

**A Case-Based Approach To
Modeling Legal Expertise¹**

Kevin D. Ashley and Edwina L. Rissland

COINS Technical Report 88-17

**Department of Computer and Information Science
University of Massachusetts
Amherst, Massachusetts 01003**

¹To appear in IEEE Expert

A CASE-BASED APPROACH TO MODELING LEGAL EXPERTISE

Kevin D. Ashley and Edwina L. Rissland
Department of Computer and Information Science
University of Massachusetts
Amherst, Massachusetts 01003

Designers of expert systems are turning to case-based reasoning ("CBR") to capture expertise in domains where rules are ill-defined, incomplete or inconsistent. The law is an excellent domain to study case-based reasoning. Attorneys and judges reason analogically with precedent cases as an indispensable supplement to reasoning deductively with legal rules. The rules' predicates are simply not well-enough defined to allow the right decision to be inferred deductively. In fact, there seldom is just one "right answer" to a legal question. Instead, legal experts make competing arguments pitting conflicting interpretations of cases and facts against each other.

HYPO is a computer program that performs case-based reasoning in the legal domain. It is designed to help attorneys analyze and make arguments about a new fact situation in terms of the most relevant precedent cases. To perform this task, indexing and retrieval of relevant cases are not enough. HYPO needs to make factual comparisons of the cases relative to the problem situation and to determine the legal significance of the comparisons in terms of arguments about the problem situation.

In this paper, we demonstrate techniques employed in HYPO to compare cases and choose the best cases for evaluating and making arguments about a new fact situation. In particular, we demonstrate how HYPO critically compares a problem situation (called the current fact situation or "cfs") to the most relevantly similar precedent cases (called "most-on-point cases" or "mopc's") in order to outline an argument how to decide the current fact situation based on its significant similarities to and differences from the most-on-point cases.

HYPO's main tool for this task is the mechanism of claim-lattices. A claim-lattice projects the Case Knowledge Base onto the problem situation to create a neighborhood of cases surrounding the problem situation in which the above comparisons become explicit. We present a detailed example of a claim-lattice actually generated by HYPO to analyze

a real legal case.

1. Neighborhoods of Cases

HYPO has a Case Knowledge Base ("CKB") of leading cases in trade secrets law and an indexing scheme based on *dimensions* for retrieving cases that share important features with a problem situation as input. Beyond indexing and retrieval of relevant cases, however, it is also necessary to organize the cases in terms of how relevant they are for making or responding to arguments about the problem situation. An attorney cannot adequately evaluate the problem situation without drawing certain symbolic connections among the relevant cases in terms of their comparative:

1. **Factual similarities** and differences relative to the current fact situation – Which cases are more on point (i.e., share more features with the problem situation) than other cases and which cases are less on point than a given case?
2. **Outcomes** – Of the cases that are most on point to the problem situation, how do their outcomes compare? If comparably on point cases have conflicting outcomes, what are their differences relative to the problem situation?
3. **Uses in a legal argument** about the current fact situation – Which cases make stronger legal points for a party on a claim; which cases are troublesome to an argument; which cases distinguish other cases; which take a point to extremes?
4. **Potential relevance** to the current fact situation – Which cases are nearly on point or most-on-point but for some missing facts that might jog the attorney's mind to look for and discover new facts and arguments about his current fact situation.
5. **Significance** to other parts of the legal argument – What are the cases and connections when the current fact situation is viewed from the slant of a different kind of legal claim?
6. **Possible variations** – What new cases and connections come into view when the facts of the current fact situation are changed hypothetically? How do the connections change when features are added, subtracted, exaggerated, combined with those of neighboring cases?

In effect, an attorney conceptually organizes the relevant cases into a neighborhood of cases, more or less on point to the problem situation, some with desirable outcomes, some not, some especially useful for responding to others, some only potentially so. Within the context of the neighborhood of relevant cases, small changes in the facts of the problem situation can push it closer to one precedent and farther from others – with significant results in the ways the cases can be cited and responded to in arguments.

HYPO uses *claim-lattices* to represent the conceptual neighborhood of cases. Claim-lattices allow HYPO to exploit the connections among relevant cases that arise from comparing them relative to the current fact situation. A claim-lattice projects the Case Knowledge Base onto the current fact situation to create the neighborhood of cases that surround the problem situation. The neighborhood of cases is the Case Knowledge Base as seen *modulo* the current fact situation. Claim-lattices: (1) organize and abstract all of the cases from the CKB that are relevant for analyzing the current fact situation, (2) place them in a graph according to their comparative similarities and usefulness as precedents for arguing how to decide the current fact situation, (3) focus on troublesome contrary cases, and (4) suggest fruitful combinations of facts for new hypotheticals.

