VICTORIA UNIVERSITY OF WELLINGTON

Department of Computer Science

PO Box 600
Wellington
New Zealand

Tel: +64 4 471 5328
Fax: +64 4 495 5232
Internet: Tech.Reports@comp.vuw.ac.nz

Argument Representation and Conceptual Retrieval for Litigation Support

Paul Hosking

Technical Report CS-TR-94/19
December, 1994

Abstract

Case preparation and *evidence management* are fundamental tasks faced by litigators. These tasks are not well supported by existing litigation support software. Case preparation involves constructing and evaluating arguments in support of the client; current software packages provide only limited assistance, as they are unable to represent arguments. Evidence management involves organising and retrieving documents associated with the case; existing document retrieval systems employ only crude keyword-based indexing schemes that achieve poor rates of recall.

This dissertation describes a prototype system that uses an explicit representation of arguments to enable lawyers to construct, edit, browse, and evaluate arguments graphically. The representation includes a well-defined syntax and a semantics characterising the degree of support for an argument. Argument construction is facilitated by the provision of a set of *argument packets* encoding standard patterns of inference, which may be retrieved and copied into arguments as they are constructed. Browsing is facilitated by a set of *argument views* that enable the user to focus on specific portions of the argument. Argument evaluation is supported by a further set of argument views that evaluate the level of support for a given fact in the argument and highlight areas of weakness. Documents associated with cases are stored and retrieved in an innovative manner, using the structure of the argument as an index to a hypertext database.

Publishing Information

This report was also submitted to the Victoria University of Wellington in fulfilment of the requirements for the degree of Master of Science.

Author Information

Paul Hosking is an Applications Developer for SCIENZ Softwright Limited.

Acknowledgements

I would like to thank my supervisor, Eric Jones, for the guidance and support that he has given me during the course of this project, and for the comments and constructive criticism that he has offered during the writing of this dissertation.

Thanks also to the members of the AI Group for the serious (and sometimes not so serious) discussions relating to this project and other topics. Thanks in particular to Linton Miller and Aaron Roydhouse for keeping my spirits up over the last few months, when it felt as though I would never finish.

Finally, I would especially like to thank my parents and family for their support during the years that I have been studying. I could not have done this without them.

Contents

Chapter 1

Introduction

1.1 The Problem

Two fundamental tasks faced by litigators are *case preparation* and *evidence management*. When preparing for trial, lawyers must construct arguments that support their client's position, and anticipate any counter-arguments put forward by the opposing counsel. In doing so, the lawyer must argue that a judgment in favour of the client would be consistent with the law as it stands, by citing precedent cases whose facts are similar to the case at hand and whose judgments were in favour of the client's position. Lawyers must also have some means of organising the mass of evidence associated with the case. Evidence is typically contained in documents, in the form of transcripts of past precedents, transcripts of the testimony of witnesses, copies of contracts, photographic images, and so forth. Documents containing evidence for the case at hand should be stored in a manner that allows easy retrieval. This is especially important in complex cases such as the Equiticorp fraud trial in New Zealand, in which over 43,000 pages of documents were entered as evidence [Greenleaf 93].

A task related to case preparation is that of *case analysis*. Analysis of a historical case by students of the law involves reconstructing the arguments presented in the case. An understanding of the legal issues involved in a case can be obtained by examining both the facts of the case and how those facts combined to support the conclusion reached in the case.

To illustrate the issues involved in case preparation, case analysis, and evidence management, consider the case *Henderson v BNZ*—a recent case decided in the High Court in Whangarei, New Zealand [Blanchard 92]. (This case will be used as a running example throughout this dissertation.) In early 1988, Alexander Henderson owed approximately \$90,000 to the Bank of New Zealand. This sum was secured over a property worth less than \$45,000. After failing to meet several repayment dates, the BNZ offered him three separate repayment options in writing. The first two options involved clearing part of the debt via a mortgagee sale of the property, differing only with regard to the means by which Mr Henderson would pay the residual debt. The third option, however, made no mention of a mortgagee sale:

> "Mrs Henderson may be prepared to make an offer in full and final settlement of our debt to avoid her son's bankruptcy. We would consider any reasonable figure, say \$30,000."

All three options required Henderson to pay \$200 per week in the interim to offset the interest accruing on the debt. These payments were already being made.

Henderson immediately realised that option 3 was the most favourable to him. He discussed it with his mother, whereupon she instructed her solicitor to accept the third option. The solicitor sent a letter to the BNZ stating:

> "... she is in a position to pay $28,000 ... in full and final settlement of her son's debt ... If such an arrangement can be reached would you kindly advise."

When the BNZ received the letter, it realised that the third option had been misinterpreted—it was intended to apply only after the bank had recovered part of the debt via a sale of the mortgaged property, as per the first two options. The BNZ rejected Mrs Henderson's "counter offer" and sold the property. Mr Henderson sued the bank for damages relating to the breach of a contract that he alleged was formed between himself, his mother and the bank.

There were three main issues to be decided:

1. Was the third repayment option an offer by the BNZ to Mr Henderson and his mother?
2. If so, was the solicitor's letter an acceptance of the bank's offer or a counter offer that destroyed the original offer?
3. If the letter was an acceptance, did Mr Henderson or his mother provide sufficient consideration for the offer?

When preparing for trial, the lawyers for each side must construct an argument that answers these questions in favour of their client. Mr Henderson's lawyer must argue that all three questions should be answered in the affirmative, whereas the lawyer for the BNZ must argue that at least one of the questions should be answered in the negative. Case preparation is a complex, time consuming task. A system for planning and constructing arguments could significantly reduce the complexity of this task. However, the planning of arguments is an almost completely unsupported activity [Smolensky 87]. One reason for this may be the wide variety of problems faced by lawyers when preparing for trial. A system supporting case preparation must enable lawyers to

- Explicitly represent the structure of the argument for the case—the way in which the facts combine to support or rebut the issue in question. For example, in *Henderson v BNZ*, the system must represent (among other things) the requirement that in order for a contract to have been formed between the BNZ and Mr Henderson, there must have been an offer by the BNZ that was accepted by Mr Henderson, and valid consideration must have been provided by Mr Henderson in exchange for the offer.
- Ensure that all the issues that need to be considered are included in the argument. There would be no point in the counsel for Mr Henderson arguing that there was an offer made by the BNZ and that the offer was accepted by the Hendersons if the argument failed to consider whether or not valid consideration was provided.
- Provide backing for the argument in the form of references to relevant past precedents—previous cases whose facts are similar to the case at hand. Cases that are potentially relevant to *Henderson v BNZ* include *Foakes v Beer* [Foakes 84] and *Williams v Roffey Bros* [Williams 91] (see below).
- Help make explicit any implicit assumptions underlying the argument. Such assumptions often prove to be unfounded and thus susceptible to rebuttals.

- Evaluate the argument so that any weaknesses can be eliminated.

An integral part of argument construction is the use of precedents as backing for the argument. Precedents in favour of the client's position can be used to construct an argument in support of the client, justified by citations of the precedent cases. For example, in *Henderson v BNZ*, the lawyer for the BNZ cited *Foakes v Beer*, a case in which a part payment of a previous debt was judged to be insufficient consideration for the waiving of the requirement to pay the remainder of the debt. This case was cited in support of the argument that no consideration was provided by Mr Henderson.

The constructed argument must also distinguish precedents in favour of the other party by finding features of the current case that are not present in the precedents and vice-versa. For example, in *Williams v Roffey Bros.*, the court ruled that an agreement to perform an existing contractual obligation could be regarded as sufficient consideration in certain circumstances. In *Henderson v BNZ*, the BNZ's lawyer argued that this ruling was not applicable to the current situation, as the purported agreement did not result in the BNZ obtaining an additional benefit or Mr Henderson incurring an additional detriment.

Precedents that are referred to by the argument must be stored in a manner that allows easy retrieval. Documents containing physical evidence, such as the repayment options letter and the letter from Mrs Henderson's solicitor, should also be stored for subsequent use. Conventional document retrieval systems provide full text search as their only means of retrieval. Documents are indexed solely in terms of the words they contain, and retrieved by matching text strings connected by boolean operators against the index. For example, when preparing for *Henderson v BNZ*, the lawyers may have searched for documents containing the words "invitation to treat" and "counter offer". Such systems, although appearing to perform well, typically have a poor rate of recall due to the inherent limitations of keyword search [Bing 78, Blair 85, Hafner 87]. For example, if there are any cases that refer to "indication of willingness to negotiate" instead of "invitation to treat", the above search will fail to retrieve those cases.

1.2 A Potential Solution: Conceptual Retrieval Using Argument Representations

A hybrid approach that has application in both document retrieval and legal reasoning is *conceptual retrieval*. Under this approach, documents are organised according to the legal concepts that are relevant to the information contained in the documents [Hafner 78, McCarty 83, Hafner 87]. The advantage of such an approach is that documents related to a specific legal concept can be retrieved directly, rather than by guessing how the author of the document described the concept in words. Having retrieved the documents, the cases associated with the documents can be used to generate an argument for supporting the presence or absence of the associated legal concept in the case at hand [Ashley 87, Rissland 89].

This report describes a system that uses representations of the structure of a legal argument as the basis for conceptual retrieval. The system explicitly represents the structure of the legal argument underpinning a case and uses this argument structure to index a hypertext database of documents associated with the case. The system allows users to construct and display argument structures using one of a number of graphical notations. Documents associated with the case can be attached to nodes in the argument structure, and later retrieved by traversing the argument structure. Documents relating to past precedents can be retrieved by traversing the structure of *argument packets*: generalised, self-contained argument structures relating to specific points of law. By copying the relevant argument packets into the argument structure, users can quickly generate an argument based upon relevant past precedents.

The system provides a novel alternative to previous systems for conceptual retrieval that do not incorporate argument representation. By focusing on the structure of the argument as opposed to its content, the system avoids the necessity of developing a detailed knowledge base for the legal domain, thus enabling the system to scale well. Furthermore, graphical representations of argument structures provide an effective interface for document retrieval. Finally, the system enables lawyers to simultaneously construct an argument in favour of their client and organise documents relating to the case.

The system has a number of applications. The system can assist lawyers in preparing for trial, by allowing them to construct and analyse an argument in favour of their client. The system facilitates evidence management by enabling lawyers to store and retrieve documents relating to the trial. The system should also be useful as a teaching aid. Students can increase their knowledge of a particular domain of law by browsing through the argument packets and associated precedents for the domain. Students can also improve their understanding of legal issues by analysing the argument structure of historical cases.

1.3 Outline of Thesis

Chapter 2 examines and evaluates previous research in AI and in law dealing with conceptual retrieval and argument representation.

Chapter 3 proposes a new model of argument representation that builds on the strengths of the existing representations while eliminating a number of weaknesses.

Chapter 4 presents an overview of an implemented system that enables users to construct and edit arguments using the argument representation. A summary of the system architecture is presented, followed by a description of how the system may be used to store and retrieve documents.

Chapter 5 describes how the system can be used to construct arguments, and how argument packets can simplify the construction of complex arguments.

Chapter 6 describes a number of *argument views*: mechanisms that enable the user to collapse portions of the argument or to evaluate the argument in terms of how well it is supported.

Chapter 7 examines how the user can change the way in which the argument is presented graphically on the screen, and briefly describes some of the implementation issues associated with the system.

Chapter 8 summarises the significant features of the representation and the system, concluding with a discussion of possible future research directions.

Chapter 2

Previous Research

Research in the field of Artificial Intelligence and law has been steadily increasing over the last twenty years [Rissland 90]. This chapter describes past research relating to conceptual retrieval and argument representation. Section 2.1 examines a proposal for a system incorporating conceptual retrieval. Section 2.2 describes recent developments in a related area: case-based reasoning applied to law. Section 2.3 examines and evaluates existing models of argument representation.

2.1 Conceptual Retrieval

One of the earliest systems to feature conceptual retrieval was proposed (but apparently not implemented) by Carole Hafner [Hafner 78, Hafner 87]. Hafner's system provides support for conceptual retrieval via two different mechanisms: *individual case descriptors* and *issue dependency trees*. Each case in the knowledge base is assigned a *case descriptor*—a frame-based structure that is used to encode summary information about the case, such as

- *The legal basis*: the parties and cause of action involved in the case.
- *The background to the case*: the underlying facts of the case.
- *The issues involved*: a set of legal questions upon which the parties disagree.
- *The procedural context*: whether the case is an appeal and, if so, the decision being appealed.
- *The holdings of the court*: decisions on all of the issues.
- *Legal theory*: how the issues were related to the result.

Case descriptors enable users to retrieve cases on the basis of the concepts associated with the case, by matching a representation of a concept with the knowledge contained in the slots of the case descriptor.

The second means of retrieving cases described by Hafner is provided by *issue dependency trees*—structures that represent the legal issues and factors involved in a particular legal domain and how they are related. Each node and link in the issue dependency tree may have cases attached to them. Cases attached to nodes are those in which the issue or factor represented by the node was important. Cases attached to links are those in which the issue represented by the upper node was decided and the issue or factor represented by the lower node was relevant to the decision. Each set of cases is structured according to the decision made by the court (for or against the plaintiff) and whether or not the issue or factor was present. The issue dependency tree effectively provides an index into the case law collection based on legal concepts.

