, now and forever, *saecula saeculorum*, without exception. In such domains, cases—or "examples" as they are usually called in mathematics—are not strictly necessary (in the mathematical deductive sense) but they are exceedingly useful and central to carrying out mathematical tasks: discovering proofs, posing conjectures, refining hypotheses, illustrating ideas, etc. [Polya, 1957; Lakatos, 1976; Rissland, 1978; Davis & Lenat, 1982]. In computational terms, cases help focus the attention and resources of the problem solver.

Cases play a central role across the entire spectrum of domains and reasoning tasks. Cases are obligatory in weak domains and exceedingly helpful in strong domains, even on deductive tasks. However, since most practical domains, including the law, are weak, the necessity of reasoning with cases is inescapable, no matter what one's opinion about their status in strong domains. Since most domains require harnessing all the knowledge and capabilities one can, hybrid systems—combining case-based, rule-based, model-based, and other reasoning methods—are a natural approach.

2.1 The Problem of Statutory Interpretation

In the law, "statutory interpretation" refers to the process of determining the meaning of a legal rule, including the meaning of its constituent terns, and then applying it to a particular set of facts [Levi, 1949; Llewellyn, 1989; Twining & Miers, 1982]. Fundamental difficulties that can arise in this process include:

- Terms are not sufficiently well-defined to enable straightforward application; many suffer from what is known as "open texture."
- Rules have uncodified exceptions and additional requirements making them impossible to apply in a mechanical way.
- Both terms and rules change in meaning and scope over time, for instance, before new rules can be drafted or old rules amended.
- Certain terms are deliberately left vague, for instance, to allow for flexibility.
- Concepts implicitly defined by a rule can be defined differently in alternative rules.
- There are often at least two competing interpretations.

All statutory domains engender such interpretation problems, even those whose statutes have been written with the greatest care and vision, such as tax, commercial, and bankruptcy law. Such difficulties are inevitable, and not necessarily the result of faulty draftsmanship or legislative short-sightedness.

2.1 An Example

The challenges presented in statutory interpretation can be illustrated in a very simple example. Suppose a teacher has announced the following rule to a class:

If a paper is late, then it is not be acceptable.

Such a rule makes much sense. However, as experience grows, situations arise where applying the rule to particular situations is not so facile. For instance, consider the following typical situations:

1. A paper is submitted a week late by a student who has been in the hospital.

2. An excellent paper is submitted a day late.

3. A paper is submitted 3 days late by a student who has a written excuse from the Dean.

Such fact situations test the mechanical application of the rule. If one's goal as a teacher is to be fair, inspire students, enable them to learn as much as possible, etc., then rote application of the rule might not be the best approach. These cases raise the need for exceptions. For instance, is one day late really so "late" as to disqualify a paper? Can a one day grace period be allowed for "excellent" papers; how is "excellence" to be defined? Are there no exceptions to be made for mitigating circumstances, such as sickness? Note, interpretation problems can arise with few cases, even the very first one.

Also note that there can be cases that should allow easy interpretations but in fact raise the possibilities of tacit requirements in the rule:

4. A paper is submitted on-time but is totally illegible.

5. A paper is submitted 3 days early but is only two sentences in length.

6. An on-time paper is a copy of a paper written by someone else.

These cases are "hard" in the sense that they test the meaning (and intent) of the rule and its constituent terms [Hart, 1958, 1961; Fuller, 1958]. A given paper may not be clearly "in" or "out" of the class of "acceptable" papers. There is no hard-edged black-and-white boundary between acceptable and non-acceptable papers; rather there is a gray penumbra in which classification is debatable. Of course, some cases are "easy." There are even "prototypes" of clearly acceptable and non-acceptable papers in the landscape of instances [Rosch & Mervis, 1975; McCarty & Sridharan, 1981; Bareiss, 1989]. For example, not many are likely to argue about an original, legible, proper length, on-time paper being acceptable or a month-late unexcused submission or a plagiarized paper being not acceptable.

