An example of Integrating Legal Case Based Reasoning with Object-Oriented Rule-Based Systems: IKBALS II

George Vossos John Zeleznikow Tharam Dillon Vivian Vossos

{vossos,johnz,tharam}@latcs1.oz.au

Database Research Laboratory
Applied Computing Research Institute
La Trobe University
Bundoora Victoria Australia, 3083
Phone: (61) 3 479 2598
Fax: (61) 3 470 4915

Abstract:

Although case based reasoning (CBR) is still a research paradigm, it has been shown to offer significant advantages over conventional rule-based expert systems when attempting to reason through experience. This paper discusses extending an existing legal object-oriented/rule-based system to successfully handle CBR. In particular, we discuss the architecture of IKBALS II, a legal knowledge based system that performs statutory interpretation in the area of accident compensation. IKBALS II combines reasoning with rules and reasoning with previously tried precedent cases in order to determine if an injured employee is entitled to compensation under the Accident Compensation Act, 1985, Victoria. A worker is entitled to compensation only if the Accident Compensation Commission is satisfied that the worker falls within certain statutory definitions.

Keywords:

Legal knowledge based systems, combining rule-based and case-based reasoning, legal analogical reasoning, case based reasoning, integrating case based reasoning into object oriented/ rule based systems.

1 Introduction

1.1 Automating Legal Reasoning

Legal reasoning can be viewed as an intellectual process by which lawyers and judges use rules (statutes or regulations) and previously tried cases (precedents) to solve legal problems. Legal practitioners primarily combine two forms of reasoning when dealing with litigation: reasoning by deduction and reasoning by analogy [Levi, 1948], [Burton, 1985]. Legal reasoning is more than 'deduction', whereby lawyers rely on annotations and explanatory material (reports, practice guides, precedent cases, opinions of academic and distinguished lawyers) to help add some contextual information to legal rules. Hence, legal reasoning can be viewed as an attempt to interpret statutes initially through the use of the rules, referencing precedent cases only when the rules run out, or when the use of rules prove insufficient in elicitating concepts.

This paper describes a prototype legal knowledge based system (LKBS) code named IKBALS II which attempts to aid in the process of statutory interpretation by supplementing deductive reasoning with a form of analogical reasoning referred to as precedent-based or interpretive case based reasoning (CBR) in the area of accident compensation (WorkCare). The system allows lawyers to intelligently retrieve and analyze the sources of the law, namely The Accident Compensation (General Amendment) Act, 1989 and the associated case law, including

commentaries and reports, so as to successfully build a favourable argument for their client in court.

The problems with viewing legal reasoning as a 'system of rules' has been well documented [Hafner & Berman, 1990], [Tyree et al, 1987], [Gardner, 1987]. Besides the 'syntactic difficulties' inherent in normalizing statutes in the form of rules, there are 'semantic' difficulties such as the presence of conflicting rules, imprecise terms and incompleteness. The goal of this project is to design a computer model which will be capable of 'lending a hand' to a rule-based reasoner when it encounters open-textured legal predicates which it cannot resolve given the rule set(s) available and the facts of the current case. A precedent CBR module will then attempt to analogise the context of the current case to other cases in the case knowledge base (CKB) in order to first of all shed some light on the open-textured concept causing concern and secondly, to help interpret the legal category that the case belongs to.

1.2 Current Legal Knowledge Based Systems (LKBS)

Until recently, most legal knowledge based systems have generally tried to model the problem of statutory interpretation by simply normalizing sections of the legislation in the form of 'if_then_' rules, and then applying control and heuristic information in order to efficiently guide the logic of the system; (programs=logic + control [Kowalski,1979]). In consequence, the knowledge in these early legal knowledge based systems amounted to nothing more than collection of rules reliant on static necessary and sufficient conditions and meta-rules (heuristic information used to control the system) chained together in order to determine if a current case could be classified as belonging to a particular legal category [Skalak, 1989]. Once again, the bulk of the knowledge base was composed of 'control' or meta-rules rather than domain-specific rules. Such deductive reasoning allowed rules to be either 'forward chained': considered analogous to firing sufficient conditions, and/or 'backward chained': analogous to firing necessary conditions.