Although the case descriptors and issue dependency tree provide a useful means of indexing documents, the proposed system has several deficiencies. The expressiveness of the issue dependency tree is limited—for example, it cannot distinguish between conjunctive and disjunctive support for a factor or issue. Case descriptors require a detailed model of the legal domain, which is difficult to develop due to the uncertainty associated with the application of many concepts in the legal domain. Many concepts are "open-textured", having no concrete description [Gardner 87]. Furthermore, it is not apparent how the case descriptors are used in conjunction with the issue dependency trees. Finally, it is not clear whether or not Hafner actually implemented the system.

2.2 Case-Based Reasoning

Case-based reasoning (hereafter referred to as "CBR") is a technique in AI for using past cases to construct a solution to a problem or to justify an existing solution [Hammond 86, Ashley 87, Rissland 89]. CBR is effectively a form of conceptual retrieval, where the objects retrieved are representations of cases as opposed to case transcripts. The Common Law can be seen to be a form of *precedent-based CBR* [Rissland 89], in which the problem is to determine whether the defendant is innocent or guilty and the solution is a decision either way, justified by a detailed argument involving past cases that are relevant to the current situation. The precedent-based nature of the Common Law has lead a number of researchers to develop systems of legal reasoning based upon CBR.

One of the first CBR systems to be developed was McCarty and Sridharan's TAXMAN system [McCarty 81]. TAXMAN represents concepts by a prototypical description and a series of deformations of the prototype, where deformations are made by mapping one concrete description into another. TAXMAN allows arguments to be made by analogy by using deformations to incrementally move from a useful precedent to the case in question. The same method can be used to distinguish the facts of a case from those of a precedent. However TAXMAN requires cases to be represented in great detail at a low level.

The next significant development was the model of legal reasoning proposed by Gardner [Gardner 87]. Gardner's system contains a rich knowledge base for her chosen domain (contract law), consisting of a set of rules relating to the concepts of offer and acceptance, a state transition network for the parties involved in the contract, common-sense knowledge in the form of object hierarchies, and a set of positive and negative examples to represent how "open-textured" concepts have been interpreted by the courts in the past. Each example describes patterns of facts abstracted from the judgments of one or more cases. The system first attempts to answer questions using the non-example based knowledge. If this succeeds, then the results are checked against all relevant examples, otherwise the system attempts to derive an answer from the examples. In either case, if the system is able to match the current situation to both positive and negative examples of a concept, the question is considered to be "hard" and no answer is returned. Thus, although Gardner's use of examples embodies some aspects of CBR, her system does not have any means of dealing with situations involving conflicting precedents.

The problem of conflicting precedents was addressed by Ashley's HYPO system for legal reasoning in the domain of trade secrets law [Rissland 84, Ashley 87, Ashley 88]. HYPO contains a large knowledge base of cases indexed by *dimensions*—data structures used to represent important factors that have been identified in previous cases as being indicative of the strength of a party's argument. Each dimension has a set of prerequisites that must be true of the problem case for the dimension to be applicable, and focal slots that describe how a case is made stronger or weaker with respect to the dimension. An example of a dimension in the domain of trade secrets law is the "Accessible by Others" dimension, whose focal slot is the number of people outside the company that had access to the information that is claimed to be a trade secret. The higher this number, the weaker the plaintiff's argument. The relevancy or "on-pointness" of past precedents with respect to the problem case is assessed by comparing the set of dimensions present in the past precedents with the set of dimensions present in the problem case. A precedent is relevant to the problem case if it shares at least one dimension with the problem case. One precedent is more relevant than another if it has more dimensions in common with the problem case than does the other precedent. HYPO uses a *claim lattice* [Ashley 87] to find the "most on point" cases for the plaintiff and defendant, which are then used as the basis for each party's arguments. HYPO constructs an argument for both parties by citing cases that support each party's position and then responding to each case citation by attempting to distinguish the cited case from the case at hand. This typically involves finding dimensions that the cases do not have in common or that differ markedly in strength between the two cases. An alternative means of attacking the support provided by a cited case is to find a more on point case with the opposite result. When HYPO wishes to explore how modifying a particular aspect of the case will affect the strength of the argument, it generates a hypothetical case whose facts are modified to support that aspect of the case, and then constructs an argument for the hypothetical case as before.

The most recent development in CBR systems for legal reasoning has been to combine CBR with rule-based reasoning. A multi-paradigm approach seems to be appropriate because there are many areas in the law that cannot be represented via a completely case-oriented approach or via a completely rule-oriented approach. For example, complex legal statutes can be primarily encoded in the form of logical rules. However, many of the rules will contain references to "open-textured" predicates that are not defined in the statute, such as "regular use". The meaning of these predicates can only be determined by examining the cases where they have been pivotal to the result. Two systems that use both cases and rules to perform legal reasoning are Skalak and Rissland's CABARET system [Skalak 92] and Branting's GREBE system [Branting 91].

2.3 Argument Representation

2.3.1 The Nature of Legal Arguments

The type of argument that is employed by a lawyer in a case differs according to whether the setting is a court of original jurisdiction or a court of appeal. In a court of original jurisdiction, the primary issue is to resolve *questions of fact*—disagreements relating to descriptions of previous events. Each side attempts to convince the court of their side of the story, by describing the facts as they believe them to be and providing evidence to support their claims. The importance of evidence is emphasised by the fact that the lawyers cannot present their arguments until all of the evidence has been presented or "admitted" to the court. Furthermore, there are strict rules relating to the admissibility of evidence—what kind of evidence the jury is allowed to hear. The type of evidence presented in court varies from the testimony of eye-witnesses (which may be first-hand knowledge of events or "circumstantial" evidence that supports inferences to events) to the opinions of experts, and physical evidence such as documents, photographs etc.

In an appellate court, on the other hand, the facts have already been established. The primary issue is to resolve *questions of law*—areas in which the law is vague or there exist conflicting precedents. The court is bound to follow the rules contained in legal statutes and the rules developed over the years through common law. In particular, the decision of the court must conform to the precedents established by decisions in previous cases that involved similar circumstances. If there are potentially relevant precedents with conflicting outcomes, then the court must determine which of the precedents are most relevant to the current case. One means of choosing between precedents is given by the principle of *stare decisis*, which states that courts should prefer the decisions of higher courts to those of courts below them in the court hierarchy. The argument constructed by the court will draw analogies between the current case and the relevant precedents by citing shared features, and will distinguish the other precedents by citing features of the current case that are not present in those precedents. In some cases, the court will derive a new rule from the rules contained in the existing precedents, and thus the decision becomes a new precedent that must also be considered in future cases.

The distinct nature of the two types of arguments employed in law means that a representation for legal arguments must be rich enough to describe the unique features of both. The representation must therefore allow the user to

- Distinguish between the types of evidence that may be presented in court.
- Encode the relationships between the facts and how they combine to produce the argument as a whole.
- Use past precedents as a means of justifying portions of the argument.

The representation should also be flexible and easy to use.

The representation focuses on the *structure* of the argument as opposed to its *content*. The representation is thus *semi-formal*: information about the argument structure is represented formally, while argument content is represented informally (c.f. [Smolensky 87]). The following sections describe and evaluate other semi-formal models of argument representation.

2.3.2 The Wigmore Notation

One means of argument representation is that described by Wigmore. Wigmore argued that the mind can only assimilate a limited amount of material at any one time. By reducing an argument to its constituent parts and identifying the relationships between the parts, the mind can assimilate them step by step "until the mind can consciously juxtapose them with due attention to each, so as to produce its single final idea" [Wigmore 13]. Wigmore described a means by which argument structures could be "charted" using a complex vocabulary of symbols, each representing a proposition in the argument. The symbols are arranged in a hierarchical fashion, with the *ultimate probandum*[1] at the top of the tree and the evidence at the leaves. Lines drawn between the symbols show relationships between propositions, and

[1] The top-level hypothesis e.g. "A contract was formed between Smith and Jones".

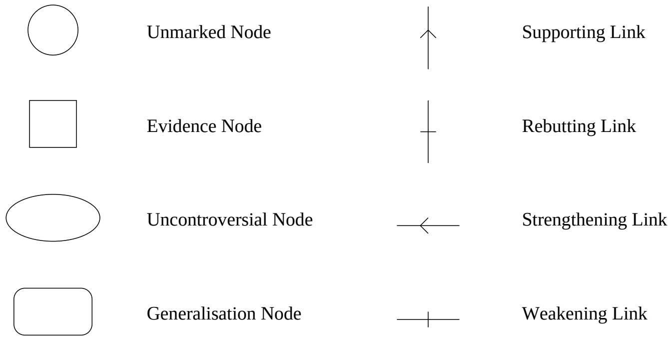


Figure 2.1: Node and link types in the updated Wigmore notation.

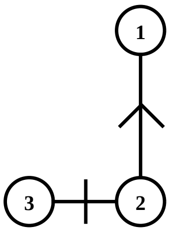
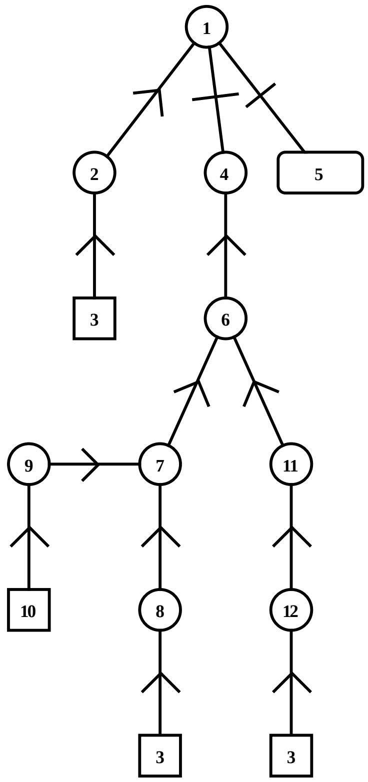


Figure 2.2: Example of a weakening link.

each symbol is labelled to show whether or not the person constructing the diagram believes the corresponding proposition.

The vocabulary of symbols contained in the original Wigmore notation is overly complex—for example, Wigmore uses different symbols depending upon whether the proposition is offered by the plaintiff or the defendant, and whether the proposition is positive or negative. Robertson [Robertson 90] proposes a simplified version of the Wigmore notation with a vocabulary of symbols half the size of the original, eliminating the distinctions mentioned above. Robertson also introduces an explicit symbol for propositions that are uncontroversial. The resulting simplified notation (hereafter referred to as the updated Wigmore notation or simply as the Wigmore notation, as opposed to the *original* Wigmore notation) consists of four node types for representing facts and four link types for representing relationships between facts, as displayed in figure 2.1.

A square node represents a proposition relating to the presentation of evidence, as distinct from facts that are deemed to be uncontroversial or common knowledge (oval node), facts representing generalisations (rectangular node with rounded corners) and all other facts (circular node). A vertical link connecting two nodes indicates either that the lower node supports the upper node (indicated by an arrowhead attached to the link) or that the lower node rebuts the upper node (indicated by a horizontal bar across the link). A horizontal link connecting two nodes indicates either that one node strengthens a relationship between the second node and its parent (indicated by an arrow on the horizontal link) or that the node weakens the relationship between the second node and its parent (indicated by a vertical bar across the horizontal link). For example, in figure 2.2 the support for node 1 provided by node 2 is weakened by node 3.



1. The BNZ made an offer to Mr Henderson on May 16.
2. The BNZ sent a letter to Mr Henderson on May 16, listing three options for repayment.
3. Letter specifying repayment options, as presented to the court.
4. The letter was merely an invitation to treat.
5. An invitation to treat is not an offer.
6. The option in question—the third option—was merely an invitation to treat.
7. The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.
8. The third option did not specify a definite repayment sum, but said that it would "consider" a reasonable figure.
9. The bank manager indicated that an opportunity would be given to increase the repayment sum.
10. Testimony of Mr Henderson.
11. The third option required someone other than Mr Henderson to take some action.
12. The third option required Mrs Henderson to ". . . make an offer . . ."

Figure 2.3: Representing the subargument relating to offer using the Wigmore notation.

Figure 2.3 shows the portion of the argument for *Henderson v BNZ* dealing with offer displayed using the Wigmore notation. The claim that an offer was made by the bank to Mr Henderson (node 1) is supported by the fact that the bank sent the repayment options letter to him (node 2), which is supported by the physical evidence provided by the letter itself (node 3). The claim is rebutted, however, by the bank's claim that the letter was merely an invitation to treat (node 4), or more specifically, that the third option was merely an invitation to treat (node 6). This is supported by the fact that the bank was only saying that they would consider any offer made by Mrs Henderson in response to the third offer (node 7), which is supported by the wording of the third option (node 8). The inference from node 7 to node 6 is strengthened by the statement by the bank manager saying that the bank may give Mrs Henderson an opportunity to increase the repayment sum if her initial proposal was unacceptable (node 9), which Mr Henderson recalled being said (node 10). The claim that the third option was merely an invitation to treat is also supported by the fact that it required an action by a third party—Mrs Henderson was required to "... make an offer ..." (nodes 11 and 12). The rebuttal itself depends on the general rule that something that is an invitation to treat is not an offer (node 5).