Of course, one can try to cope with deficiencies in a rule or its terms by rewriting it or adding new ones:

If a paper is late and there are no mitigating circumstances then it is not acceptable.

If a paper is submitted more than 24 hours after the deadline then it is late.

And one can split the process of grading from the process of accepting papers for grading, define minimal standards for style, etc. Regardless, however, of how carefully one engages in legislative draftsmanship, cases will arise that will test the amended rules as well. In fact, all the problems of interpretation will be replayed on the revised rule set. (E.g., What are "mitigating circumstances?" Are there exceptions to the 24-hour rule?) No amount of fiddling with the rules and constituent terms will make them impervious to challenges posed by cases thrown up from the world, even if one relentlessly rewrites the rules to handle each new challenge. No amount of definitional backchaining will fix all the problems in the meaning of terms [Waterman & Peterson, 1981; Sergot et al., 1986; Berman, 1989]. As Gardner puts it, the rules will "run out" before interpretation problems are solved [Gardner, 1987].

In effect the meanings of rules and terms emerge only through application and cannot be captured by rules alone (in non-mathematical domains). The rules and terms plus the case base of situations in which they have been interpreted define their meanings. Note in a common law domain, cases carry the lion's share of the burden since there are no statutory rules. In a statutory domain, the burden is joint.

Of course, a decision-maker (judge, administrator, teacher) tries to be as consistent as possible. In fact, that is the mandate of *stare decisis*—the doctrine of precedent—at the heart of Anglo-American law. It requires that similar cases be decided similarly. The doctrine of precedent acts as a constraint on decision-making. However, since the meaning of "similarity" itself is open-textured, this constraint is no enforcer of absolute consistency, and it was not meant to be. No matter how hard one tries to avoid it, the time will come that a case will present a set of facts that challenge the consistency of the case base.

In particular, meanings tend to drift over time as more cases are considered. Eventually there might be so much drift as to cause once easily decided cases to become hard, or vice versa. In our example, a paper that might have been clearly accepted at one time might no longer be so, and vice versa. This is because there is learning and accommodation (by both students and teacher) to the changing world.

Much of the learning is spurred on by cases, and cases capture much of what is learned. In this way, the law is an adaptive system (just as is a teacher plus a class) that learns through doing. Without such learning, the system would be hopelessly brittle. Cases offer one way to import some critically needed learning into a problem-solving system.

In summary, statutory interpretation—especially as one makes exceptions to rules or refines the meaning of terms—can never be a simple, rote mechanical process. Mechanical jurisprudence is simply not viable. Statutory interpretation requires that cases and rules be taken together.

3. The CABARET Architecture

CABARET uses a domain-independent architecture to integrate CBR and RBR. See Figure 1. It has the following major components:

1. Two primary **co-reasoners**: a HYPO-style case-based reasoner and a traditional rule-based reasoner. Each is capable of being run in a stand-alone manner.

2. For each reasoner, there is a dedicated **monitor** that makes observations on the processing, results, and partial results of its assigned reasoner, and recasts these observations in a language (the "control description language") understandable to the controller.

3. A **controller** that uses the observations harvested by the two monitors to post and select tasks to be acted on by the individual co-reasoners. An agenda is used to keep track of the tasks, which are suggested by heuristic control rules.

The overall cycle of processing is that one of the co-reasoners works on a task. The monitor modules make observations. Based on the observations, the controller posts new tasks and selects a task to be worked on next by one of the co-reasoners.

The overall behavior of CABARET is:

1. A **user inputs** a problem case (the "current fact situation" or **cfs**), a point of view (pro or con a particular interpretation), and an overall goal (e.g., produce an argument).

2. **CABARET analyzes** the current fact situation by using rules and cases. It opportunistically pursues various case-based and rule-based tasks, which reflect the facts, point-of-view, intermediate results, overall goals, and CABARET's strategies and tactics for statutory interpretation.

3. **CABARET outputs** an argument or explanation in the form of a structured memo as to why a certain interpretation should or should not hold, complete with case-based and rule-based support pro and con.