Although such rule-based models are both theoretically and practically very powerful, their suitability in reasoning with open textured domains such as law, however, must be doubted especially considering their inefficiency in adequately capturing contextual information contained in previously tried cases which must be considered when reasoning about the merits of a current case. Descriptions of recent legal expert systems using the rule-based approach can be found in [Waterman & Peterson, 1986], [Scholobohm & Waterman, 1987], [Sergot et al., 1986] and [Susskind, 1987].

From a technical perspective, representing legal rules in a strictly rule-based system is prone to the following limitations if sophisticated control techniques are not incorporated:

1. Inefficiency - The inference engine may have to search through a very large number of rules at each stage of execution. This results in a high problem solving overhead;

2. Maintainability - A disadvantage associated with rule based systems is that it is hard to follow their flow of control;

3. They may not properly reflect the reasoning structure used by the expert;

4. The open textured nature of legal knowledge and reasoning.

The last two factors often result in the knowledge mis-match problem which leads to cumbersome representations and laborious reasoning to reach decisions, which may not be acceptable. In the following sections, we will explain the reasoning processes employed by lawyers when reasoning about a legal problem in the WorkCare domain and then proceed to illustrate the underlying architecture used to support these processes in the IKBALS II prototype.

1.3 Forms of Legal Reasoning

In Australia Parliament drafts the laws and the courts interpret the laws with the assistance of the common law when appropriate. The courts also interpret the law where parliament has left a gap, e.g., legislation re murder under the Crimes Act is relatively short, but the precedent base and

literature about it is voluminous. Hence lawyers when considering a particular dispute normally try to find all the relevant cases to that particular dispute. They extract the applicable principles or interpretations from these previous precedent cases in order to form a judgement as to whether to proceed with the dispute. If so, they develop an argument in support of their position in the dispute, citing appropriate sections of the legislation and appropriate cases. They search for arguments which indicate that the precedents supporting their case are relevant to the particular case, whilst arguments supporting the opposition case are not.

Some means of organizing the overwhelming mass of legal material a lawyer must process when reasoning about a case is essential. As discussed previously, it is difficult to express such knowledge simply as production rules. It appears therefore that in modelling legal reasoning, one requires the doctrines of the law (e.g. Habeas Corpus, Mens Rea, negligence); the particular statutes; and the legal arguments and particular facts and circumstances contained in precedents.

Whilst there might be some chance that rule based or logic based systems could capture some of the elements of the statutes, it is highly unlikely that they will be able to capture the remaining requirements. This has led to our interest in IKBALS II as a means of dealing with the subtlety and complexity inherent in legal reasoning problems by experimenting with both deductive and case based reasoning.

2 Problem Definition

IKBALS II deals with the statutory interpretation of the Accident Compensation (General Amendment) Act, 1989, (WorkCare). Emphasis is paid to unsuccessful claims which must be appealed to by the worker's council to the Workcare Appeals Board with a further right of appeal to the Full Bench of the Appeals Board and/or to the Accident Compensation Tribunal. In particular, the system focuses on elements giving rise to a WorkCare entitlement.

The Act gives a worker an entitlement to compensation if s/he suffers an injury arising out of or in the course of employment (S82). Unlike actions at common law, it is not necessary to show fault (negligence). It is however, crucial to the success of any claim that the applicant falls within the following statutory definitions:

Worker- The Act extends the normally understood meaning of employer to include such people as contractors, owner/drivers and commissioned agents. Casual and part time workers are covered in the same way that full time workers are covered;

Injury- This means any physical or mental injury, including industrial deafness and diseases contracted in the course of employment. It also includes the recurrence, aggravation, acceleration, exacerbation or deterioration of any pre-existing injury where the worker's employment was a contributing factor;

Out of or in the course of employment- The 'work connection' is at the core of any claim for compensation. In general, a worker who suffers an injury in the following circumstances is protected by the Act:

1) the injury occurred at work or was caused by work;
2) the injury occurred in circumstances prescribed by s. 83 - deemed worker;
3) a disease caused, aggravated, accelerated, exacerbated or which was made to deteriorate or recur by employment.

In the case of a disputed claim, the lawyer first proceeds to prepare a written submission for the worker in order to present it before the Workcare Appeals Board. IKBALS II helps the lawyer evaluate the merits of a case by keeping an up to date case-base of previously tried cases, (precedents), in addition to expert domain specific knowledge. Lawyers are able to investigate the likely outcome of their input case by comparing it against other previously tried cases which have come before the Appeals Board of the Tribunal by contrasting it against facts that the court identified as significant in determining the case. The system bases its comparison of the case on information obtained from the worker and medical practitioner concerning the circumstances of the injury and the degree of incapacity.