The Wigmore notation is in effect a form of semantic network. However, as the original notation was developed in 1913, before the advent of modern symbolic logic, it has a number of deficiencies, which are still present in the updated notation. The most fundamental deficiency is that no distinction is made between the ways in which propositions can combine to support another proposition. A proposition may be jointly supported or rebutted by each of several propositions (disjunction), or it may be supported or rebutted by a combination of all of the propositions (conjunction). For example, the claim that the BNZ made an offer to Mr Henderson (node 1) is rebutted by the claim that the repayment options letter was an invitation to treat (node 4) and the general rule that invitations to treat are not offers (node 5) *in combination*—if either were proved false, the rebuttal would fail. There is no way to encode the distinction between conjunctive and disjunctive inferences using the existing Wigmore notation. Further points of confusion relate to strengthening and weakening links. Such links do not affect the node to which they are attached, instead they affect the inference from the node to its parent, as represented by the link from the node to its parent. This suggests that strengthening and weakening links should be directly attached to the links that they affect rather than indirectly via the node. (Unfortunately, due to the limitations of the argument layout algorithm in our system, we have been forced to adopt the Wigmorian approach—strengthening and weakening links are connected to nodes rather than links.) There also appear to be two (unstated) classes of strengthening and weakening links: links supporting or rebutting the inference (as opposed to a fact); and links affecting the graded strength of the inference. Perhaps because of these limitations, the original Wigmore notation went down "like a lead balloon" [Robertson 90]. However, there has been a recent resurgence of interest in the Wigmore notation among academic lawyers [Anderson 91].

2.3.3 Toulmin's Notation

An alternative means of representing the structure of arguments is that proposed by Toulmin [Toulmin 58, Toulmin 79]. Toulmin identifies six different elements that can be found in any argument:

- **claims**—assertions put forward publicly for general acceptance.
- **grounds**—the facts upon which a claim is made.
- **warrants**—the means by which a claim can be inferred from its grounds.
- **backing**—generalisations making explicit the body of experience that establishes the trustworthiness of a warrant.
- **modal qualifiers**—qualifications on the degree of reliance to be placed on the conclusions, given the arguments available to support them.
- **rebuttals**—exceptional conditions that, if present, undermine the authority of the warrant for the inference from the grounds to the claim.

The relationship between the elements is as follows: "Given grounds, G, we may appeal to warrant, W (which rests on backing, B), to justify the claim C—or, at any rate, the presumption (M) that C—in the absence of some specific rebuttal or disqualification (R)" [Toulmin 79]. Figure 2.4 displays the portion of the argument dealing with the rebuttal of the claimed offer using Toulmin's notation.

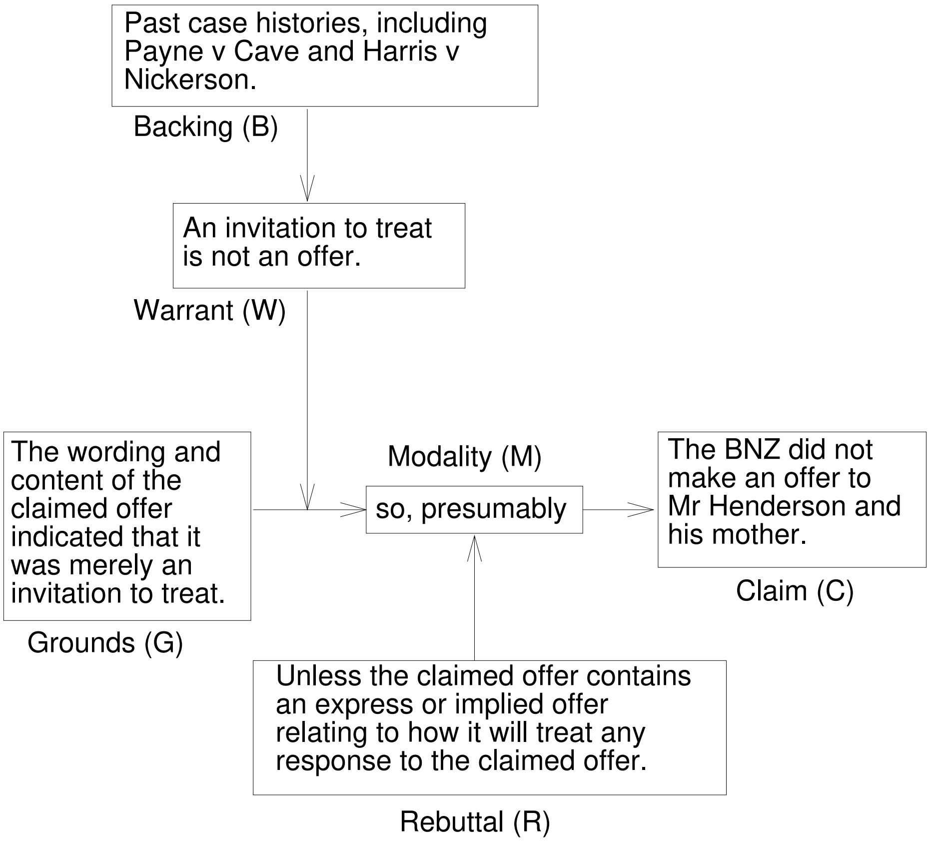


Figure 2.4: Representing the subargument relating to offer using Toulmin's notation.

A comparison between Toulmin's representation and the Wigmore notation reveals that they are similar in some respects. The grounds represent propositions that are connected to the claim via supporting links. However, Toulmin's model represents the warrants and backing for inferences explicitly—something that is lacking in the Wigmore notation. Furthermore, Toulmin's use of modal qualifiers and rebuttals differs from that of Wigmore. Toulmin describes three examples of modal qualifiers:

- "G, so *certainly* C"—the warrant is relevant and has solid backing, and the grounds are all present and certain. No conditions are placed on the claim. This is equivalent to support relationships in the Wigmore notation that are not themselves affected by any strengthening or weakening relationships.
- "G, so *probably* C"—either the warrant is not 100% conclusive or there are doubts as to the existence of some of the grounds. The conclusion is therefore qualified. Uncertain warrants can be represented in the Wigmore notation by weakening relationships that affect the support provided by the generalisation node that represents the warrant. Doubts as to the existence of some of the grounds can be represented by attaching rebutting links to the nodes representing the grounds, or by leaving those grounds

unsupported. However, the Wigmore notation provides no means to express the graded strength of the warrant or of the doubts relating to the grounds.

- "G, so *presumably* C"—the warrant only applies so long as none of the rebuttals hold. Toulmin's rebuttals differ from Wigmorian rebutting links in that they represent exceptions to the rule contained in the warrant. There does not appear to be any means of directly rebutting a claim in the Toulmin model, nor any way to represent exceptions to default inference rules in the Wigmore model.

Toulmin's representation works well for simple arguments. However, as the arguments become more complex, the facts that make up the grounds for the claim may themselves be separate claims that must also be inferred from other facts on the basis of warrants. This makes the argument hard to chart unless the node representing grounds is separated into a number of nodes with each node representing a single fact, in a similar fashion to the Wigmore notation. Clark [Clark 91] has modified the Toulmin notation in this manner in his system for assessing the likelihood of finding oil at a given location. Furthermore, Clark's argument model clearly represents the logic of the support for each claim: warrants are represented as inference rules relating conjunctions of grounds to claims, and disjunctive support for a claim may be represented via multiple warrants supporting the claim. However, Clark's argument model does not represent rebuttals explicitly: instead, they are represented by making the negation of the rebutting fact a ground for the affected warrant.

2.3.4 Birnbaum's Representation

Birnbaum [Birnbaum 80] has developed a theory of argument representation for use in argument understanding. Arguments are represented in a similar manner to the Wigmore notation: a tree of nodes linked via *relations*, with the ultimate probandum at the top of the tree. There are two types of relations: *support* relations and *attack* relations. Support relations are essentially similar to supporting links in the Wigmore notation, but are augmented by *interpretation rules*—generalisations that are similar to Toulmin's warrants. For example, the *Responsibility* interpretation rule states "If an actor performs some action (or refrains from performing some action) which causes (prevents) some state of affairs, then that actor is responsible for causing (blocking) that state of affairs, unless its action (failure to act) was justified." Attack relations are a more limited form of rebutting link, in that they can only be used to link propositions that contradict each other in a fairly straightforward manner. This means that attack relations are symmetric. More complex rebuttals can be represented as a combination of support and attack links. For example, figure 2.5 illustrates how a portion of the argument relating to whether or not the BNZ made an offer to Mr Henderson could be represented using Birnbaum's notation. Note the use of the interpretation rule relating to invitations to treat. Once again, Birnbaum's representation makes no distinction between the ways in which propositions can combine to support another proposition.

Perhaps the most important contribution of Birnbaum's proposal is the notion of *jeopardy*. A proposition is in jeopardy if it has been challenged by the opponent (either directly or indirectly) and the challenge has not yet been satisfactorily addressed. Jeopardy provides each participant in the argument with a means of determining how to best proceed in the argument—a portion of one's own argument that is in jeopardy should be removed from jeopardy by providing further support for it or by attacking the portions of the opponent's argument that jeopardise it. Birnbaum's definition of jeopardy is dynamic, in that it determines how a *new* utterance made by either party affects the rest of the argument. The new

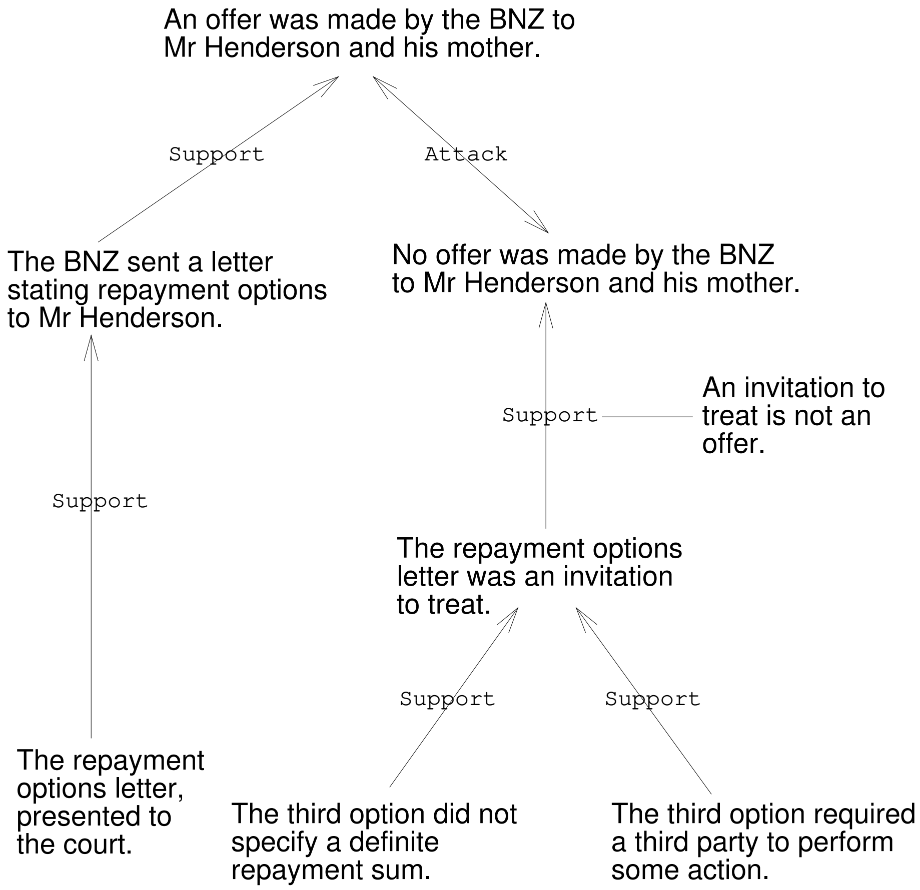


Figure 2.5: Representing the subargument relating to offer using Birnbaum's notation.

jeopardy status of the argument thus depends upon both the new utterance and the current status of the argument:

1. A new attack on a proposition places it in jeopardy.
2. A new support of a jeopardised proposition removes it from jeopardy.
3. If a proposition becomes jeopardised, any proposition that it supports also becomes jeopardised.
4. If a proposition is removed from jeopardy, any proposition that it supports is also removed from jeopardy (as long as no other basis of the support relation is in jeopardy).
5. If a proposition becomes jeopardised, any proposition that it attacks is removed from jeopardy.
6. If a proposition is removed from jeopardy, any proposition that it attacks becomes jeopardised.

The dynamic nature of jeopardy propagation means that the order in which utterances are made is important. For example, if two utterances are made in succession, one supporting a proposition and the other attacking it, then the jeopardy status of the proposition will differ according to which utterance was made more recently.

Dung has developed a theory of argument acceptability that, like jeopardy, may be used to evaluate arguments [Dung 93]. Dung defines an *argumentation framework* to be a set of *arguments*, and a set of *attack relations* between arguments of the form (X, Y) , indicating that argument X attacks argument Y. For example, an argumentation framework concerning whether or not there was an offer in *Henderson v BNZ* might consist of three arguments:

B_1. No offer was made by the BNZ to the Hendersons.

H_1. The third option in the repayment options letter was an offer made by the BNZ to the Hendersons.