2. Knowledge Representation in HYPO

HYPO uses two kinds of domain knowledge to construct claim-lattices: (1) the **Case Knowledge Base (CKB)** containing actual legal cases and (2) the library of dimensions. HYPO's current CKB contains 30 or so cases on trade secrets misappropriation and a few related areas, like basic contract law. For a complete description of the HYPO program, see [Ashley, 1988].¹

Each legal case in the CKB corresponds to real legal dispute, tried by a court, whose decision as reported in a published opinion is represented in a case representation language consisting of hierarchical clusters of frames (implemented as flavors) which describe the main components of the case including: plaintiff (π), defendant (δ), legal claim, prevailing party (plaintiff or defendant), holding and facts. Some of the factual features are in turn

¹Other researchers like Gardner, Schlobohm, and Waterman, place less or no emphasis on representing and indexing cases and hypotheticals but instead employ generalized fact patterns [Gardner, 1984], if-then rules [Schlobohm and Waterman, 1987; Waterman and Peterson, 1980] or formal logic [Sergot *et al.*, 1986]. McCarty's early work did emphasize adversarial reasoning including the use of hypothetical reasoning [McCarty and Sridharan, 1981; McCarty and Sridharan, 1982]. More recently, he has focussed on a deontic logic for representing permissions and obligations [McCarty, 1985].

expanded and represented as frames (e.g., plaintiff, products, employees, disclosure events, secret information and agreements). See [Ashley, 1988; Rissland *et al.*, 1984] for examples. A legal claim is a recognized kind of complaint for which the courts will grant relief (e.g., breach of contract, negligence, trade secrets misappropriation, copyright infringement). The holding is the decision of the court as to the legal effect on each claim of the facts of the case, either in favor of the plaintiff or defendant.

Current fact situations are represented in the same case representation language as the cases in the CKB. From this basic level of representation of a fact situation, HYPO computes whether certain higher level descriptors, called **factual predicates**, are satisfied. **Factual predicates** state whether or not a particular legal fact is true (e.g., there-exist-disclosees, employee-has-switched-employers) in a fact situation. Factual predicates form a language used to encode the second source of legal knowledge in HYPO, the dimensions.

Dimensions capture the legal relevance of a cluster of facts to the merits of a claim. For a particular kind of case, dimensions generalize collections of facts that constitute strengths and weaknesses in a party's position. Each of the generalizations can be backed up by one or more cases where a court held in favor of a party, in part because of the cluster of facts associated with the dimension. Dimensions allow HYPO to view the cases from various perspectives. They can be thought of as "cross sections" of the facts of a case emphasizing the features that are important from the viewpoint of a particular legal claim.

At this point, thirteen dimensions are implemented, although we know about thirty dimensions in all for the trade secrets and related contracts domains. We do not compile these ourselves but rather take them from scholarly analyses and treatises like [Gilburne and Johnston, 1982; Milgrim, 1985]. Figure 1 shows examples of some potential strengths and weaknesses in a trade secrets situation and the dimensions that capture them.

A dimension is also a frame-like knowledge source. It has several facets that enable HYPO to perform various tasks: *First*, the program can test if a dimension applies to a case or is a near miss using the dimension's prerequisites which are stated in terms of factual predicates. For instance, the prerequisites of the *Bribe-Employee* dimension are that two corporations, plaintiff and defendant, compete with respect to a product, plaintiff has confidential product information to which defendant has gained access by luring plaintiff's former employees to work for defendant and to disclose the information. *Second*, HYPO compares cases along a dimension using its focal slots. The focal slot of *Bribe-Employee* is what defendant offered the employees to lure them to switch employment; its range is a set of possible enticements including salary increases, stock options, bonuses, promotions,

or no enticement. To strengthen the plaintiff's position in a fact situation to which this dimension applies, add more enticements for inducing the employees to defect. *Third*, HYPO finds similar cases by retrieving cases that the dimension indexes. *Bribe-Employee* indexes at least two cases in the CKB: (1) *Telex v. IBM* in which the court held for the plaintiff's, IBM's trade secrets claim where Telex had offered stock options, higher salaries and bonuses (one for \$500,000) to IBM's employees and (2) the *Midland Ross* case where defendant won eventhough the former employee gained a modest salary increase in switching employers. Other examples of dimensions can be found in [Ashley, 1988; Ashley and Rissland, 1987b; Rissland and Ashley, 1987; Ashley and Rissland, 1987a].