The Workcare legislation is relatively new and hence it is simple to maintain a complete and up-

to-date precedent case base. Nevertheless the legislation has undergone many changes, and hence lawyers find it difficult to keep abreast of the amendments. IKBALS II is capable of helping alleviate many of the problems associated with frequent amendments.

Suppose that an injured worker walks into a solicitor's office requesting advice on the nature of his injury. The worker was injured while returning home from a union meeting. The meeting was originally scheduled on a Friday, but due to difficulties in hiring a hall the meeting was re-scheduled for Sunday. On her return home from the Sunday meeting, the worker was crossing the road when she was struck by a motor vehicle. The worker's doctor claims the worker will be incapacitated for at least twelve weeks. The lawyer, not being too experienced with Workcare will use the system to recommend the best way to pursue the claim. We will refer to this *c*urrent *i*nput *c*ase as [CIC] in the remainder of this paper.

IKBALS II reflects the expertise that goes into making these legal decisions for a wide variety of circumstances. It takes into account the nature of the injury, the degree of incapacity and identifies any work connection.

2.1 Legal Case Based Reasoners

Because of the deficiencies of traditional rule-based expert systems to successfully model the legal analogical process, considerable interest has been focused on precedent case based reasoning. CBR uses past cases, or precedents, to find an interpretation of a current input case based on a) the point of view of the user and b), the intersection of 'similar' features occurring between the set of features present in the input case and the set(s) of features present in precedent cases. From a set of most relevant cases retrieved, a smaller subset of most promising cases(s) is selected by focussing on the importance of shared, relevant similarities. A case based reasoner would then proceed to justify the line of argument by explaining its interpretation. This is distinct from 'problem-solving' or 'planning-oriented' CBR in which past cases are used to find a new plan or course of action by the adaption of a past one [Rissland and Skalak, 1989].

Although CBR is still a research paradigm, it has been shown to offer significant advantages over conventional rule-based expert systems when attempting to reason with experience, or in our case, with previously tried cases. Research issues still confronting CBR include the representation of episodic knowledge, memory organization, indexing, case modification and learning [Slade, 1991]. Despite these technical issues, CBR has been moderately successful in dealing with the subtlety and complexity inherent in legal reasoning problems. The books of [Riesbeck and Schank, 1989] and [Kolodner, 1988] describe numerous legal case based reasoning systems. JUDGE [Bain, 1986], works in the domain of criminal sentencing by modelling a judge who is determining sentences for people convicted of crimes; HYPO [Ashley & Rissland, 1988; Ashley, 1987], does case based reasoning in the area of patent law generating plausible arguments for the prosecution or the defence; and PERSUADER [Sycara, 1987], proposes resolutions for dispute situations.

A legal case based reasoner essentially reasons from previously tried cases, comparing the contextual information in the current input case with that of cases previously tried and entered into the system. Legal classification of the current case is achieved by analogising the facts and circumstances of the current case with those in the case knowledge base (CKB) and then applying similarity metrics in order to retrieve the most 'on-point' cases. Precedent cases in the knowledge base are organized so that the features of the current case can be used to index them. One particular indexing scheme proposed by [Ashley, 1988] in his Phd thesis involved indexing on "dimensions". Dimensions represent the features of a case which have been identified by the court as significant in its determination of a case in a certain way. Ashley's dimensions are composed of:

1) Prerequisites- a list of factual predicates that must be satisfied for the dimension to apply;

2) Focal slots- one or two of the dimension's prerequisites are designated as being indicative of a case's strength along that dimension. These prerequisites make a case stronger or weaker along the dimension;

3) Range Information- tells how a change in focal slot value effects the strength of a party's case along a given dimension.

Although dimensional analysis has been shown to be to some extent domain-sensitive, [Mendelson,1989], a variation of it has been adapted by the authors to index cases that have come before the Accident Compensation Tribunal.

Indices will only help to retrieve a set of potentially relevant precedent cases. The next step is to compare the current case with the retrieved cases, so that we can reject cases that are different to the input situation and determine which of the remaining cases is most similar to the input. The relevance of cases depends on how well they match on each dimension and the importance of the dimension. Dimensional analysis involves characterising the importance of particular antecedents and giving them an associated importance factor. The important antecedents of dimensions are referred to by Ashley as 'focal-slots'.