B_2. The third option was merely an invitation to treat, not an offer.

and a set of attack relations $\{(B_1, H_1), (H_1, B_1), (B_2, H_1)\}$. An argument is *acceptable* with respect to a set of arguments if all arguments that attack it are themselves attacked by arguments in the set. For example, B_1 is acceptable with respect to the set $\{B_1, B_2\}$: although B_1 is attacked by H_1, H_1 is itself attacked by B_2. Dung provides a means of calculating sets of well-founded arguments in a bottom-up fashion, via the *characteristic function* and the *grounded extension* of an argumentation framework. The characteristic function of an argumentation framework AF, $F_{AF} = \{A \mid A\, is\, acceptable\, wrt\, S\}$. That is, F_{AF} is a mapping from a set of arguments in AF to a second set of arguments in AF that the first set makes acceptable. The grounded extension of an argumentation framework AF, GE_{AF}, is the least fixed point of F_{AF}. It turns out that the least fixed point can be computed by starting with the empty set and repeatedly applying the characteristic function to the set until further applications of the characteristic function do not yield a larger set. For example, the grounded extension for the argumentation framework defined above can be calculated as follows:

$$F_{AF}(\phi) = \{B_2\}. \, F^2_{AF}(\phi) = \{B_1, B_2\} = F^3_{AF}(\phi), \text{ so } GE_{AF} = \{B_1, B_2\}.$$

The grounded extension is a *sceptical* interpretation of the argument framework: it only contains those arguments that are not attacked or that are only attacked by arguments that themselves are attacked (either directly or indirectly) by arguments in the set that are not attacked. A *credulous* interpretation of an argumentation framework is provided by *preferred extensions*, which differ from grounded extensions in that members of a preferred extension need only be acceptable with respect to the preferred extension—a preferred extension need not be "grounded" by arguments that are not attacked. Dung argues that all non-monotonic logics are in fact forms of argumentation, by showing how his theory of argumentation can account for Reiter's default logic, Pollock's inductive defeasible logic, and logic programming.

The semantics for jeopardy differ from the semantics for membership of grounded extensions, due in part to the different models of argument representation used by Birnbaum and Dung: Dung's argumentation frameworks cannot represent explicit support for an argument. Furthermore, the means by which grounded extensions and jeopardised nodes are calculated differs: the derivation of grounded extensions is a static process, whereas the calculation of jeopardy is dynamic, and the order in which utterances are made is important. Chapter 3 describes a semantics for evaluating arguments in terms of the level of support for a fact, that builds on some of the same underlying concepts as grounded extensions and jeopardy.

2.3.5 Other Models of Argument Representation

Statutor is a tutoring system designed by Tom Routen [Routen 92] to help students learn how to apply statute-based law to a given situation. The student is presented with a set of facts and a legal conclusion, and is asked to construct an argument that derives the conclusion from the given facts. Arguments consist of a number of inferences, and are constructed graphically by the user by shifting propositions around the screen and linking them via arrows to form a tree structure. Statutor evaluates the user's argument by simultaneously constructing a proof tree for the argument and matching this with a proof tree extracted from the system's knowledge base. Statutor can also require the user to cite warrants for the constructed inferences in the form of references to sections in the appropriate statute. Routen notes that a natural extension is to store the full text of the statutes on line so that the system can refer the user to sections requiring further study.

The main focus of Statutor is learning by students. The system evaluates the logic of a student's argument and provides feedback. Routen acknowledges that the argument representation used by the system is rather limited. Arguments cannot contain disjunctions or rebuttals. Warrants are simply tags attached to inferences and cannot be used as the basis for further argument. No semantic distinction is made between the types of propositions used in the argument. Furthermore, Statutor can only be used for arguing about statutory law—no support is provided for arguing about case law.

An interesting alternative to purely graphical notations for argument representation is the EUCLID hypertext system proposed by Smolensky et al [Smolensky 87]. EUCLID presents arguments in the form of hypertext: propositions in the argument are represented on the screen by textual descriptions. The relationships between propositions may be represented explicitly via labelled arrows between the appropriate pieces of text, or implicitly by the spatial positioning of the text. For example, the top level display of the argument starts with the main claims of both parties, below which are displayed the related supporting arguments and refutations. Refutations are horizontally aligned with the arguments that they refute. More detail can be provided by expanding one of the subarguments to reveal its supporting arguments and refutations. To assist the user in constructing arguments and to facilitate easier browsing, EUCLID enables the user to display and match the full text of the argument against the components of the argument displayed on-screen.

EUCLID's primary function is the use of arguments to construct and browse documents in a structured manner. The final outputs of the argument construction process are a structured argument *and* a document containing the full text of the argument. Although two of the goals of the system presented in this dissertation (argument construction and browsing) are similar to EUCLID's goals, our system has a different focus. Our system uses a fine-grained argument representation that is tailored to the legal domain. Furthermore, we place a considerable emphasis on argument evaluation: facilities are provided to ensure that the argument considers all the relevant issues, that the assumptions underlying the argument are made explicit, and that any weaknesses in the argument are highlighted. An additional goal of our system is document retrieval. Documents stored and retrieved by the system are static, and are used primarily to justify points made in the argument. However, a facility enabling the user to simultaneously construct a document containing the full text of the argument, as the argument is developed, would be a useful extension to the system.

2.3.6 Towards a Better Argument Representation

Each of the representations described above has its own strengths and weaknesses. What is needed is a model that incorporates the strengths of the existing models while eliminating their weaknesses. The model should explicitly represent the type of each fact as does the Wigmore notation. The model should also explicitly represent the relationships between facts via links such as those used in the Wigmorian and Toulmin notations, and adequately distinguish between conjunctive and disjunctive support for facts. The model should distinguish between direct rebuttals of facts as does the Wigmore notation and exceptions to defeasible inference rules as does Toulmin's model. The model should incorporate some notion of warrants or general reasons why facts support each other. The model should have a well-defined notion of the relationship between facts in which one fact strengthens or weakens the support for another. A means of evaluating the argument such as acceptability or jeopardy should be incorporated into the semantics of the model. Finally, the core vocabulary of the model should remain small and simple, while allowing the possibility for extension by users. Chapter 3 describes such a model.

Chapter 3

The Argument Representation

This chapter introduces a new model of argument representation that eliminates a number of the weaknesses of the existing representations. The representation consists of two parts—the *conceptual representation* and the *graphical representation.* The conceptual representation is a canonical encoding of the structure of a legal argument. This is mapped to one of a number of graphical representations that contains the argument as displayed to the user. The advantage of layering the representation in this manner is that the graphical representation can be changed without having to change the underlying conceptual representation, improving the flexibility of the model. Section 3.1 describes the syntax and semantics of the conceptual representation. Section 3.2 describes how arguments can be represented graphically.

3.1 The Conceptual Representation

The syntax and semantics of a propositional form of the conceptual representation are defined in sections 3.1.1 and 3.1.2. An extension that allows variables in propositions is outlined in section 3.1.3.

3.1.1 Syntax of Argument Structures

Definition

An *argument structure* A is an ordered pair (F, I), where F is a set of *facts* and I is a set of *argument inferences.*

There are five types of facts: *evidence* facts, *secondary* facts, *uncontroversial* facts, *warrant* facts, and *unmarked* facts.

An *argument inference*[1] is an expression $(\{S_i\}, \{R_j\}) \succ f$ relating a fact f to its *supporting arguments* $S_1, S_2 \ldots S_n$ and *rebuttals* $R_1, R_2 \ldots R_m$. Each supporting argument and rebuttal is a conjunction of facts $f_1 \wedge f_2 \wedge \cdots \wedge f_n$.

Evidence Facts

Evidence facts represent the physical presentation of evidence in court: for example, the act of giving testimony by a witness or expert, or a document presented to the court. Evidence

[1] Throughout this dissertation, the term "inference" is used to refer to relationships between facts.

facts cannot be rebutted.

Secondary Facts

Secondary facts represent assumptions that are assumed to be true by default. That is, secondary facts should be believed in the absence of any evidence to the contrary—the burden of proof for the secondary fact lies upon the party seeking to discredit the secondary fact. As such, secondary facts have a form of inherent, "built-in" support. Sartor [Sartor 91] describes how secondary facts can be mapped to a number of languages for non-monotonic reasoning, including Reiter's default logic and Poole's THEORIST language.

As secondary facts are assumptions, they may be rebutted by the opponent. Secondary facts can thus represent *opportunities* to be exploited or *threats* to be avoided. For example, a secondary fact that is critical to the plaintiff's argument represents both an opportunity for the defendant and a threat to the plaintiff's argument—if the secondary fact is rebutted, then the plaintiff's argument may be severely weakened. Thus, in preparing for a case, it is wise not to rely upon secondary facts to provide support without either strengthening the fact with additional supporting arguments or preparing counter-arguments for any rebuttals that the opposing counsel is likely to raise.

In practise, the rules governing admissibility of evidence require lawyers to declare all the evidence that they intend to use before the trial begins, during the process of *discovery*. However, if the opposing counsel introduces evidence late in the period of discovery, the lawyer may not have time to find evidence that can be used to rebut the new evidence introduced by the opposition. It follows that lawyers should always seek to find evidence providing support for secondary facts, in case the opposing counsel introduces evidence late in the discovery process that rebuts the assumed fact. Opportunities and threats only arise when a lawyer makes a mistake, by failing to introduce evidence supporting a secondary fact before the trial begins. However, consideration of opportunities and threats can be very useful during case preparation, in situations where the lawyer has not had time to collect all of the evidence relating to a claim, but he or she is reasonably sure that enough evidence to support the claim can be found.

Secondary facts are often derived from exceptions to well-understood rules of law [Sartor 91]. For example, the rule derived from case law for communication of acceptance via post is as follows.

> Acceptance is communicated by the offeree at the time at which the letter of acceptance is posted, unless one of the following conditions holds:
>
> 1. The offeror specified a mode of communication other than via post.
> 2. The offeror and offeree intended for there to be no agreement until the offeror received the letter of acceptance.
> 3. The letter of acceptance was not properly stamped or addressed.

In arguing that legal requirements for communication of acceptance were satisfied at a certain time, counsel must argue that the letter of acceptance was posted at that time. The burden of proof for the exceptions falls upon the opposing counsel, which suggests that the exceptions can be represented as secondary facts by changing them into positive statements. For example, the statement "The letter of acceptance was properly stamped and addressed" is a secondary fact that, in combination with the fact that the letter was posted, supports the claim that legal requirements for communication of acceptance were satisfied at the time at which the letter was posted.

Uncontroversial Facts

Uncontroversial facts are those that are uncontested by either side. Such facts may be *judicially noticed* as being common knowledge, meaning that no evidence is required to support the fact. For example, in *Henderson v BNZ*, the judge treated as uncontroversial the proposition that any commitment by Mr Henderson to continue making the weekly interest payments was small in relation to his obligation to pay the entire debt.

Warrant Facts

Warrant facts denote generalisations or rules that provide support for inferring one fact from a set of other facts. An example of a warrant is the rule that if an offer is made and accepted, valid consideration exchanged and both parties intended for the agreement to have legal consequences, then a contract is formed between the offeror and the offeree. It is usually wise to provide some form of backing for warrants, as it is possible that the validity of the warrant may be challenged. Note that the rule-like nature of warrants is not explicit in the representation: as the representation focuses on the structure of the argument as opposed to its content, only the notion that the fact is to be thought of as a rule is encoded explicitly.

Unmarked Facts

Unmarked facts represent claims other than evidence facts, secondary facts, uncontroversial facts, and warrant facts.

Example Syntax

The conceptual representation of the subargument related to offer in *Henderson v BNZ* is as follows:
$A = (\{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_{11}, f_{12}\}, I)$, where:

f_1 = The BNZ made an offer to Mr Henderson.

f_2 = The BNZ sent a letter to Mr Henderson, listing three options for repayment.

f_3 = Letter specifying repayment options, as presented to the court.

f_4 = The letter was merely an invitation to treat.

f_5 = An invitation to treat is not an offer.

f_6 = The option in question—the third option—was merely an invitation to treat.

f_7 = The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.

f_8 = The third option did not specify a definite repayment sum, but said that it would "consider" a reasonable figure.

f_{11} = The third option required someone other than the offeree (Mr Henderson) to take some action.

f_{12} = The third option required Mrs Henderson to "... make an offer ..."

$$
\begin{aligned}
\text{and } I = \{\ & (\{f_2\},\{f_4 \wedge f_5\}) \succ f_1, \\
& (\{f_3\},\phi) \succ f_2 \\
& (\{f_6\},\phi) \succ f_4, \\
& (\{f_7, f_{11}\},\phi) \succ f_6, \\
& (\{f_8\},\phi) \succ f_7, \\
& (\{f_3\},\phi) \succ f_8, \\
& (\{f_{12}\},\phi) \succ f_{11}, \\
& (\{f_3\},\phi) \succ f_{12}\}
\end{aligned}
$$

The example shows that constructing and examining arguments using the canonical conceptual representation can be confusing. It is simpler for the user to examine a graphical representation of the argument, such as that displayed in figure 3.4 on page 28, below.

3.1.2 Semantics of Argument Structures

The semantics of the conceptual representation is defined in terms of the extent to which a fact occurring in a given argument structure is *supported* by related facts. Facts are either *supported*, *unsupported*, or *rebutted*, as described below.