3. Representing Neighborhoods of Cases with Claim-lattices

A current fact situation is presented to HYPO in the case representation language. In analyzing a new current fact situation, HYPO runs through the library of dimensions and produces a **case-analysis-record** that contains: (1) applicable factual predicates; (2) applicable dimensions; (3) near-miss dimensions; (4) potential claims and (5) relevant cases from the CKB. **Near-miss** dimensions are those for which some, but not all, of the prerequisites are satisfied. The combined list of applicable and near-miss dimensions is called the **D-list**. Figure 2 describes a current fact situation (cfs) based, for purposes of illustration, on *Telex v. IBM*, a real case in the CKB. Figure 3 shows the case-analysis-record for the cfs.

HYPO uses the case-analysis-record to construct the **claim-lattice**, which is a lattice such that: (1) the root is the current fact situation together with its D-list; and (2) successor nodes contain pointers to cases that share a subset, usually proper, of the dimensions in the cfs's D-list. Figure 4 shows the claim-lattice actually generated by the HYPO program for analyzing the current fact situation of Figure 2 from the viewpoint of a trade secrets misappropriation claim. (There is a separate claim-lattice for each possible claim.)

The main idea about claim-lattices is that they order relevant cases from the CKB in terms of how **on point** they are to the current fact situation, where this is measured by the degree of overlap between the set of dimensions of the cfs and that of retrieved cases. In other words, the claim-lattice captures the extent that cases share the strengths and weaknesses of the current fact situation. More specifically, of all the dimensions in the current fact situation's D-list, all and only those in a particular successor node's D-list (which is subset of the cfs's D-list) apply to each of the cases associated with that node. Each successor node is the ancestor of all nodes whose subsets of dimensions are proper

subsets of *its* subset. The descendants are ordered in terms of the set inclusiveness of their subsets of the current fact situation's D-list. Each successor node may contain pointers to more than one case and comprise cases won by plaintiffs (π) and defendants (δ).

4. Interpreting the Claim-lattice

The ordering scheme enables claim-lattices to capture a sense of closeness to the current fact situation of cases in the Case Knowledge Base. It views the relevant cases in terms of their relevance – or on-pointness – to the current fact situation. Computationally, this means that the nodes closest to the root can be considered most-on-point cases (mopc's) to the current fact situation (as long as their subsets of the cfs's D-list do not contain near-miss dimensions); leaf nodes are the least on point. All of the cases displayed are relevant to the current fact situation because they all share some legally important strengths or weaknesses with the fact situation as represented by the dimensions shared with the cfs.

HYPO makes a dynamic judgment of which case appears to be most-on-point. It is possible for a well-known landmark case not to appear since it is not relevant to the case at hand because, for instance, it does not share enough overlap of dimensions (and therefore facts) and thus the basis of its decision rests on factors not present in the current case. Likewise, a seemingly unimportant case might be very relevant modulo the current fact situation because not only does it share the same base of facts (and therefore dimensions) but also it might share more than any other case. This is shown in the claim-lattice by positioning those cases sharing more dimensions nearer to the current fact situation.

Different major branches of the lattice indicate different ways to argue the case, effectively one way for each group of most-on-point cases. HYPO can argue the case for *Side 1*, let us say the defendant in the current fact situation of Figure 2, by *citing* a pro-defendant mopc. In the claim-lattice for the cfs, Figure 4, the most-on-point cases are *Telex v. IBM*, *Raycorp v. Tronic* and *Modern Controls*. Of course, *Telex v. IBM* is also the basis of the current fact situation in Figure 2. A case *should* be most on point to itself! *Midland Ross*, for example, is not a most-on-point case because, although it is very close to the root, the *Disclose-Secrets* dimension which applies to *Midland Ross*, and which would help δ if it applied to the current fact situation, is only a near-miss for the cfs. (Note the presence of an asterisk (*) next to *Disclose-Secrets* in Figure 4.) Of the three most-on-point cases, only *Raycorp v. Tronic* held for the defendant (δ), so that is the case HYPO cites for *Side 1*'s argument.