After the best match has been determined, the lawyer can then engage in constructing a sound case that he will present before the court in favour of his client. This is achieved by allowing the lawyer to strengthen his argument or weaken his opponent's argument by investigating the consequences of adding/subtracting or strengthening/weakening key attribute values of his input case that will effect the subset of applicable dimensions retrieved. This is often referred to as adaption in the literature.

2.2 Choice of Methodology

Rissland and Skalak are currently extending the HYPO CBR system [Ashley and Rissland, 1988] to incorporate rule based reasoning (CABARET) [Rissland and Skalak, 1989]. They use a mixed paradigm approach to the problem of statutory interpretation, combining case-based reasoning with rule-based reasoning (RBR). These two co-reasoners are supervised by their own dedicated monitors which make observations towards the progress of a solution. Observations are reported to a controller that uses the monitors' observations to decide on how i) the system as a whole is to proceed and ii) how the individual co-reasoners are to proceed using the controller's set of 'control heuristics'. The system has been implemented using a frame-based, 'blackboard' expert-system architecture.

At the same time the authors of this paper are extending the hybrid object oriented rule based system of IKBALS to incorporate case-based reasoning [Vossos et al, 1990c]. CBR is not used in its most general form but merely to retrieve relevant cases when the rules are either inadequate or silent in their definition of WorkCare concepts (i.e., the employment relationship, injury or disease, and work related injury).

We view legal reasoning as an attempt to interpret statutes initially through the use of the rules. Hence the basis of any such reasoning system must necessarily be rule/object based. Eventually rules must run out, otherwise the reasoning system would be infinite. It is precisely at this stage that we need to use precedents to guide the legal decision reasoning process, i.e., at the point where the rules themselves prove insufficient in their elicitation of concepts. We hence need techniques to retrieve the relevant cases once reasoning with the rules is exhausted or inconclusive.

It should be noted that any precedent base consists entirely of cases where the rules have proved inadequate. No case which is a clear interpretation of the statute would be litigated under the WorkCare system. Thus in some way, only the exceptional cases which go to litigation are recorded in the precedent base. Hence given the domain of application, the legal reasoning system must primarily be rule/object based. However to deal with certain situations, the system requires the efficient intelligent retrieval of relevant precedents.

In order to successfully combine CBR with the present rule-based architecture of IKBALS, it was important to structure the key elements of both the rules and cases so as to facilitate the efficient interleaving of these knowledge sources. Our technique involves the use of a lattice comprising of object/class structures used to represent both the taxonomic hierarchies that exist when reasoning in the WorkCare domain as well as providing an organization for the clusters of dimensions used to index the cases in the Case Knowledge Base (CKB). Intersecting decision

sets, in the form of rule sets, are also used to represent the collection of legal heuristic knowledge needed to reason in the WorkCare domain. Such a representation leads to a more natural and elegant model of the legal reasoning process in our domain, as well as a large reduction in the number of cases that need to be examined in the Case Knowledge Base (CKB) for relevance during the matching process.

More formally, the lattice formed by the object/class structures consists of classes and objects connected in a fashion that supports direct inheritance and multiple inheritance. These classes have associated data structures and methods. The data structures essentially provide a mechanism for storage of attributes. In effect, these attributes comprise the pre-requisite facts needed to satisfy the dimensions. Furthermore, these attributes (or slots as they are represented in the classes) have a value facet and/or a default value facet associated with them which is used to compare cases in the retrieval process. Methods, on the other hand are stored in slots which have associated if-needed and if-changed facets. These are passed as messages of communication among objects in IKBALS II. For a full discussion about the design of IKBALS, including the inferencing strategies, refer to [Vossos. et al. 1990a and Vossos et al 1990b].

IKBALS II relies on a hypertext engine for its text representation of the actual cases and statutes, as well as its user interface. Hypermedia is a very powerful information representation technology allowing the designer to represent information in a linear, hierarchical or arbitrary connected network. Because of the requirement for lawyers to be able to navigate through the vast amount of precedent cases and legislation, it was decided to use a hypertext engine to drive IKBALS II. This decision means that the lawyer can now browse through related pieces of information by pointing and clicking buttons or key words that appear in a text document, effectively linking him with sources of related information. The lawyer can then trace back through the sources of information (nodes) in order for him to get back to a previous reference. Hypercard v1.1.2 is currently being used for this purpose.