Definition

A fact is **supported** if either of the following conditions holds:

- It is an evidence fact, uncontroversial fact, warrant fact, or secondary fact.
- At least one of its supporting arguments is supported, and all of its rebuttals are either unsupported or rebutted.

A supported fact is one that has a valid supporting argument that is based upon sound evidence and whose rebuttals are not based on sound evidence or have been discredited. For example, in *Henderson v BNZ*, the proposition stating that the third repayment option required Mrs Henderson to make an offer is supported due to the physical evidence provided by the content of the repayment options letter (which has built-in support) and the absence of any rebuttals.

Definition

A fact is **unsupported** if it is an unmarked fact, it is not rebutted, and either of the following conditions holds:

- It has no supporting arguments.
- All of its supporting arguments are either unsupported or rebutted.

Unsupported facts are those that are derived from a chain of argument that remains unrefuted, but for which some underlying facts have not been established. The argument is thus not yet complete—it must be extended by providing concrete support for the unproven facts, usually in the form of evidence. For example, one cannot simply state, "The wording of the third repayment option indicates that it was merely an invitation to treat," as evidence must be provided to back up the claim, such as the physical evidence provided by the repayment options letter.

Definition

A fact is **rebutted** if at least one of its rebuttals is supported.

Rebutted facts are those that are directly rebutted by facts that are themselves based upon valid supporting arguments and sound evidence. They indicate serious flaws in the argument that must be dealt with by either rebutting the argument upon which the original rebuttals are based, or by providing a different supporting argument to which the rebuttal does not apply. For example, in *Henderson v BNZ*, Henderson may base his argument upon consideration being provided by the $200 weekly repayments that he was making to the bank. If this argument is rebutted, for example by the bank's claim that the weekly repayments are not sufficient consideration because they relate to his previous debt, Henderson would have to either rebut the bank's claim or argue that consideration was provided in a different manner.

Definition

A supporting argument or rebuttal is

- **Supported** if all of the facts in the supporting argument or rebuttal are supported.
- **Unsupported** if it is not rebutted and at least one fact in the supporting argument or rebuttal is unsupported.
- **Rebutted** if at least one fact in the supporting argument or rebuttal is rebutted.

As supporting arguments and rebuttals are conjunctions of facts, a change in the level of support for one fact in the conjunction can change the level of support for the supporting argument or rebuttal as a whole. For example, a supporting argument for the claim that a contract exists is that there was an offer, the offer was accepted, valid consideration was provided for the offer, and both parties intended for the agreement to have legal consequences. Assuming that the other facts in the supporting argument are supported, if the claim that valid consideration was provided for the offer was rebutted, then the overall level of support for the supporting argument would change from supported to rebutted.

The semantics of support for an argument represent a sceptical interpretation of the argument. A fact is supported if it has a supporting argument that is supported *and* no rebuttals that are supported. A credulous interpretation of the argument would ignore rebuttals when evaluating the support for a fact.

Another means of presenting the semantics of the conceptual representation is as the following calculus. The calculus has the advantage of providing a formal algorithm for determining the support for a given fact. The calculus consists of three operators, which may be used to compute the level of support for a fact from the level of support for its supporting and rebutting arguments. Table 3.1 describes the three operators. S denotes supported, U denotes unsupported, and R denotes rebutted.

The first operator, $\times$, can be used to derive the level of support $\mathcal{S}(X)$ for a supporting argument or rebuttal $X = f_1 \wedge \ldots \wedge f_n$, by calculating the product of the levels of support for each conjunct f_i in the supporting argument or rebuttal:

$$\mathcal{S}(X) = \prod_i \mathcal{S}(f_i)$$

$\times$	S	U	R
S	S	U	R
U	U	U	R
R	R	R	R

$+$	S	U	R
S	S	S	S
U	S	U	U
R	S	U	R

$\star$	S	U	R
S	R	S	S
U	R	U	U
R	R	U	U

Table 3.1: The support calculus.

The product formed represents the level of support for the least well-supported conjunct in the supporting argument or rebuttal. The $\times$ operator thus acts to *minimise* the level of support provided by its arguments. The $\times$ operator is both commutative and associative, so the order in which the conjuncts are combined to form the product does not matter.

The second operator, $+$, can be used to derive the support for a set of supporting arguments or a set of rebuttals $\{X_i\}$, by calculating the sum of the levels of support for each supporting argument or the sum of the levels of support for each rebuttal:

$$\mathcal{S}(\{X_i\}) = \sum_i \mathcal{S}(X_i)$$

The sum formed represents the level of support for the most well-supported supporting argument or rebuttal. The $+$ operator thus acts to *maximise* the level of support provided by its arguments. Once again, the operator is both commutative and associative, so the order in which the supporting arguments or rebuttals are combined is immaterial. If the set of supporting arguments or set of rebuttals is empty, then there is no support for the supporting arguments or rebuttals:

$$\mathcal{S}(\phi) = U$$

The third operator, $\star$, can be used to derive the total level of support for a fact f from its supporting arguments $\{S_i\}$, rebuttals $\{R_k\}$ and its inherent, built-in support $\mathcal{B}(f)$:

$$\mathcal{S}(f) = (\mathcal{B}(f) + \mathcal{S}(\{S_i\})) \star \mathcal{S}(\{R_k\})$$

This can be expanded to become

$$\mathcal{S}(f) = (\mathcal{B}(f) + \sum_i \prod_j \mathcal{S}(f_{ij})) \star \sum_k \prod_l \mathcal{S}(f_{kl})$$

where $S_i = \bigwedge_j f_{ij}$ and $R_k = \bigwedge_l f_{kl}$.

The built-in support for a fact f, $\mathcal{B}(f)$, depends upon the type of the fact:

$$\mathcal{B}(f) = \begin{cases} S & \text{if } f \text{ is an evidence fact, a secondary fact, a warrant fact or an} \\ & \text{uncontroversial fact} \\ U & \text{if } f \text{ is an unmarked fact} \end{cases}$$

It is easy to show that this calculus evaluates the level of support for a fact in the same manner as the definitions presented earlier in this section.

The support calculus provides a static means of evaluating an argument, similar to Dung's notion of acceptability described in section 2.3.4. However, the support calculus is richer than Dung's notation in that it is able to deal with explicit support for propositions, conjunctive and disjunctive supporting arguments and rebuttals, and different types of propositions.

Example Semantics

The semantics for the subargument related to offer in *Henderson v BNZ* is as follows:
$\mathcal{S}(f_3) = (\mathcal{B}(f_3) + \mathcal{S}(\{\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_8) = (\mathcal{B}(f_8) + \mathcal{S}(\{f_3\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_{12}) = (\mathcal{B}(f_{12}) + \mathcal{S}(\{f_3\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_2) = (\mathcal{B}(f_2) + \mathcal{S}(\{f_3\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_7) = (\mathcal{B}(f_7) + \mathcal{S}(\{f_8\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_{11}) = (\mathcal{B}(f_{11}) + \mathcal{S}(\{f_{12}\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_6) = (\mathcal{B}(f_6) + \mathcal{S}(\{f_7, f_{11}\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_4) = (\mathcal{B}(f_4) + \mathcal{S}(\{f_6\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_5) = (\mathcal{B}(f_5) + \mathcal{S}(\{\})) \star \mathcal{S}(\{\}) = \mathbf{S}$.
$\mathcal{S}(f_1) = (\mathcal{B}(f_1) + \mathcal{S}(\{f_2\})) \star \mathcal{S}(\{f_4 \wedge f_5\}) = \mathbf{R}$.

The example shows that manually calculating the level of support for a fact is a tedious exercise, and would be impractical for large, complicated arguments. This suggests that a system for representing arguments should have a means of calculating the support for a fact automatically.

3.1.3 Argument Packets

An *argument packet* is a self-contained argument structure stored in a generalised form. Argument packets differ from argument structures in that they explicitly encode general patterns of inference. In particular, propositions in an argument packet can contain variables, and argument packets do not contain evidence facts. The inferences contained in argument packets often have backing in the form of links to the precedent cases upon which they are based. Argument packets can be instantiated and added to a particular argument structure by copying the relevant portions of the argument packet into the argument structure and providing values for the unbound variables that have been added to the argument. Appendix A describes a number of argument packets that have been developed for the domain of contract law.

3.2 The Graphical Representation

The graphical representation is layered above the underlying conceptual representation, and is completely independent of the conceptual representation. The link between the two layers is provided by *graphical styles* that map the information contained in the conceptual representation to the graphical representation. Defining a new graphical representation is simply a matter of defining a new graphical style to map the underlying conceptual representation to the new graphical representation. Defining new graphical styles is discussed in more detail in chapter 7.

The goal of graphical styles is to provide a mechanism that is flexible enough to allow the user to swap between accurate graphical representations of the Wigmore notation (both the original and updated versions), the Toulmin notation, Birnbaum's notation, and so forth. This requires a means of mapping the canonical conceptual representation to

- Notations that are less expressive than the conceptual representation. For example, both the Wigmore and Toulmin notations do not distinguish between conjunctive and disjunctive supporting arguments and rebuttals.
- Notations that represent facts and inferences in a different manner to the conceptual representation. For example, secondary facts in the conceptual representation would have to be mapped to Toulmin's rebuttals as exceptions to inference rules, and vice-versa. There must be some means of transforming the conceptual syntax in such a manner.
- Notations that use different methods to lay out the argument. For example, arguments in the Wigmore notation are presented hierarchically with the leaves at the bottom and the conclusion at the top, whereas the Toulmin notation presents arguments left to right.

The graphical styles currently provided by the system fall some way short of achieving this flexibility. At present, there is no means of transforming the underlying syntax contained in the conceptual representation to the syntax of a particular notation. Each graphical style must be able to represent every type of fact defined in the conceptual representation. Moreover, the basic method of laying out arguments is the same for each graphical style (see below). However, the system does provide a means of changing the way in which different types of facts and inferences are displayed graphically.

The argument structure is represented graphically in the form of a tree, with facts represented by nodes and inferences by sets of links between the nodes. Leaf nodes represent the grounds for a particular chain of argument, such as evidence facts, warrant facts, unrebutted secondary facts, or uncontroversial facts. Each level of the tree derives additional facts based on the facts from the previous level, culminating in the ultimate probandum at the root of the tree. Each node is assigned a type that describes what the node represents in general terms. Links between nodes are also assigned a type that describes the relationship between the connected nodes. The base types supported by the graphical representation are as follows:

1. *Unmarked Node*—an unmarked fact.
2. *Evidence Node*—an evidence fact.
3. *Secondary Node*—a secondary fact.
4. *Uncontroversial Node*—an uncontroversial fact.
5. *Warrant Node*—a warrant fact.
6. *Supporting Link*—indicates that the lower node supports its parent node.
7. *Rebutting Link*—indicates that the lower node rebuts its parent node.
8. *Strengthening Link*—indicates that a node strengthens the force of the support or rebuttal that a node confers on its parent.
9. *Weakening Link*—indicates that a node weakens the force of the support or rebuttal that a node confers on its parent.

Figure 3.1: Conjunctive and disjunctive supporting arguments and rebuttals.

Figure 3.1 illustrates how conjunctive supporting arguments are distinguished from each other in the graphical representation. Supporting arguments that consist of a conjunction of two or more nodes are connected to the parent node by a single link that branches out to all nodes in the conjunctive supporting argument. An arc across the branching links reinforces the fact that the nodes are combined conjunctively. Conjunctive and disjunctive rebuttals are distinguished in exactly the same manner. Nodes can participate in more than one supporting argument or rebuttal for the same parent node by having multiple links from the node to its parent. For example, in the first argument structure in figure 3.1, node 1 is supported by both node 2 and node 3 in combination, and also by node 3 and node 4 in combination. In the second argument structure, nodes 2, 3 and 4 combine to rebut node 1. Note that in situations where a node has a single supporting argument and no rebuttals, or a single rebuttal and no supporting arguments (as in the second argument structure), the line from the branching links to the node is not required.

Nodes and links are displayed using one of a number of graphical styles. Each graphical style contains a list of node and link types, and their corresponding graphical representation under that style. Graphical styles allow the user to switch graphical representations without changing the underlying conceptual representation. Figure 3.2 displays a graphical style based upon a version of the Wigmore notation that has been extended to include secondary nodes. The first argument structure in figure 3.4 uses this style to display the argument relating to whether or not there was an offer in *Henderson v BNZ*. An example of a graphical style loosely based upon the Toulmin notation is displayed in figure 3.3. An exact representation of the Toulmin notation has not been possible because of the necessity (under the current implementation of graphical styles) to represent all base node and link types. Furthermore, limitations of the argument layout algorithm used by the system prevent the argument from being displayed horizontally, as per the Toulmin notation. The most significant difference between the Wigmore and Toulmin styles relates to the placement of warrant nodes. The Wigmorian style places warrants within the supporting argument or rebuttal to which they relate. However, in the Toulmin style, warrant nodes are attached horizontally to the branching point for the related supporting argument or rebuttal, as shown in the second argument structure in figure 3.4. The difference in placement is encoded by the explicit warrant link types defined for the Wigmore and Toulmin styles, as shown in figures 3.2 and 3.3. Examples presented throughout the rest of this dissertation use the Wigmorian style, unless noted otherwise.