CABARET does not have any natural language capabilities. Case entry is essentially a slot-filling activity carried out by the user. Memo creation is essentially a template-filling activity carried out by CABARET.

The primary domain-specific knowledge sources used in CABARET are:

1. A **case knowledge base** (CKB) and an **index knowledge base** (IKB). Cases encode facts and outcomes in a classic frame-based approach. Indices (known as "dimensions" or "factors") are encoded in terms of "factual predicates" that can be computed from information in a case.

2. A **rule-base** encoded in terms of the same factual predicates.

3. **General domain knowledge**, especially hierarchies, available to all modules.

The other major source of knowledge in CABARET is its set of domain-independent **control rules**, encoded in the control description language. These rules suggest and order tasks based on the observations made by the monitors. They embody CABARET's theory of statutory interpretation.

The primary repositories of information developed during processing are:

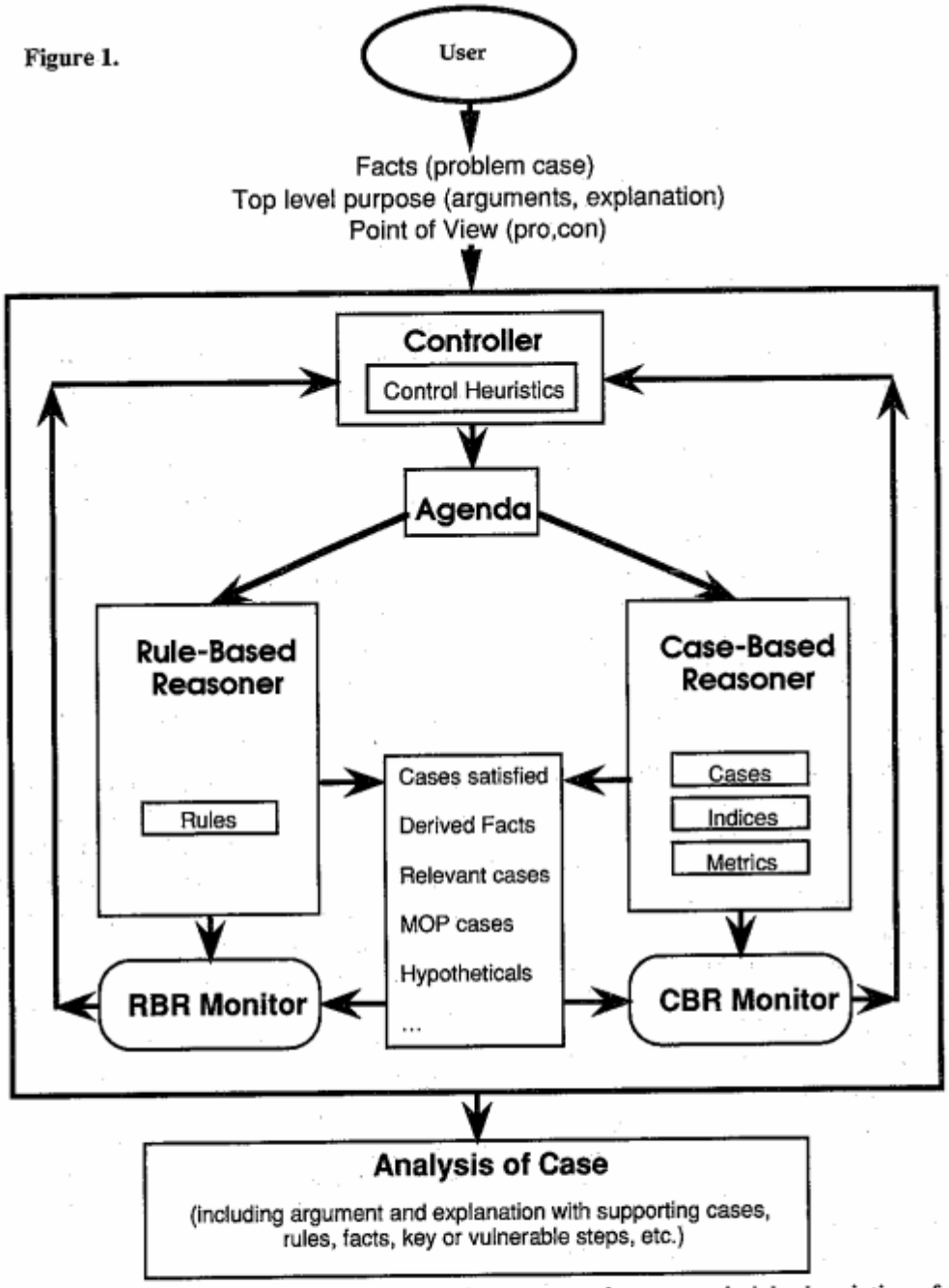
1. The **CBR Report**—a data structure containing descriptions of intermediate and final results of the CBR module.