HYPO produces the following argument on behalf of the defendant as *Side 1* that it

should win on a claim for Trade Secrets Misappropriation in the current fact situation:

[A] ↔ Side 1's Point: (δ)

Cite: *Raycorp v. Tronic* (δ should win even though employee who disclosed information signed nondisclosure agreement with π .)

[B] ↔ Side 2's Response: (π)

Distinguish *Raycorp v. Tronic*: (In the current fact situation, defendant's access to plaintiff's product information saved it more time or expense than in *Raycorp v. Tronic* and defendant paid plaintiff's former employee to switch employment.)

Cite as Counter-example: *Modern Controls* (This case is just as on point and held for π where the employee who disclosed info to δ signed a nondisclosure agreement with π .)

In support of its point, HYPO draws the analogy between the most-on-point case and the current fact situation. Since most-on-point cases share the most legally important strengths and weaknesses with the current fact situation (i.e. mopc's are the closest analogies to the cfs), *Raycorp v. Tronic* is the most persuasive case HYPO could cite for the defendant as *Side 1*. The relevantly similar facts are just those summarized by the dimension[s] that the mopc shares with the cfs. In this case there is but one shared dimension *Agreed-Not-To-Disclose*. The argument point in [A] is not very strong but it is well formed from a legal viewpoint.

HYPO also uses the claim-lattice to figure out how to respond to an argument like *Side 1's* by citing counter-examples to the cited case. In a legal argument, an attorney can discount the persuasive effect of a cited case by citing another case that is just as or more on point but held for the opponent. Finding such a case is a simple matter in the claim-lattice; if it exists it must be a most-on-point case residing in one of the nodes between the root and the node containing the cited case. (Mopc's on other major branches of the lattice won't do – that would be like comparing apples and oranges.) In Figure 4 mopc's that occupy positions as counter-examples are *Telex v. IBM* and *Modern Controls*, both of which held for plaintiff (π) and are as or more on point than *Raycorp*. In other words, HYPO makes *Side 2's* (i.e., the plaintiff's) response to *Side 1's* argument by citing the

Modern Controls case as a counter-example. (See [B] above.) (It would be silly for HYPO to cite the *Telex* case in response, but that is just because, for illustration, we have chosen *Telex* as the current fact situation.) In supporting the response, HYPO underscores the additional facts, if any, that make the counter-example as or more on point than the case cited for *Side 1*.

The claim-lattice supports other ways of responding to a point by distinguishing the cited case. Suppose HYPO had cited the *Midland Ross* case on behalf of *Side 1* as in the following exchange:

[C] \hookrightarrow Side 1's Point: (δ)

Cite: *Midland Ross* (δ should win eventhough defendant paid plaintiff's former employee to switch employment.)

[D] \leftarrow Side 2's Response: (π)

Distinguish *Midland Ross*: (In the current fact situation, defendant's access to plaintiff's product information saved it more time or expense than in *Midland Ross* and plaintiff's former employee entered into a nondisclosure agreement with plaintiff. In *Midland Ross* plaintiff disclosed its product information to more outsiders than in the current fact situation.)

In distinguishing, HYPO points out the "dis-analogy" between the current fact situation and *Midland Ross*, that the facts associated with the *Disclose-Secrets* dimension, namely that the plaintiff disclosed its secrets to outsiders, obtained only in *Midland Ross*, *not* in the current fact situation. Also, in the cfs, defendant gained a competitive advantage and plaintiff's employee had entered into nondisclosure agreements. In other words, using the claim-lattice and reciting *unshared* dimensions that help π in the current fact situation or hurt π in *Midland Ross*, HYPO knows that and why the current fact situation presents a much stronger case for plaintiff than *Midland Ross*.

As the above examples illustrate, the claim-lattice embodies HYPO's knowledge of how to compare cases relative to the current fact situation and what the legal significance of those comparisons are in terms of arguments about the current fact situation. See [Ashley and Rissland, 1987b].