3.2.1 Strengthening and Weakening Links

It is easy to see how the first seven base types of nodes defined above can be mapped to the conceptual representation. However, no mention was made in the conceptual representation of strengthening and weakening links. An analysis of these links has revealed that

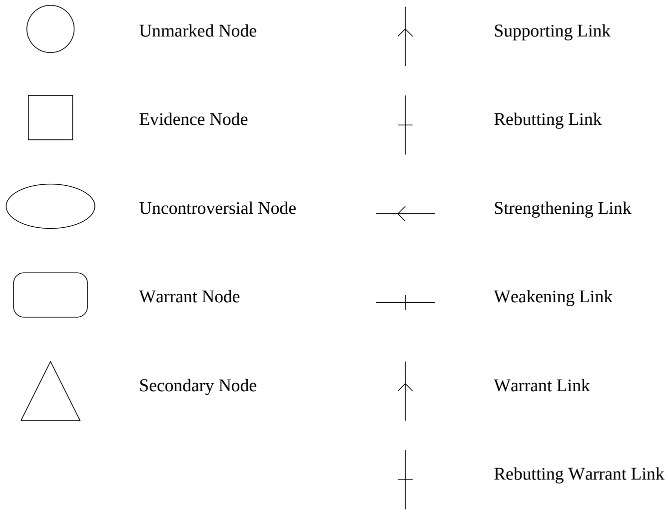


Figure 3.2: Node and link types in the graphical style based upon the Wigmore notation.

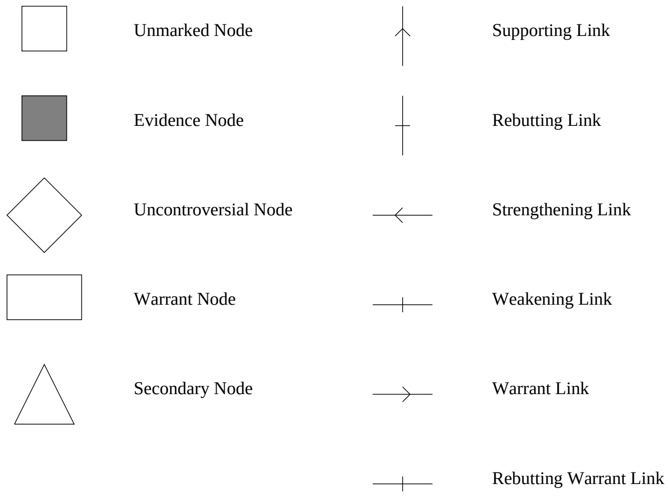
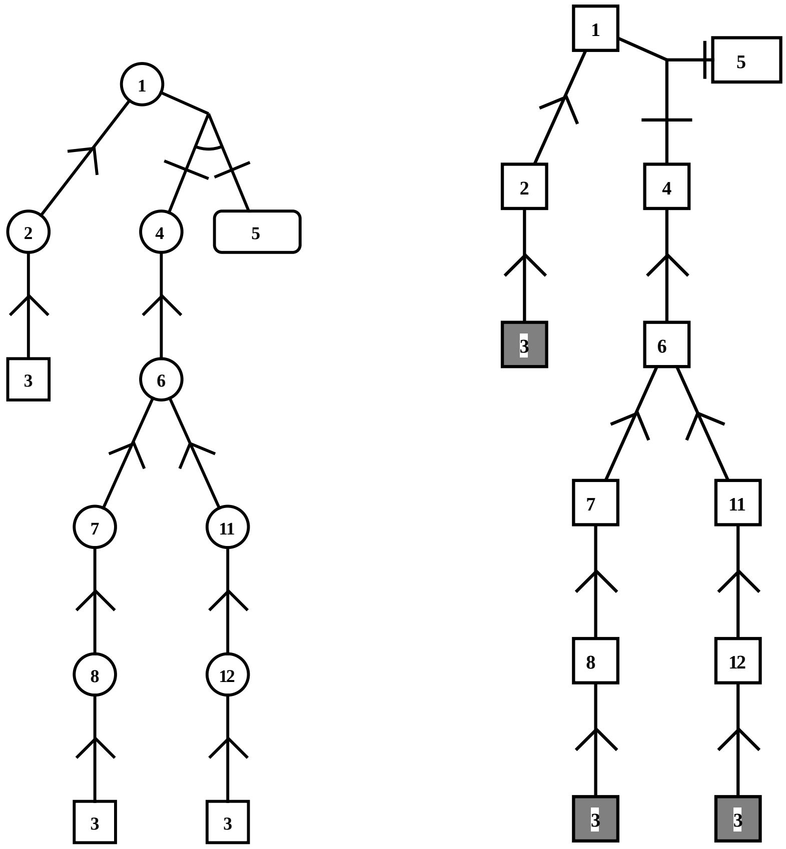


Figure 3.3: Node and link types in the graphical style based upon the Toulmin notation.

all such links appear to be based upon an implicit assumption. By making explicit the implicit assumption upon which the link is based, the link can be mapped into the conceptual representation. Strengthening links are links that support the implicit assumption, while weakening links rebut the validity of the assumption. Figure 3.5 displays how structures containing strengthening or weakening links are mapped to structures that do not contain such links, using the updated Wigmore notation. In each case, a new node (labelled "A") is introduced to represent the assumption underlying the relationship affected by the strengthening or weakening link.

For example, in *Henderson v BNZ* the claim that the third option was an invitation to treat is supported by the claim that the BNZ was only saying that it would consider a reasonable repayment sum proposed by Mrs Henderson. This supporting relationship is strengthened by the statement by the bank manager to the effect that an opportunity to increase the repayment sum would be given if the bank rejected Mrs Henderson's initial proposal (see the first subargument in figure 3.6). This is because there is an unstated



1. The BNZ made an offer to Mr Henderson on May 16.
2. The BNZ sent a letter to Mr Henderson on May 16, listing three options for repayment.
3. Letter specifying repayment options, as presented to the court.
4. The letter was merely an invitation to treat.
5. An invitation to treat is not an offer.
6. The option in question—the third option—was merely an invitation to treat.
7. The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.
8. The third option did not specify a definite repayment sum, but said that it would "consider" a reasonable figure.
11. The third option required someone other than Mr Henderson to take some action.
12. The third option required Mrs Henderson to "... make an offer ..."

Figure 3.4: A graphical representation of the subargument relating to offer in *Henderson v BNZ.*

assumption in support of the claim that the third option was an invitation to treat: the bank was inviting Mrs Henderson to commence negotiations rather than inviting her to accept an offer. This underlying fact is supported by the bank manager's statement as shown in the second subargument in figure 3.6.

Examples of weakening links can be found in the argument relating to the case *R v Thomas* [NZLR 72], a famous New Zealand criminal law case in which a farmer was accused and convicted of the murder of a neighbouring farmer and his wife, Harvey and Jeanette Crewe. Much of the case rested on forensic evidence used to match Thomas's rifle with the bullets found in the bodies of the deceased. The Crown also argued that Thomas's rifle had been fired in the Crewes' garden, on the basis that a cartridge case fired from Thomas's rifle was found in the garden. However, the argument is weakened by the accusation that the

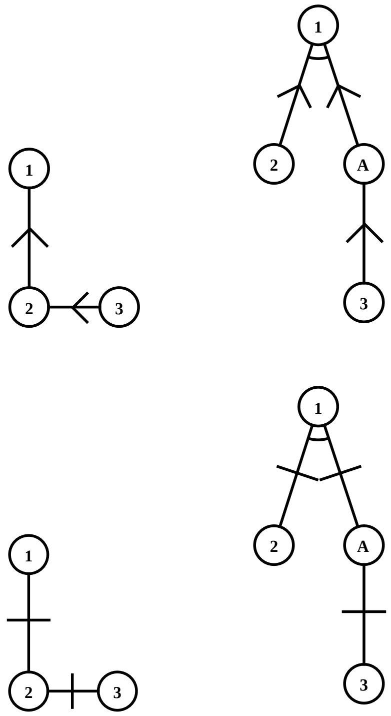


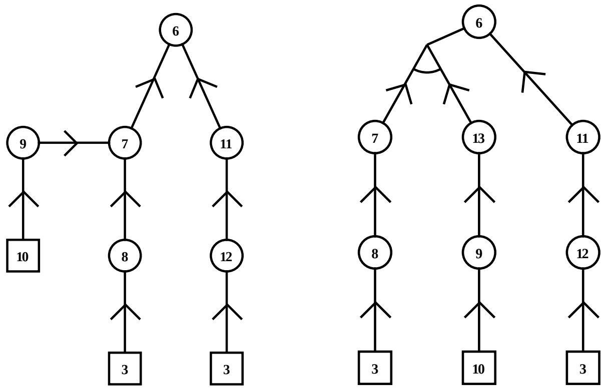
Figure 3.5: Eliminating strengthening and weakening links.

police had planted the cartridge case.[2] In actual fact, the argument depends on the unstated assumption that the cartridge case was not moved into the garden after it had been fired, a proposition that is rebutted by the proposition that the police planted the cartridge case. Figure 3.7 illustrates the argument graphically, both with and without the use of a weakening link.

Figure 3.8 illustrates a more complex example from *R v Thomas*. The police forensically examined rifles belonging to potential suspects to determine whether the rifles could have fired the bullets that were recovered from the bodies of Jeanette and Harvey Crewe. They could not positively identify a rifle that definitely fired the bullets, but they were able to rule out all rifles apart from the one belonging to Thomas. The police inferred from the results of the sample that the bullets were fired from Thomas's rifle. This inference is weakened by the claim that the sample was too small—had the sample been bigger, more rifles may have been found that also could not have been eliminated. However, the police responded to the assertion that the sample was too small by arguing that the sample had been taken from all potential suspects, so examining any other rifles would be superfluous. The mapping from the original argument to one that does not contain any weakening links is a two-step process, with each step eliminating one of the weakening links. The assumption behind the first inference is that the sample was sufficiently large to return a meaningful result (node 5). The further assumption underpinning the rebuttal to the first assumption is that the sample was not only small, but that it was also not well chosen (node 6). Figure 3.8 shows how the assumptions underlying the argument are revealed by removing the weakening links one by one.

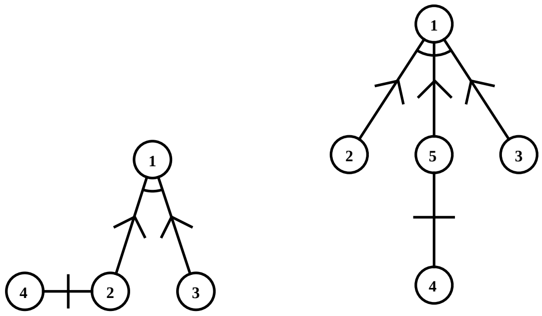
Chapter 6 describes a mechanism for automatically mapping structures involving strengthening and weakening links to structures that make the underlying assumptions explicit and

[2]The cartridge case was found in plain sight during the third search of the garden, three months after the murder.



3. Letter specifying repayment options, as presented to the court.
6. The option in question—the third option—was merely an invitation to treat.
7. The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.
8. The third option did not specify a definite repayment sum, but said that it would "consider" a reasonable figure.
9. The bank manager indicated that an opportunity would be given to increase the repayment sum.
10. Testimony of Mr Henderson.
11. The third option required someone other than Mr Henderson to take some action.
12. The third option required Mrs Henderson to "... make an offer ..."
13. The bank was inviting Mrs Henderson to commence negotiations.

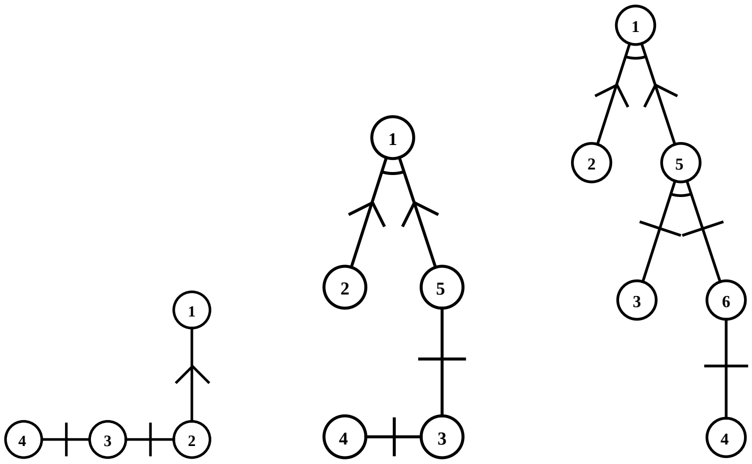
Figure 3.6: Unpacking the strengthening link in *Henderson v BNZ*.



1. Thomas's rifle was fired in the Crewes' garden.
2. A cartridge case was found in the garden.
3. The cartridge case had been fired from Thomas's rifle.
4. The cartridge case was planted by the police.
5. The cartridge case was not moved into the garden after it had been fired.

Figure 3.7: The planted cartridge case in the Arthur Alan Thomas case.

that contain only supporting and rebutting links.



1. Both fatal bullets were fired from Thomas's rifle.
2. Thomas's rifle was the only rifle examined that was not eventually excluded.
3. The sample of rifles was too small.
4. The sample was not taken at random—it was taken from people for whom there existed a prior probability of guilt.
5. The sample was sufficiently large to return a meaningful result.
6. The sample was not well chosen.