2. The **RBR Report**—a data structure containing descriptions of intermediate and final results of the RBR module.

3. The **Control Report**—observations made by the monitors, tasks posted and acted upon, etc.

Note that CABARET was designed to serve two purposes. In addition to providing an environment in which to experiment with hybrid CBR-RBR, CABARET was designed to serve as a shell for building such systems. In particular, CABARET provided tools for building, modifying, and maintaining the basic components of a CBR system, such as case-bases, libraries of indices, and different similarity metrics.

Figure 1.



3.1 HYPO-style CBR—a brief review.

The three fundamental components of a HYPO-style CBR system are:

1. A case-knowledge base of cases;
2. A index knowledge base of "dimensions";
3. A similarity metric to assess "on-pointness."

The CKB consists of cases represented as fact situations plus outcomes. They are encoded as hierarchical sets of frames in a classic frame-based manner. The structure of the CKB is flat: no processing relies on explicit links between cases. (Cf, other systems, such as BankXX [Rissland et al., 1994] which do.) Cases encode the facts of the case: the taxpayer, the taxpayer's job, description of the home office, description of the taxpayer's business activities, including hours spent in the office, income derived from activities performed there, the nature of activities, etc. Cases also encode who won the case (i.e., government or taxpayer) and in what court.

Dimensions encode various perspectives or argumentative approaches. For instance, in the home office deduction domain, the greater the percentage of time that is spent in business activities in the home office, the stronger the argument that one can make for establishing that the home office satisfies

the statutory predicate *principal place of business* and the statutory rule of Section 280(A)(c)(1). A dimension typically encodes a factor considered in past cases. Dimensions provide a way to assess the strength of a case from this perspective and to compare cases that share this analytic perspective, *ceteris paribus*. Thus, a case in which the taxpayer spends 70% of his time in the home office is stronger (from the point of view of the taxpayer) than one in which only 40% is spent there.

Dimensions are encoded in terms of factual predicates (e.g., *exists-other-office*, *relative-home-work-time-information*). The values of factual predicates are computed from the data stored in the cases. Prerequisite factual predicates are listed for each dimension. Dimensions also specify the procedures to make comparisons with respect to the dimension, and the so-called **focal** slots that encode the information central to the comparison (e.g., *home-office-hours-worked*, *other-office-hours-worked*). Dimensions use a repertoire of comparison methods (e.g., linear ordering, calculated ratio or percentage, set inclusion).

In HYPO-style systems, a case in the CKB is **relevant** to the current fact situation if it shares one or more applicable dimensions with it. A Case A is **more on point** than Case B if the set of dimensions Case A shares with the cfs includes those that Case B shares with the cfs. Note on-pointness is not a total order based simply on numbers of applicable dimensions. It is a partial order based on set inclusion of subsets of applicable dimensions. Cases which are maximal in the on-pointness ordering—that is, there is no case which is more on-point—are called **most on-point** cases. Most on-point cases that satisfy further requirements—in particular, that they were won by same side as the point-of-view specified for CABARET by the user—are called **best** cases. Best cases are the kind of cases that one seeks for case support when making precedent-based arguments.