Apart from the representational advantages associated with modelling highly qualitative domains such as law with Hypertext, Hypertext systems also facilitate for rapid prototyping of the end-user system. This feature is tied in with the fact that some hypertext systems are object oriented, permitting developers to build a great deal of the system by 'pointing, clicking and dragging' icons on the screen. HyperCard allows us to quickly and efficiently design and build the user interface as well as some of the knowledge structures, in particular, the full text summaries of all the cases reported with their associated links to other cases and regulations. Nexpert v2 is currently being used as the knowledge engineering environment interfacing with HyperCard under Nexpert's HyperBridge.

3 IKBALS II

Legal practitioners are in general not solely interested in the likely outcome of a court case. They are more concerned with providing support to argue the case that their client wishes to put forward, although admittedly they do encourage their client to pursue a path that has a reasonable chance of success. Hence what is really needed is a litigation support system which will allow the lawyer to navigate through the vast amount of legal sources available, permitting him to find the relevant rules and precedent cases to successfully argue his client's case.

IKBALS II works thus: The input to IKBALS II is a problem situation such as the [CIC] example described above. The situation is not expressed in natural language but rather entered via a series of templates. The system then proceeds to:

(a) identify the relevant legal norms;

(b) attempts to categorize the worker' claim for compensation under the Act by chaining backward and forward rules in order to satisfy the necessary and sufficient conditions required to satisfy the legal norms;

(c) identify relevant precedent cases for concepts or terms that cannot be resolved given the facts of the case and the rule set(s) available, in a reliable efficient manner, from a very large number of possible cases;

(d) having identified the relevant precedent cases, it proceeds to compare it with other cases in the system, and in particular, distinguish the

current case from others whose conclusions are to the contrary;

(e) suggest arguments which can be made in favour of the current case and the facts and precedents which can be cited to support them.

3.1 Knowledge Representation

IKBALS II represents three types of domain knowledge that allow it to perform this kind of CBR:

- the legal precedents which are represented and indexed in the Case Knowledge Base, (CKB);

- the statutes and regulations which are contained in the Statute Knowledge Base, (SKB), and the

- expert heuristic knowledge needed to reason with both the precedents and legislation. This type of knowledge is represented in two places, as inferential knowledge contained in rule sets in the Case Rule Base, (CRB), and as messages contained in the meta-slots of various object/classes.

Currently, the IKBALS II CKB contains thirty real legal cases with twenty dimensions that are the index to cases in the CKB. The SKB contains three rule sets with close to fifty rules per set. Most of the meta-knowledge used to drive the system is contained in the meta-slots of certain object/classes.

Faster indexing of similar cases is facilitated by classifying resolved cases under as many case types as possible in order to maximize the range of applicable precedents. In particular, resolved cases are indexed by ptas [Vossos et al, 1990c]. Ptas differ from Ashley's dimensions only in representation. Each pta represents a way of arguing about a case. Points To Argue are the subset of facts of a case that the court deemed significant in determining the case's outcome; that is, the subset of facts that were relied upon in identifying the significant legal principle in question. Ptas are not a summary of a case, and would vary depending on how the lawyer actually argues a case. Hence the same case could have two or more ptas depending on how the council presented it. In our [CIC] case, the following ptas are retrieved: action-incidental-to-employment, action-agreed-to-by-employer. Whilst the points meeting-not-on-employers-premises, and meeting-on-a-day-off are elements of the case, they are not ptas since if the employer agrees to his workers attending a union meetings during normal working hours then these points are irrelevant to the case.

In IKBALS, which is a rule/object based system, problems are input in a form, based on what the Workcare experts have deemed 'important' attributes in determining Workcare disputes. Note, that the choice of these attributes is quite crucial to the success of the system. IKBALS II uses the rule/object base of IKBALS. When the current legal problem requiring a solution is input, a rule based engine determines which ptas apply to the current case. These ptas are then used to retrieve those cases that are indexed under the same ptas in the system. The pta's prerequisites determine what features to look for in a case. Since a case can be indexed under several different ptas, after a number of candidate resolved cases have been retrieved, the most similar case(s) is chosen. In our [CIC] case, the three cases retrieved as being similar were [Johansen v ACC], [ACC v Gardiner] and [Riego v ACC]. The focus of any comparison would be based on the action-agreed-to-by-employer pta.