Figure 3.8: Unpacking the argument relating to the sample of rifles taken by the police.

3.3 Summary of Argument Representation

This chapter has presented a representation that incorporates the strengths of the existing representations while eliminating a number of weaknesses. Significant features of the representation include

- Separation of the representation into a canonical conceptual representation and a graphical representation to enable the user to easily swap between different graphical representations.
- A conceptual representation with a clearly defined syntax and semantics that represents the contribution of each fact and inference to the argument as a whole. The conceptual representation incorporates
  - An explicit representation of different *types* of facts, including warrants.
  - A distinction between direct rebuttals of facts and exceptions to inferences, using *secondary* facts.
  - A well-defined notion of strengthening and weakening relationships.
- A calculus for evaluating the level of support for a given fact in the argument.

Chapter 4 describes a system that allows users to construct and edit arguments using the representation.

Chapter 4

System Overview

Even seemingly straightforward legal cases can require fairly elaborate preparation by lawyers, and require large quantities of evidence to be organised. This complexity often makes it impractical to carry out the tasks of case preparation and evidence management by hand. This chapter describes a system that simplifies these tasks by enabling users to construct and evaluate arguments, and to use arguments as a basis for indexing and retrieving documents associated with the case at hand. The system employs the representation developed in chapter 3.

The primary functions of the system are argument construction, argument evaluation, and document retrieval. The features provided to support this functionality are described below.

Argument Construction

An argument editor is provided that enables the user to construct, edit, and browse arguments via a graphical user interface. Argument construction is also facilitated as follows.

- The construction of complex arguments is simplified by the provision of a set of *argument packets* encoding standard patterns of inference, which may be retrieved and copied into arguments.
- Browsing is facilitated by a set of *argument views* that enable the user to focus on specific portions of the argument, by collapsing the portions of the argument that are not of current interest.
- A set of *graphical styles* is provided that allows the user to display the argument in different ways. The user can also create new styles by editing the existing ones.

Argument Evaluation

A further set of argument views is provided that enables the user to evaluate the level of support for a given fact in the argument, and to highlight areas of weakness in the argument.

Document Retrieval

The system provides a high-level interface to a hypertext database of documents relating to a case. The system enables the user to index the document database by associating

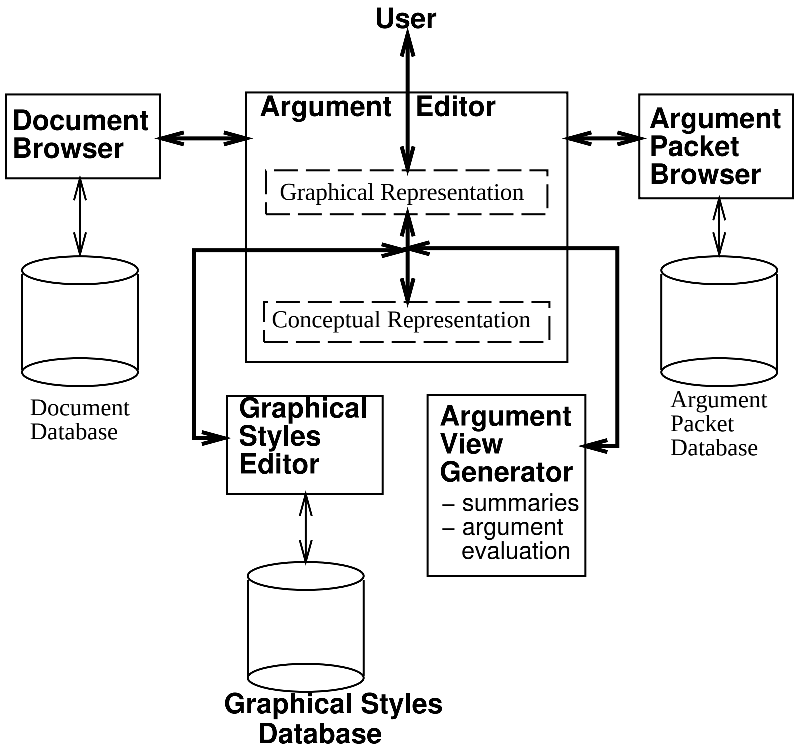


Figure 4.1: Major components of the system.

documents with facts and inferences in the argument structure. Documents associated with physical evidence such as contracts, photographic images, and so forth, can be attached to the evidence nodes in the argument structure for the case at hand that represent the content of the documents. In contrast, documents containing the decisions of past precedents can be retrieved from argument packets and copied into the argument structure.

Section 4.1 presents an overview of the system architecture, while section 4.2 describes how the system may be used to store and retrieve documents.

4.1 System Architecture

The overall system architecture is shown in figure 4.1.

There are five main components of the system:

- The *argument structure editor* is the core component of the system that provides a graphical user interface for editing arguments and allows the user to invoke the *argument packet browser*, the *document browser*, the *argument view generator*, and the *graphical styles editor*. The argument structure editor is described in more detail in section 4.1.1.
- The *argument packet browser* allows the user to traverse the hierarchy of argument packets and copy selected argument packets into the argument structure. The argument packet browser and the use of argument packets is described in more detail in chapter 5.

- The *document browser* allows the user to retrieve and display documents attached to the argument structure or argument packets. The document browser is described in more detail in section 4.2.

- The *argument view generator* allows the user to activate one of a number of *argument views* or mechanisms for summarising and evaluating arguments. Argument views are discussed in more detail in chapter 6.

- The *graphical styles editor* enables the user to alter the way in which the argument structure is presented graphically. The interface to the graphical styles editor is described in chapter 7.

4.1.1 The Argument Structure Editor

The argument structure editor consists of two windows: the main editing window, and the argument description window. The main editing window displays the argument structure in a graphical form. The argument description window contains the content of the argument, in the form of a list of the text descriptions associated with the nodes, ordered by node number. The main editing window and the argument description window are displayed in figure 4.2.

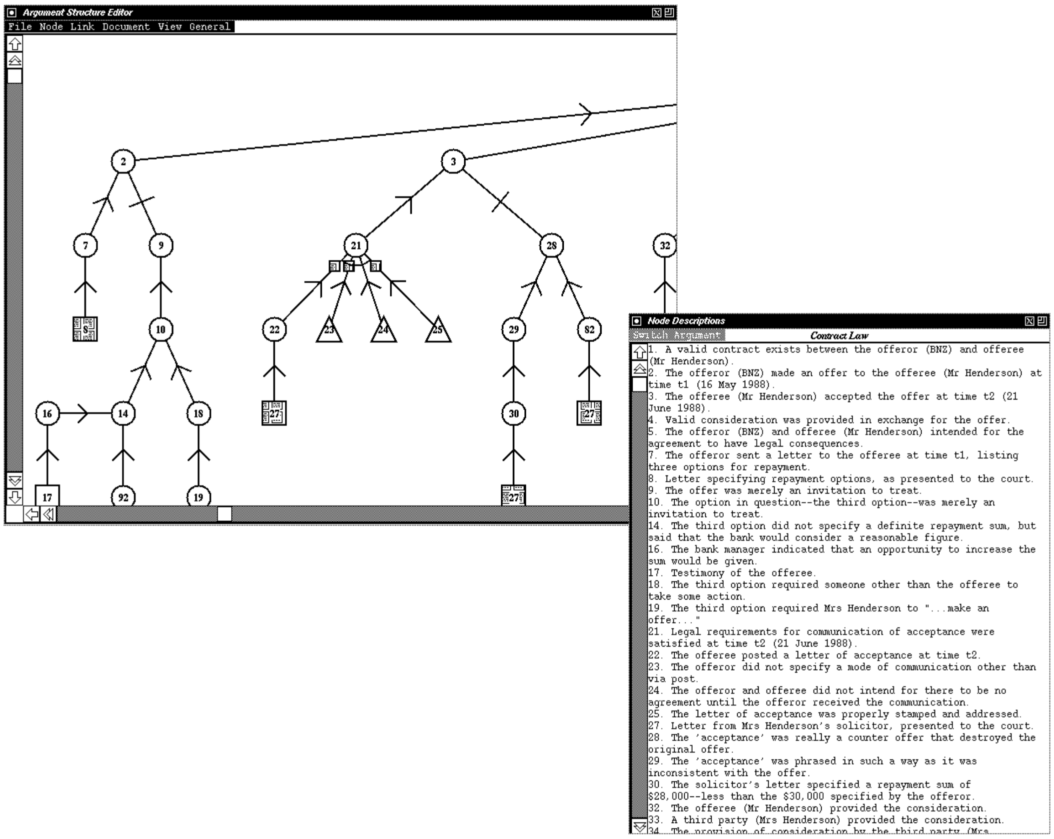


Figure 4.2: The argument structure editor.

The main editing window contains a pull-down menu that allows the user to invoke a number of different options. The "File" menu contains options allowing the user to load, save, or print argument structures. The "Node" menu allows the user to add new nodes to the argument structure, edit existing nodes, or remove nodes from the argument structure. The "Link" menu provides the same options for links. The "Document" menu allows the user to attach documents to nodes or links in the argument structure, as described in section 4.2. The "View" menu allows the user to invoke argument views for collapsing and unfolding portions of the argument, and for evaluating arguments. These are described in more detail in chapter 6. Finally, the "General" menu provides a number of miscellaneous options such as invoking the argument packet browser or graphical styles editor, or reducing the size of the graphical representation so that the entire argument fits in the main editing window.

4.2 Document Indexing

Under the conceptual retrieval approach to document retrieval, documents are organised according to the legal concepts that are relevant to their content. Argument structures provide a means of describing the legal issues that are present in a particular case. This suggests that argument structures can be used as a means of indexing the documents associated with a case, by linking documents to the legal issues to which they relate. The argument representation is layered on top of the underlying document database to act as the external interface to the documents.

4.2.1 Types of Documents

In general there are two types of document stored in the document database: documents containing transcripts of court decisions in previous cases, and documents relating to evidence presented in court, such as copies of contracts and transcripts of the testimony of witnesses. The type of a document, along with its content, determines the point in the argument structure to which it is attached. Documents relating to past precedents are attached to the links in the argument structure representing the inferences that they justify. Note that documents relating to past precedents are initially attached only to links in argument packets, but may be added to the argument structure when an argument packet is instantiated, as described below. Documents relating to evidence presented in court are attached to the appropriate evidence nodes in the argument structure.

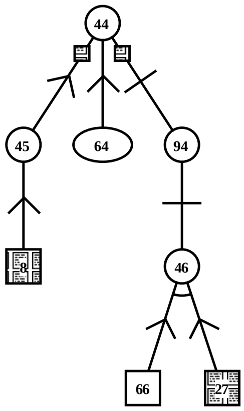
4.2.2 Attaching Documents to Argument Structures

Documents can be attached to both nodes and links. Selecting either option results in a dialogue box being displayed asking the user to enter the document type (case transcript or evidence), the name of the document, the name of the file containing the document text, and the case verdict if the document is a transcript of a previous case. After entering the necessary information, the user then clicks on the node or link to which the document is to be attached. A document associated with a node is indicated by changing the pattern of the node, whereas a document associated with a link is indicated by adding a "hot button" to the link (see figure 4.3).

Another means of adding a document to the argument structure is provided by the argument packet interface described in chapter 5. Any documents attached to nodes or links in an argument packet are inherited by instances of the argument packet that are added to the argument structure. The documents provide implicit backing for the warrants contained in the argument packet instance, in the form of the past precedents used to construct the argument packet. An alternative means of representing the backing for these warrants would

Figure 4.3: Indicating the presence of documents attached to nodes and links.

be to explicitly attach "backing" nodes to the warrants via supporting links. Each backing node would have a relevant precedent case attached to it. An (unimplemented) view that allows the user to switch between implicit and explicit representation of backing is described in chapter 6.



8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, as presented to the court.
44. The consideration was insufficient because Mr Henderson was already legally bound to provide it due to an existing contract with the BNZ.
45. The consideration was the payment of interest on a debt owed by Mr Henderson to the BNZ.
46. Mr Henderson was already making the interest payments and he made no fresh commitment to continue paying.
64. Mr Henderson's obligation to pay the interest payments was a very small matter in relation to his obligation to pay the entire debt.
66. Testimony of the bank manager.
94. The consideration was sufficient because it resulted in the BNZ obtaining an additional benefit.

Figure 4.4: Examples of document links in *Henderson v BNZ*.

Figure 4.4 shows an example of documents attached to a portion of the argument structure for *Henderson v BNZ*. Nodes 8 and 27 have attached documents that were used as evidence to support facts in the subargument: the repayment options letter, and the letter from Mrs Henderson's solicitor, respectively. In addition, two references have been made to precedents as backing for inferences in the subargument. The link from node 45 (the fact that the potential item of consideration being discussed was the payment of interest by Mr Henderson to the BNZ) to node 44 (the claim that the payments were insufficient consideration) is justified by the ruling made in *Foakes v Beer* that a part payment of an existing debt could not be used as consideration for the waiving of obligation to pay the rest of the debt. The link from node 94 (the claim that the BNZ obtained an additional benefit from the payment of interest by Mr Henderson) that rebuts node 44 is justified by the ruling made in *Williams v Roffey Bros.* that if the provision by the offeree of something that the offeror was already contractually entitled to receive results in the offeror obtaining an additional benefit, then sufficient consideration is deemed to have been provided by the offeree.