HYPO uses that knowledge to help an attorney build an argument, for example, by finding cases that potentially are most-on-point. A *potential mopc* is very similar to the fact situation, except that some dimensions (i.e. a strength or weakness) that apply to it are near-misses with respect to the cfs; they are located in the nodes closest to the root whose subsets of dimensions contain near-misses. Potential most-on-point cases are useful: (1) in fact finding about the current fact situation, (2) when looking for cases to cite as counter-examples to a particular most-on-point case and (3) in planning for the contingency that an opponent might be able to prove that the missing facts are true. In Figure 4, the *Midland Ross*, *Structural Dynamics* and *Automated Systems* cases are all potential mopc's. If it were true that the plaintiff in the current fact situation had made disclosures of his secrets to, let's say two hundred outsiders, *Midland Ross* would become a very important case to *Side 1*, allowing defendant's point in [C] to be recast as follows:

[C'] \leftrightarrow Side 1's Point: (δ)

Cite: *Midland Ross* (δ should win where plaintiff disclosed secrets to outsiders even though defendant paid plaintiff's former employee to switch employment.)

[D'] \leftrightarrow Side 2's Response: (π)

Distinguish *Midland Ross*: (In the current fact situation, defendant's access to plaintiff's product information saved it more time or expense than in *Midland Ross* and plaintiff's former employee entered into a nondisclosure agreement with plaintiff.)

The point in [C'] is stronger than that of [C] because the analogy between the cited case and the current fact situation is stronger. Correspondingly, the distinction in response [D'] is weaker than that in [D].

Alternatively, if the current fact situation were more like the potential most-on-point case *Structural Dynamics*, *Side 2* would have an even stronger response to the point in [A] in which it can cite a more-on-point counter-example:

[B'] ↔ Side 2's Response: (π)

Distinguish *Raycorp v. Tronic*: (In the current fact situation, defendant's access to plaintiff's product information saved it more time or expense than in *Raycorp. v. Tronic* and defendant paid plaintiff's former employee to switch employment.)

Cite as Counter-example: *Structural Dynamics* (This case is *more on point* and held for π where the employee who disclosed info to δ signed a nondisclosure agreement with π and brought π 's product development tools to δ .)

The claim-lattice also focuses on another kind of extreme case: *boundary cases*, the cases in which the plaintiff still won [or lost] despite the fact that of all the cases in the CKB, they presented the weakest [or strongest] fact situations for the plaintiff along some dimension. A boundary case may not be as on point as a mopc, but it is still useful to HYPO as a counter-example. For instance, suppose in the current fact situation that IBM had made disclosures to 200 outsiders and that *Side 1* had cited what would then be a most-on-point case for the defendant: the *Midland Ross* case. In responding for *Side 2*, HYPO could also cite the *Data General* case as a counter-example (see Figure 4):

[D'] (cont.) ↔ Side 2's Response: (π)

Cite as Counter-example: *Data General* (This case held for π even though π disclosed to 6000 outsiders, more than in cfs or *Midland-Ross*.)

The new response implies that π 's behavior could have been much worse and still it should win. Here, *Data General* is used as an example of a much weaker case for the plaintiff along the *Disclose-Secrets* dimension (with disclosures to 6000 outsiders) whose outcome still favored plaintiff.

In the previous example we posed a *hypothetical* variation of the current fact situation (that IBM had made disclosures to 200 outsiders) to make a legal point about the connections between the *Data General* and *Midland Ross* cases. That is a very natural thing for attorneys to do and points out one of the most important uses of claim-lattices: they suggest interesting hypothetical cases.

The structure of the claim-lattice provides clues about how to “flesh out” sparse areas of the Case Knowledge Base. It suggests how to modify hypothetically either actual cases or the current fact situation to construct hypothetical cases and new connections between the cfs and its neighbors. The idea is, having located the current fact situation in its neighborhood of cases, to use hypothetical modifications to explore the neighborhood.

Interesting hypotheticals reside in two locales of the claim-lattice:

First, as the above hypothetical variant of the current fact situation suggests, in the nodes containing potentially most-on-point cases. Using the potential mopc’s as targets, the current fact situation can be modified slightly to incorporate the “missing” facts associated with the near-miss dimensions, for example, in Figure 4: (1) by adding disclosures as in *Midland Ross*, (2) to favor the plaintiff, by having former employees bring plaintiff’s product-related notes and copies of code as in the *Structural Dynamics* case, or (3) by making the secret information be vertical knowledge about customer’s business methods as in *Automated Systems*. For a more complete account of posing hypotheticals in HYPO see [Ashley, 1988; Rissland and Ashley, 1986].