In [Johansen v ACC] the worker was injured returning home form a strike meeting. The issues which arose were whether the worker's contract of employment was subsisting at the time of the injury, and if so, whether the injury arose out of that employment. Whilst the court decided that the worker contract was not terminated, it was also satisfied that the employer in no way condemned or gave permission either explicitly or implicitly to any person to strike or attend the strike meeting. Hence the Tribunal was not satisfied that there was a nexus between the worker's attendance at the meeting whilst on strike and her employment so as to establish that her attendance at the meeting was reasonably incidental to her employment.

In [ACC v Gardiner], the worker was injured while attending a union meeting which was not on the employer's premises and for which she

was not being paid. Nevertheless, the Tribunal found that the injury occurred whilst the worker was performing an act incidental to the course of employment since she had the permission of the employer and indeed used the employer's vehicle to attend the meeting. The meeting was specifically called to discuss superannuation, an issue effecting employment conditions.

The case of [Riego v ACC] was decided on similar grounds. In this case, a worker was responding to a request made by the secretary of the union to come to work for the purposes of attending a union meeting. The worker's employer also requested the employee to attend the meeting. The meeting was scheduled for the worker's rostered day off, when the applicant was injured in a traffic accident on her way to work. The issue before the Tribunal was whether the injury which occurred on the way to the workplace was in the course of employment. The Tribunal found that her travel from her place of residence to the workplace was incidental to her employment obligations.

For each of the most relevant precedents, IKBALS II then proceeds to justify that the outcome of the current case should, (or shouldn't be) the same as the precedent's outcome. It does so by drawing the analogy between the two cases, focussing on their important similarities and differences. At this stage, we envisage using heuristic control rules similar to those used by the agenda-based controller of CABARET [Rissland and Skalak,1989] to support or distinguish particular lines of argument.

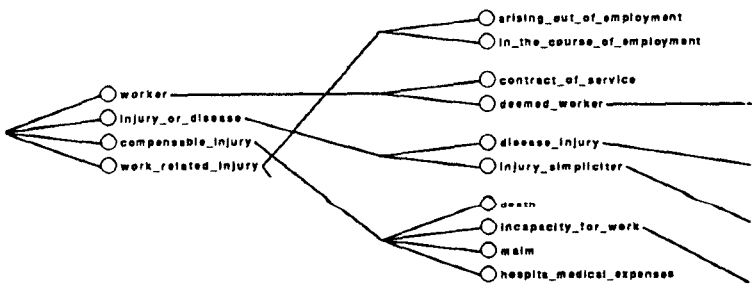

Fig. 1 An abstracted view of part of the CKB.

The CKB can be viewed as containing the indexes into the precedents and statutes. Hence precedents and statutes have been abstracted in two different ways: one used by the knowledge base and the lawyer for matching and comparing similar cases, and the other used exclusively by the lawyer to scan the legal sources in their entirety: the hypertext extension. Essentially, the system is designed around a two stage working model, an Investigation module and an Analysis module

3.2 The Investigation Model

The Investigation module starts by trying to satisfy the statutory elements which give rise to a WorkCare entitlement given the significant facts of the current input case. This is done by forward-chaining and backward chaining normalized statutory rules in the SKB. When a particular statutory predicate cannot be resolved, the module then proceeds to interpret this concept in the context of previously tried cases which have come before the Accident Compensation Tribunal. Unfortunately, we have not included cases litigated in front of superior courts, but nevertheless our system's heuristics have been learnt from Supreme and High Court cases. This is done by determining the relevant ptas which can be used to index past similar cases. It then proceeds to retrieve all the cases in the CKB which are indexed by those ptas. Furthermore, all the relevant sections of the legislation will also be determined at this stage. Associated with each pta (represented as a rule in the system) are necessary and sufficient conditions identified by the court as being significant in determining the applicability of a particular line of argument. These key factors constitute the head of a pta rule which is used to identify applicable indices.

In general, the holding of a court in a particular dispute is dependent on the subset of facts deemed significant in identifying the relevant norms which are applicable in the dispute. Alternatively, the holding of the court is dependent on the subset of slot attribute values which in effect influences the subset of ptas retrieved.