The partial ordering of relevant cases according to the on-pointness measure can be represented in a lattice (also known as a Hasse diagram), called the **claim lattice**. Most on-point cases are found in the root or maximal nodes in this lattice. Best cases, if they exist, are a subset of most on-point cases. (Various definitions of best case have been used in the family of HYPO-style systems. With some definitions, there may not be any best cases.)

4. CABARET's Control Rules

CABARET uses various types of control rules to guide its problem-solving. (See [Rissland & Skalak, 1991] for more details.) Examples include:

1. If one mode of reasoning fails, then switch to the other.

2. Once a conclusion is reached, switch the form of reasoning to check if it holds in the other mode.

3. If all but one antecedent of a rule can be established, then use CBR to show the missed antecedent can be established using cases.

4. If all but one antecedent of a rule can be established, then use CBR to broaden the application of the rule with respect to the missing antecedent.

5. Use CBR on terms which are deliberately open-textured.

The third and fourth "near miss" heuristics are actually elements of a three-tiered theory of strategies and tactics for statutory interpretation [Skalak & Rissland, 1992].

At the top level are four **argument stances.** Argument stances take into consideration the point-of-view being taken by CABARET and the status of a statutory rule on the current fact situation. Each argument stance indexes four argument moves. **Argument moves** are indexed by the point-of-view and the status of the rule on cases in CABARET's CKB. Argument moves are carried out by fundamental **CBR primitives** like distinguishing and analogizing, which form the third level.

In essence, argument stances look to the cfs whereas argument moves look to the CKB. One chooses an argumentative stance by seeing where one stands on a rule in one's cfs. For instance, one might choose to **broaden** a rule that on its face does not hold in order to have it cover one's current facts. Or one may seek to **discredit** a rule that does seem to apply. If the rule points to the desired result, one can **confirm** its application.

	Rule conditions met in cfs	Rule conditions not met in cfs
Pro	Confirm the hit	Broaden the rule
Con	Discredit the rule	Confirm the miss

Argument Stances

How any one of these high level argument strategies are actually carried out depends on what type of cases are available in the CKB. one must

retrieve and reason with appropriately similar cases. The matrix of moves for broadening a rule is:

	Rule consequent established in retrieved case	Rule consequent not established in retrieved case
Case has desired outcome	Analogize cfs and the case	Analogize case outcome plus Distinguish rule consequent
Case does not have desired outcome	Distinguish case outcome plus Analogize rule consequent	Distinguish cfs from the case

Argument Moves for Broadening

5. Conclusions

In this paper, I have briefly reviewed the CABARET system built at the University of Massachusetts CBR Lab. CABARET is a hybrid CBR-RBR system that employs HYPO-style CBR and traditional RBR. It is controlled by an agenda-based controller that embodies a three-tiered theory of statutory interpretation involving argument stances, moves, and primitives.

CABARET addresses many of the well-known problems of statutory interpretation including reasoning with exceptions to rules and the open-textured nature of constituent terms. CABARET's approach is applicable to a wide variety of domains, including non-legal ones. It enables a system to evolve through experience with cases.

There are many opportunities for refining the CABARET approach. For instance, one could include a standard inductive learning component, such as C4.5. One could include more information about the highlevel purposes of the user, such as presentation of a balanced pro-con analysis or a one-sided advocacy brief. While some of these issues have been addressed in some of our other research (e.g., [Rissland et al., 1993]), there are many yet unexplored.

In addition, one could view CABARET as an information retrieval engine. This opens up many possibilities, for instance, using CABARET to drive a more traditional information retrieval (IR) tool. Using CBR to drive IR is a topic of on-going research in our CBR Lab. Another CBR project in a statutory domain is the BankXX Project [Rissland et al., 1993, 1994]. It explores the use of heuristic search to guide the perusal and harvesting of information in an area of personal bankruptcy law.

6. References

Ashley, K. D. (1990). *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. Cambridge, MA: M.I.T. Press.

Bareiss, E. R. (1989). *Exemplar-Based Knowledge Acquisition*. Boston, MA: Academic Press.

Berman, D. H. (1989). Cutting Legal Loops. *Second International Conference on Artificial Intelligence and Law (ICAIL-89)*. Vancouver, B.C. Association for Computing Machinery.