HYPO uses the hypothetical variants of the current fact situation to explain the significance of the potential mopc’s and to illustrate the effect on the argument of the hypothetical facts’ becoming true. For example, if IBM did make such disclosures, not only would *Midland Ross* become a most-on-point case, but *Data General* would also become potentially an even more on point case that could be used to respond to an argument citing the *Midland Ross* case. HYPO makes these *changes* in possible arguments explicit by reconstructing the claim-lattice around the modified current fact situation. Using the claim-lattice to plan for contingencies in an argument is very practical. In the real *Telex Corp. v. IBM Corp.*, 367 F. Supp. 258, 358 (N.D. Okla. 1973), defendant Telex actually attempted to raise the issue of disclosures by IBM. In its opinion, the court cited another case involving disclosures by *Midland Ross* but disposed of the defense on other grounds.

Second, other interesting hypos reside between nodes containing most-on-point cases; they combine the features of π ’s mopc’s with those of δ ’s. In pitting pro plaintiff and pro defendant features against one another, these *hybrid mopc’s* implicitly raise the issue of which features are more important (i.e., how would such a case be decided.) In Figure 4, an interesting hypothetical hybrid most-on-point case combines the pro-plaintiff features of the *Telex* case with the pro-defendant disclosures of secrets as in the *Midland Ross* case. It happens to be the same hypothetical as above, but now we are changing the *Telex* case rather than the current fact situation (which, for purposes of illustration is based on *Telex*.) Judges and attorneys actually use hypotheticals like this one. In his opinion in

Executive Development Center of Boston v. Hirsch, Civ. Action 50441 (Suffolk Superior Ct. Mass. 1985), p.28, Judge Young makes a point by constructing a similar hypothetical that pits plaintiff's disclosures against defendant's improper means of access. The hybrid hypo can be made even more interesting by pushing the number of disclosees to extremes - one disclosure or one million. See [Rissland and Ashley, 1986].

Another use of the claim-lattice is to view a fact situation from the perspective of other kinds of claims. The lattice in Figure 4 depicts the current fact situation as a trade secrets misappropriation case. But by virtue of the *Agreed-Not-To-Disclose* dimension, the current fact situation could also be thought of as involving a claim for breach of the nondisclosure agreements entered into by the employees. The claim-lattice representing that claim is yet another projection of the CKB onto the current fact situation, this time creating a neighborhood of contracts cases organized along different dimensions like *agreement-supported-by-consideration* but subject to the same interpretive methods as described above.

5. Conclusion

In this paper, we have described a technique for dynamically viewing or reorganizing a case knowledge base around a particular fact situation to provide a sophisticated analysis of it. Claim lattices project the fact situation through the Case Knowledge Base to create a neighborhood of cases surrounding the situation in which factual and interpretive comparisons of the cases relative to the situation become both simple and explicit. We have described how the HYPO program uses claim lattices to create a skeletal legal argument about a fact situation, spot troublesome contrary cases and suggest fruitful combinations of facts for new hypotheticals.

Acknowledgements

This work was supported in part by: the Advanced Research Projects Agency of the Department of Defense, monitored by the Office of Naval Research under contract no. N00014-84-K-0017; the University Research Initiative, award no. N00014-86-K-0764; and an IBM Graduate Student Fellowship.

6. Figures

Plaintiff's (π 's) position is strengthened to extent:

Brought-Tools: π 's former employees brought π 's notes, diagrams, tools to defendant (δ).

Competitive-Advantage: δ 's access to π 's secret information gave δ a competitive advantage.

Disclose-Secrets: π did *not* voluntarily disclose his secrets to outsiders.

Noncompete-Agreement: π 's employees had entered into nondisclosure agreements.

Bribe-Employee: δ bribed π 's employees to switch employ.

Vertical-Knowledge: π 's secrets were not simply about customer's business methods.

Figure 1: Sample Dimensions and Related Factual Strengths

An attorney's corporate client, IBM, complains that Telex has been misappropriating its trade secrets in its "Merlin" disk drive system. Specifically, IBM complains:

- Telex offered IBM's Merlin project engineers extremely large salaries, bonuses (e.g., one for \$500,000) and stock options to induce them to join Telex. *Bribe-Employee*
- All of the former IBM employees had entered into nondisclosure agreements with IBM to keep confidential IBM's trade secret information. *Noncompete-Agreement*
- As a result of its access to IBM's trade secrets, Telex was able to develop its competing products in substantially less time and at lower expense. *Competitive-Advantage*

Figure 2: The Current Fact Situation ("CFS") based on *Telex v. IBM*. (Names of dimensions for which recited facts are focal are in *italics*.)