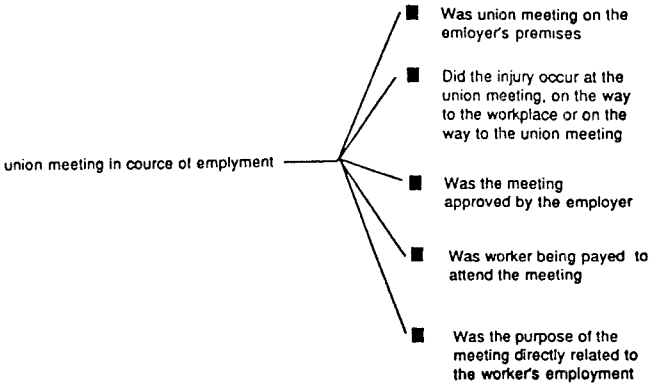

Fig 2. The 'union meeting in course of employment' pta with its associated antecedents.

The SKB determine which ptas apply to the current input case by activating and deactivating the following three rule sets in the CRB in sequence:

RuleSet1: most on point ptas;

RuleSet2: near miss ptas;

RuleSet3: connected ptas

RuleSet1 retrieves ptas which have all their pre-requisites satisfied. That is, all the relevant facts the court identified in making its decision in a previously tried case are present in this similar input case. To a first approximation, the cases retrieved by this subset of ptas are the most relevant.

RuleSet2 and RuleSet3 are provided in order to be used in the Analysis module to experiment with "what happens if" situations a lawyer might want to pose to the system. It facilitates hypothetical reasoning, permitting the lawyer to see what happens when he either adds/deletes or strengthens/weakens certain circumstances about the current input case. RuleSet2 essentially contains the same rules as RuleSet1, except for the fact that it searches for ptas which have narrowly missed out on being applicable because of some missing fact(s) which have a low priority level. This is determined by identifying facts that might be able to be proved given the current circumstances of the current input case. RuleSet3 contains expert heuristic knowledge about chasing up certain issues given the current fact situation.

3.3 The Analysis Module

The aim here is for the lawyer to experiment with 'what happens if' situations, permitting him to construct sound favourable arguments for his client. This is done by giving the lawyer an abstracted view of the CKB, allowing him to directly change the attribute values of his current input case (represented as an object inheriting down form related pta classes).

The lawyer can then proceed to strengthen his argument. This is achieved by allowing him to quickly and efficiently access the relevant legal sources. In particular, the lawyer analyses his case by:

i) having direct access to the relevant sections of the legislation (via hypertext)

ii) being able to quickly locate and retrieve precedent case documents with hypertext links to other related documents. This is done by simply clicking on key highlighted works within the text document.

iii) being able to view the slots of a retrieved case in the CKB alongside the current input case's slots. This gives the lawyer a better understanding of the comparative merits of his client's case against others in the system.

iv) introducing new supporting ptas (near miss ptas, connected ptas etc) to probe the cluster of dimensions to support the lawyer's case.

The lawyer can also discount contrary cases cited by the opposition, by showing how the opposition's cited case differs from the current input case. The program generates the distinctions by looking for other dimensions which apply to both the current input case and favourable precedent cases in which the relevance of the current input case is stronger.

4 Conclusion

Truly intelligent legal knowledge based systems need to provide advice to lawyers with which to

construct sound legal arguments. They need to argue with precedents as well as statutes.

We have provided an architecture that allows for the integration of a hybrid object-oriented rule based system together with case-based reasoning. Our prototype IKBALS II has utilised three types of domain knowledge (legal precedents, statutes and regulations and expert heuristic knowledge). It has an Investigation module (to collect facts of the case and retrieve relevant precedents) and an Analysis module (used to engage in litigation support). In particular we have highlighted how like cases can be retrieved (i.e. compared for relevancy), by using ptas. Ptas also allow us to compare and contrast cases along similar ptas.

Acknowledgements:

The IKBALS II project has two extremely willing domain experts, Graham Taylor, of Tony O'Brien and Associates Solicitors, and Ms Heather Aldred, Law tutor at Monash University.

Taylor is an acknowledged expert on the Accident Compensation Act 1985, and he consults both employers and employees. Besides his considerable legal skills, Taylor has a good appreciation of computers. For the purpose of this project, Taylor helped in developing the knowledge structure of IKBALS II. The model was based on what he considered to be the legislation's purpose.

Aldred gave important advice with regard to the interpretation of the Accident Compensation Act and helped with explaining the associated cases.