Branting, L. K. (1991). Building Explanations from Rules and Structured Cases. *International Journal of Man-Machine Studies*, 34(6), 797-837.

Davis, R., & Lenat, D. B. (1982). *Knowledge-Based Systems in Artificial Intelligence*. McGraw Hill. New York.

Fuller, L. L. (1958). Positivism and fidelity to law: a reply to professor Hart. *Harvard Law Review*. 71, 630-672.

Gardner, A. vdl. (1987). *An Artificial Intelligence Approach to Legal Reasoning*. M.I.T. Press, Cambridge.

Hart, H. L. A. (1958). *The Concept of Law*. Clarendon Press. Oxford, U.K.

Hart, H. L. A. (1961). Positivism and the separation of law and morals. *Harvard Law Review*. 71, 593-629.

Lakatos. I. (1976). *Proofs and Refutations*. Cambridge University Press. London, U.K.

Levi E. H. (1949). *An Introduction to Legal Reasoning*. University of Chicago Press. Chicago, IL.

Llewellyn, K. N. (1989). *The Case Law System in America*. University of Chicago Press. Chicago, Il.

McCarty, L. T., & Sridharan, N. S. (1981) The Representation of an Evolving System of Legal Concepts: II Prototypes and Deformations. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI-81)*, 246-253.

Morris, C. (1937). *How Lawyers Think*. Harvard University Press. Cambridge, MA.

Polya, G. (1957). *How To Solve It*. Second Edition. Princeton University Press, Princeton, N.J.

Rissland, E. L. (1978). Understanding Understanding Mathematics. *Cognitive Science*, 1(4): 361-383.

Rissland, E. L., Daniels, J. J., Rubinstein, Z. B. & Skalak, D. B. (1993). Case-Based Diagnostic Analysis in a Blackboard Architecture. *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 66-72. Washington, DC. AAAI Press/MIT Press.

Rissland, E. L. & Skalak, D. B. (1991). CABARET: Rule Interpretation in a Hybrid Architecture.

International Journal of Man-Machine Studies, 34(6): 839-887.

Rissland, E. L., Skalak, D. B. & Friedman, M. T. (1993). Case Retrieval through Multiple Indexing and Heuristic Search. *Proceedings of IJCAI-93*, Chambery, Savoie, France. International Joint Conferences on Artificial Intelligence.

Rissland, E. L., Skalak, D. B. & Friedman, M. T. (1994). BankXX: Supporting Legal Arguments Through Heuristic Retrieval." Computer Science Technical Report No. 94-76. Also submitted for publication.

Rosch, E. & Mervis, C. B. (1975). Family Resemblances: Studies in the Internal Structure of Categories. *Cognitive Psychology*, 7, 573-605.

Sergot, M. J., Sadri, F., Kowalski, R. A., Kriwaczek, F., Hammond, P. & Cort, H. T. (1986). The British Nationality Act as a Logic Program. *Communications of the ACM*, 29(5): 370-386.

Skalak, D. B. (1992). Representing Cases as Knowledge Sources that Apply Local Similarity Metrics. *The 14th Annual Conference of the Cognitive Science Society*, 325-330. Bloomington, Indiana. Lawrence Erlbaum.

Skalak, D. B. & Rissland, E. L. (1992). Arguments and Cases: An Inevitable Intertwining. *Artificial Intelligence and Law: An International Journal*, 1(1): 3-48.

Twining, W. & Miers, D. (1982). *How To Do Things With Rules*. Second Edition. London: Weidenfeld and Nicolson.

Walker, R. F., Oskamp, A., Schrickx, J. A., Van Opdorp, G. J., & van den Berg, P. H. (1991). PROLEXS: creating law and order in a heterogeneous domain. *International Journal of Man-Machine Studies*, 35(1): 35-69.

Waterman, D., & Peterson, M. (1981). *Models of Legal Decision Making*. Memo R-2717-ICJ, The Institute for Civil Justice of The Rand Corporation.