Applicable Factual Predicates: exists-corporate-claimant, exists-confidential-info, employee-switched-employers ...

Applicable Dimensions:

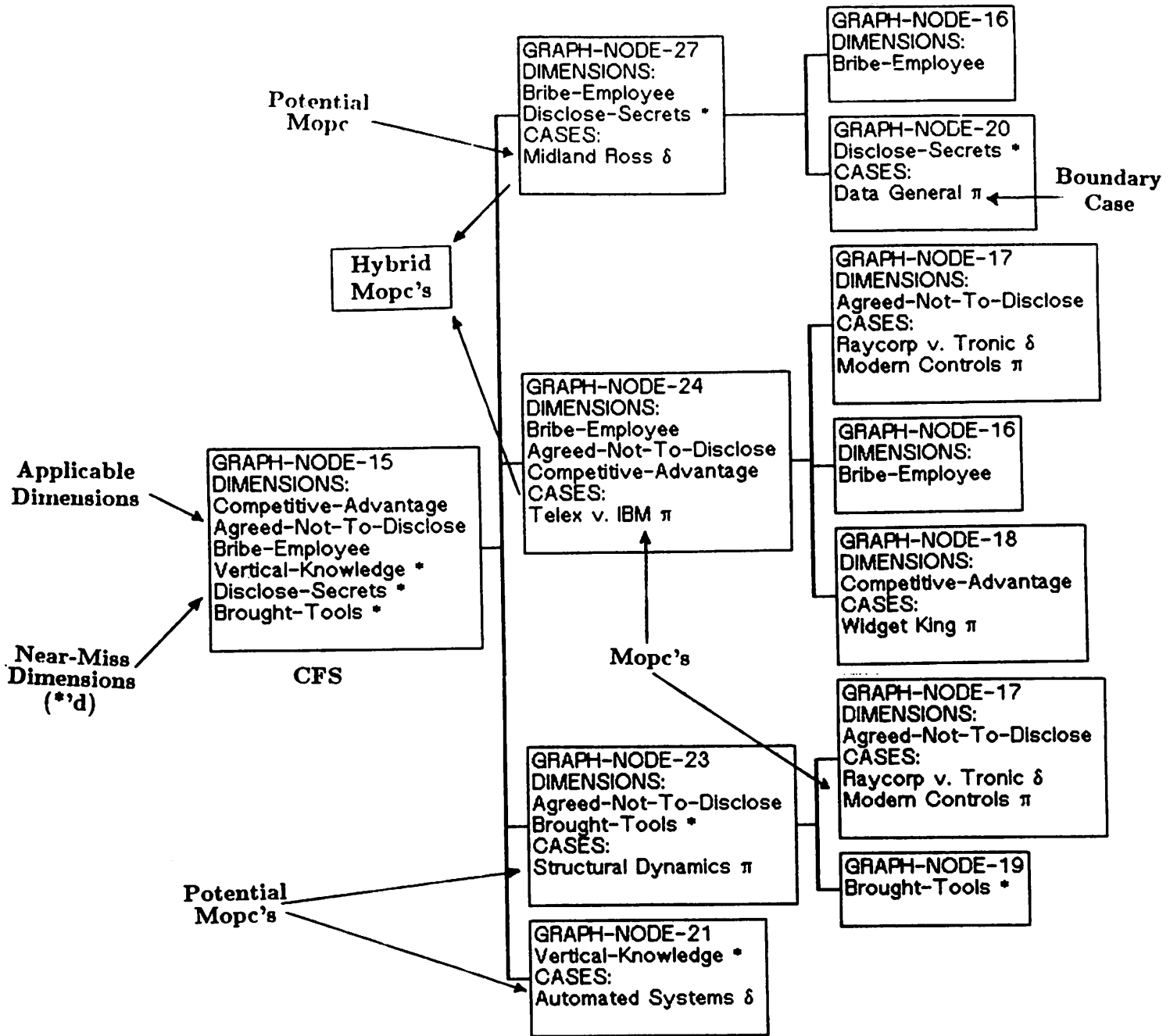
Agreed-Not-To-Disclose, Bribe-Employee, Competitive-Advantage

Near-Miss Dimensions: *Brought-Tools, Disclose-Secrets, Vertical-Knowledge*

Potential Claims: Trade Secrets Misappropriation, Breach of Nondisclosure Agreement