5 References

Ashley, K. D. (1988), "Arguing by Analogy in Law: A Case Based Model", Analogical Reasoning, Kluwer Academic Publishers, pp 205-224.

Ashley, K.D., and Rissland, E.L. (1988), "A Case-Based Approach to Modelling Expertise", in IEEE Expert, Fall , pp 70-77.

Bain, W. M. (1986), "Case Based Reasoning: A Computer Model of Subjective Assessment", PhD thesis, Yale University.

Burton, S. J. (1985), "An Introduction to Law and Legal Reasoning", Little, Brown and Company.

Gardner, A. (1987) "An Artificial Intelligence Approach to Legal Reasoning", Bradford/MIT Press .

Hafner, C.D., Berman, D.H., (1990) "Artificial Intelligence and Law", Encyclopedia of Computer Science and Technology, Marcel Dekker Inc, Vol. 22, Supplement 77, pp 43-65.

Kolodner, J. (1988), "Proceedings of a Workshop on Case Based Reasoning", May 1988, Clearwater Beach, Florida.

Kowalski, R.A. (1979), "Logic for Problem Solving", North Holland.

Levi, E.H. (1948), "An Introduction to Legal Reasoning", The University of Chicago Press.

Mendelson, S. (1989), "An Attempted Dimensional Analysis of the Law Governing Government Appeals in Criminal Cases", Proceedings of the Second International Conference on Artificial Intelligence and Law, 1989, ACM Press, pp 128-137.

Riesbeck, C. K. and Schank, R. C. (1989) "Inside Case Based Reasoning", Lawrence Erlbaum Associates Publishers.

Rissland, E.L., Skalak, D.B. (1989) "Interpreting Statutory Predicates" Proceedings of the Second International Conference on Artificial Intelligence and Law, 1989, ACM Press, pp 46-53.

Schlobohm, D.A. and Waterman, D.A. (1987), "Explanation for an Expert System that Performs Estate Planning", Proceedings First International Conference Artificial Intelligence and Law, ACM Press, 00 18-27.

Sergot, M.J., Sadri, F., Kowalski, R.A., Kriwaczek, F., Hammond, P. and Cory, H.T. (1986), "The British Nationality Act as a Logic Program" Communications ACM, Volume 29(5) , pp 370-386.

Skalak, D.B., (1989) "Taking Advantage of Models for Legal Classification", Proceedings of

the Second International Conference on Artificial Intelligence and Law, 1989, ACM Press, pp 234-241.

Slade, S. "Case-Based Reasoning: A Research Paradigm", AI Magazine, Vol. 12, No. 1, Spring 1991, pp 42-55.

Susskind,R.E. (1987) Expert Systems in Law, (A Jurisprudential Inquiry), Clarendon Press, Oxford.

Sycara, E. P., (1987), "Resolving Adversarial Conflicts: An Approach to Integrating Case Based and Analytic Methods", Technical Report GIT-ICS-87/26, Georgia Institute of Technology, School of Information and Computer Science, Atlanta, GA.

Tyree, A. L., Greenleaf, G. and Mowbray, A., (1987), "Legal reasoning: the problem of precedent", Proc. Conf. AJAI, Sydney, November 1987, pp 419-432.

Vossos, G., Dillon, T., Zeleznikow, J., Taylor, G. (1990a), "An Object Oriented System for Legal Reasoning - IKBALS", Proceedings of the Tenth International Workshop on Expert Systems and their Applications, pp 741-754, Avignon, France, 1990

Vossos, G., Dillon, T., Zeleznikow, J., Taylor, G. (1990b), "The Use of Object Oriented Principles to Develop Intelligent Legal Reasoning Systems", Australian Computer Journal, Vol. 23, No. 1, February 1991, pp 2-10.

Vossos, G., Zeleznikow, J., Dillon, T. (1990c), "Combining Analogical and Deductive Reasoning in Legal Knowledge Base systems - IKBALS II", Proceedings of the third international conference of the Dutch Foundation of Legal Knowledge Systems (JURIX-90), Legal Knowledge Based Systems- Aims for Research and Development, Koninklijke-Vermande 1991, pp 97-105.

Waterman, D.A., Paul, J. and Peterson, M. (1986), "Expert Systems for Legal Decision Making", in Applications of Expert Systems, The Rand Corporation .