Relevant CKB cites: *Midland Ross, Data General, Structural Dynamics, Raycorp v. Tronic, Modern Controls ...*

Figure 3: Case-Analysis-Record for CFS



The root node represents the current fact situation and its D-list. (Dimensions that are near-misses as to the current fact situation have *'s.) Successor nodes contain pro-plaintiff (π) or pro-defendant (δ) cases, involving trade secrets misappropriation claims, that are on point to cfs. Nodes closest to root that do not have near-miss dimensions contain most-on-point cases (i.e., mopc's); otherwise they may contain potential mopc's. Leaf nodes are least-on-point. Each major branch of lattice that contains mopc's represents one way of arguing about the current fact situation. Mopc's may be counter-examples to cases with opposite outcomes in successor nodes. Boundary cases are examples of extremes along particular dimensions. Hypothetical hybrid mopc's combine features of different mopc's that hold for π and δ . Potential mopc's suggest fruitful hypothetical variants of current fact situation.

Figure 4: A Claim-lattice.

REFERENCES

- [Ashley, 1988] Kevin D. Ashley. *Modelling Legal Argument: Reasoning with Cases and Hypotheticals*. PhD thesis, Department of Computer and Information Science, University of Massachusetts, 1988.
- [Ashley and Rissland, 1987a] Kevin D. Ashley and Edwina L. Rissland. But, See, Accord: Generating *Blue Book* Citations in HYPO. In *First International Conference on Artificial Intelligence and Law*, Northeastern University, Boston MA, 1987.
- [Ashley and Rissland, 1987b] Kevin D. Ashley and Edwina L. Rissland. Compare and Contrast, A Test of Expertise. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, August 1987. Seattle.
- [Gardner, 1984] A. vdL. Gardner. *An Artificial Intelligence Approach to Legal Reasoning*. PhD thesis, Department of Computer Science, Stanford University, 1984.
- [Gilburne and Johnston, 1982] M. R. Gilburne and R. L. Johnston. Trade Secret Protection for Software Generally and in the Mass Market. *Computer/Law Journal*, III(3), 1982.
- [McCarty, 1985] L. Thorne McCarty. Permissions and Obligations. In Charles Walter, editor, *Computing Power and Legal Reasoning*, West Publishing Co., St. Paul, MN, 1985.
- [McCarty and Sridharan, 1981] L. Thorne McCarty and N.S. Sridharan. The Representation of an Evolving System of Legal Concepts: II. Prototypes and Deformations. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence, Inc., Vancouver, B.C., August 1981.
- [McCarty and Sridharan, 1982] L. Thorne McCarty and N.S. Sridharan. *A Computational Theory of Legal Argument*. Technical Report LRP-TR-13, Laboratory for Computer Science Research, Rutgers University, 1982.
- [Milgrim, 1985] Roger M. Milgrim. *Business Organizations, Milgrim on Trade Secrets*. Volume 12, Matthew Bender, New York, 1985.

- [Rissland, 1985] Edwina L. Rissland. AI and Legal Reasoning, Panel Report. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence, Inc., Los Angeles, CA, August 1985.
- [Rissland and Ashley, 1986] Edwina L. Rissland and Kevin D. Ashley. Hypotheticals as Heuristic Device. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, August 1986. Philadelphia, PA.
- [Rissland and Ashley, 1987] Edwina L. Rissland and Kevin D. Ashley. A Case-Based System for Trade Secrets Law. In *First International Conference on Artificial Intelligence and Law*, Northeastern University, Boston MA, 1987.
- [Rissland *et al.*, 1984] Edwina L. Rissland, E. M. Valcarce, and Kevin D. Ashley. Explaining and Arguing with Examples. In *Proceedings of the Fourth National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, Austin, TX, August 1984.
- [Schlobohm and Waterman, 1987] Dean A. Schlobohm and Donald A. Waterman. Explanation for an Expert System that Performs Estate Planning. In *First International Conference on Artificial Intelligence and Law*, Northeastern University, Boston MA, 1987.
- [Sergot *et al.*, 1986] Sergot, Sadri, Kowalski, Kriwaczek, Hammond, and Cory. The British Nationality Act as a Logic Program. *Communications of the ACM*, 29(5):370-386, May 1986.
- [Waterman and Peterson, 1980] D. A. Waterman and M. Peterson. Rule-Based Models of Legal Expertise. In *Proceedings of the First National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, Stanford, CA, August 1980.