4.2.3 Retrieving Documents

Documents are retrieved by clicking the right mouse button on the node or link to which the document is attached. If only one document is attached to the node or link, then it is immediately loaded and displayed, otherwise the user is prompted to select from the list of documents attached to the node or link. Documents, once loaded, are kept in memory so that subsequent retrievals do not require the document to be loaded again.

4.2.4 Displaying Documents

Documents are stored in *HTML* format [Berners-Lee 93] and displayed using the *NCSA Mosaic* document viewer [Braverman 93a], as shown in figure 4.5. Navigating between documents can be accomplished in a number of different ways. Some documents contain embedded hypertext links to other documents. Clicking on such links causes the viewer to load and display the linked document. For example, the document in the figure contains a hypertext link to *Byrne v Van Tienhoven*—a case cited in the given document. The user also has access to the full functionality provided by NCSA Mosaic, such as the "Forward" and "Back" buttons, which can be used to move between previously viewed documents.

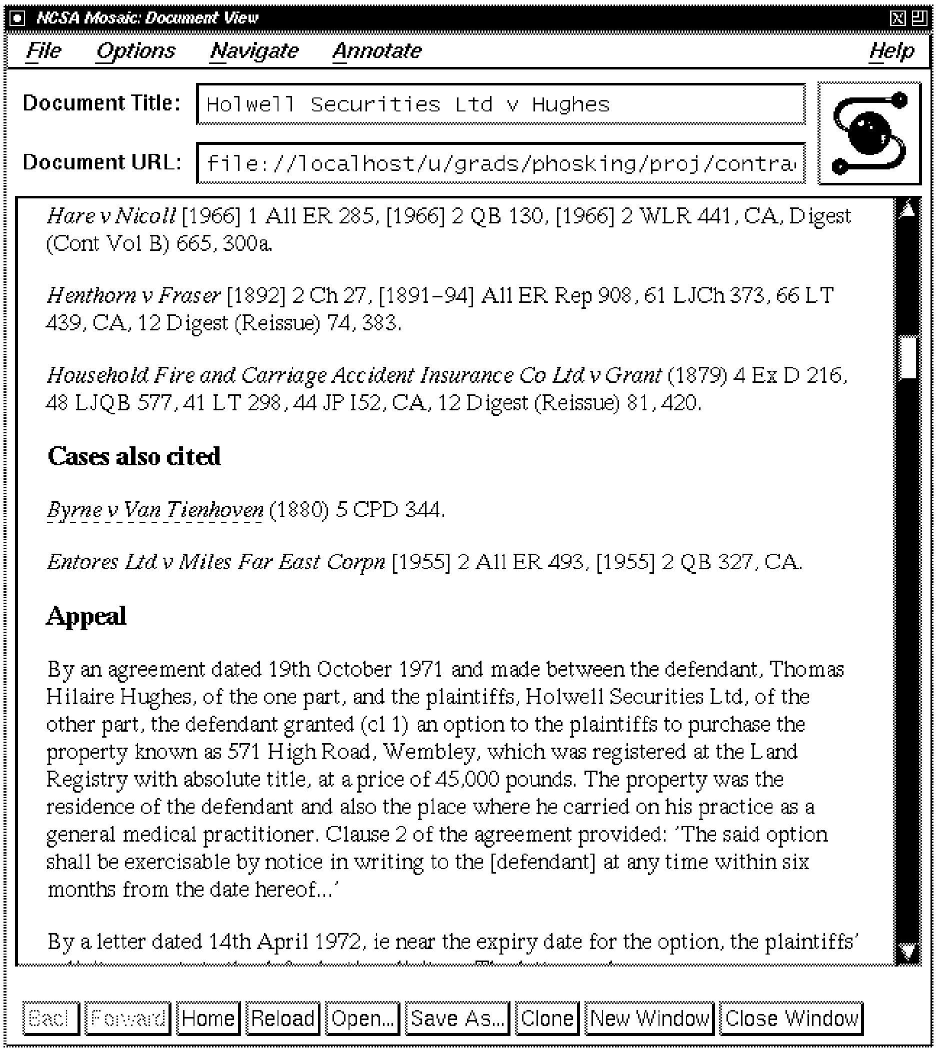


Figure 4.5: Displaying documents using xmosaic.

Chapter 5

Constructing Arguments

When constructing a new argument, the user is initially asked to choose the legal domain with which the argument is to be associated, whereupon the root node defined for the chosen domain is added to the argument structure. The user can then expand the argument using either of two different methods: by adding nodes and links to the argument one by one, or by using argument packets to add entire groups of nodes to the argument structure at once. These methods are described in section 5.1 and section 5.2.

5.1 Constructing Arguments A Node At A Time

The user can manually add new nodes and links to the argument structure by choosing options from the "Node" and "Link" menus. For example, selecting the "Add Node" option results in a window being displayed that allows the user to enter the type of the new node, the type of the new link that will attach the new node to an existing node in the argument structure, and a textual description of the fact represented by the node. The user must also indicate the manner in which the link should be connected to the new node's parent. Choosing "and" means that the node will become a new conjunct of an existing supporting argument or an existing rebuttal for its parent. Choosing "or" means that the node will create a new supporting argument or a new rebuttal for its parent. Separate menu options allow the user to edit or delete existing nodes and links in the argument structure.

The textual description for each node is input in the form of a string template that may contain references to global *template variables*. These references are replaced by the current value of the variable when the label is displayed to the user. For example, figure 5.1 shows a node whose description contains references to the variables "offeror", "offeree" and "t1", represented using brackets enclosing the variable names. The textual descriptions are displayed in the argument description window as described in section 4.1.1. A "Zoom" mode can also be activated that allows the user to display the textual description of a node in the Editing window simply by moving the mouse pointer over the node. Figure 5.2 shows the display of the description for node 44 in zoom mode.

5.2 Constructing Arguments Using Argument Packets

Entering complex arguments one node at a time can become tedious and time-consuming. Fortunately, most arguments relating to a particular domain of law are based upon standard

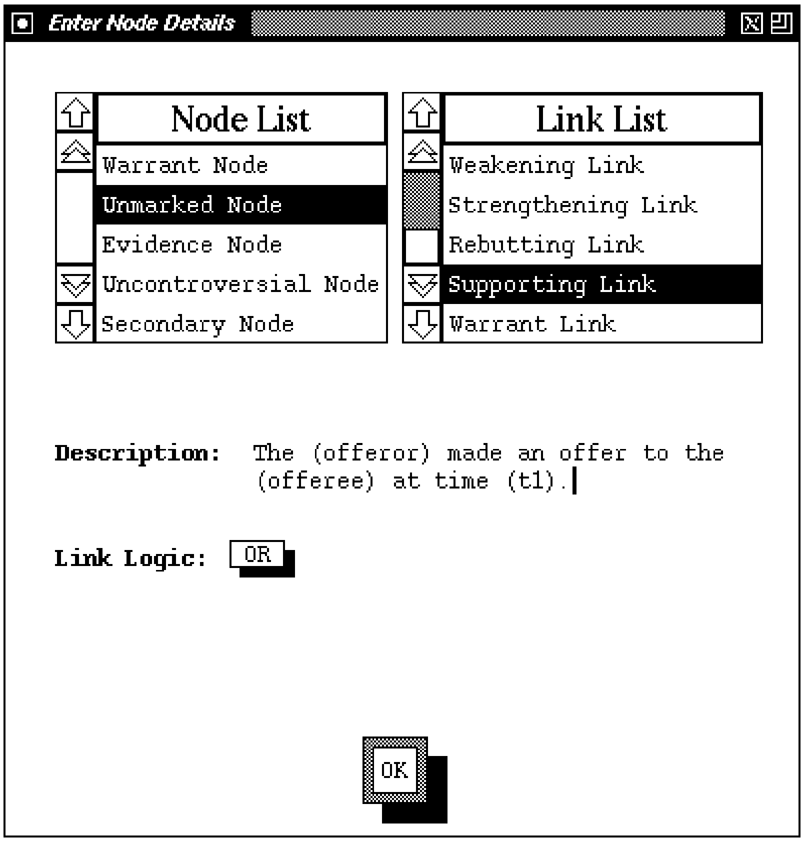


Figure 5.1: Adding a node to the argument structure.

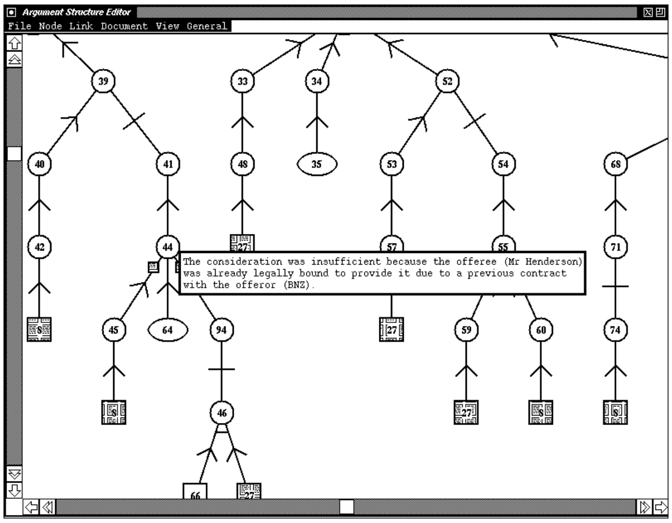
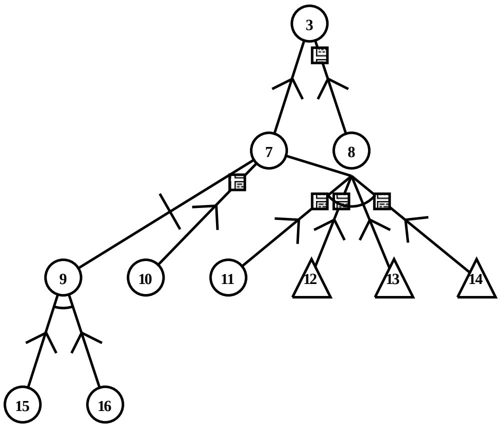


Figure 5.2: Displaying node descriptions in zoom mode.

patterns of inference (rules of law) developed from a distinct set of past precedents. By representing and storing these patterns so that they can be later retrieved and copied into an argument structure, the ease of constructing arguments can be greatly increased.

An *argument packet* is a self-contained, generalised argument structure encoding a particular rule of law. Argument packets are stored in a generalised form, and contain no specific details of any particular case. In particular, argument packets do not contain evidence nodes, and all template variables contained in argument packets are uninstantiated. Figure 5.3 displays the argument packet for communication of acceptance in the domain of Contract Law. Links in the argument packet contain "hot buttons" to documents relating to the precedent cases upon which the inferences in the argument packet are based. For example, the link between node 8 and node 3 in figure 5.3 is labelled with a box representing the past precedent of *Carlill v Carbolic Smoke Ball Co*, whose judgment held that if the person making an offer expressly states or implies that communication of acceptance is not necessary, then performance of the other conditions in the offer is sufficient acceptance without notification. The box also acts as a hypertext button for retrieving the text of the case, as described in section 4.2.



3. The offeree accepted the offer at time t2.
7. Legal requirements for communication of acceptance were satisfied at time t2.
8. The offeror expressly or implicitly waived the requirement of communication of acceptance.
9. The offeree used an inappropriate mode of communication.
10. The offeror received the communication at time t2.
11. The offeree posted a letter of acceptance at time t2.
12. The offeror did not specify a mode of communication other than via post.
13. The offeror and offeree did not intend for there to be no agreement until the offeror received the communication.
14. The letter of acceptance was properly stamped and addressed.
15. The offeror specified a particular mode of communication.
16. The offeree used a less expeditious method of communication.

Figure 5.3: The argument packet for "Communication of acceptance".

Argument packets are similar to Gardner's examples in that they represent knowledge abstracted from the decisions made in previous cases. Unlike Gardner's examples, however, they retain the links to the cases upon which they are based, and the warrants underlying the argument packet can be further supported or rebutted by the user. Argument packets in effect provide indices into the case database in a similar fashion to Ashley's dimensions. However, the application of argument packets to an argument structure constructs the argument in fine detail as opposed to the list of case citations produced by HYPO.

More generally, argument packets differ from cases in that they are represented at a different level of abstraction: cases describe specific facts or events, whereas argument packets describe general rules of law that have been extracted from the judgments of a number of cases. However, as described in section 5.2.3, below, argument packets can be adapted to fit new situations in a similar fashion to case adaptation in case-based reasoning (c.f. [Kass 89, Jones 91]).

Argument packets are grouped together in terms of the domain of law to which they are applicable. Within each domain of law, argument packets are stored in a hierarchical fashion with a "root" argument packet at the top of the hierarchy. The argument packets at a particular level in the hierarchy are linked to leaf nodes in argument packets in the previous level of the hierarchy, so that all argument packets can be reached from the root argument packet. This dissertation focuses primarily on the domain of contract law (although examples of argument packets from US Federal tax law also appear below). Figure 5.4 displays the hierarchy of argument packets developed for the domain of contract law. The argument packets themselves are described in full detail in appendix A.

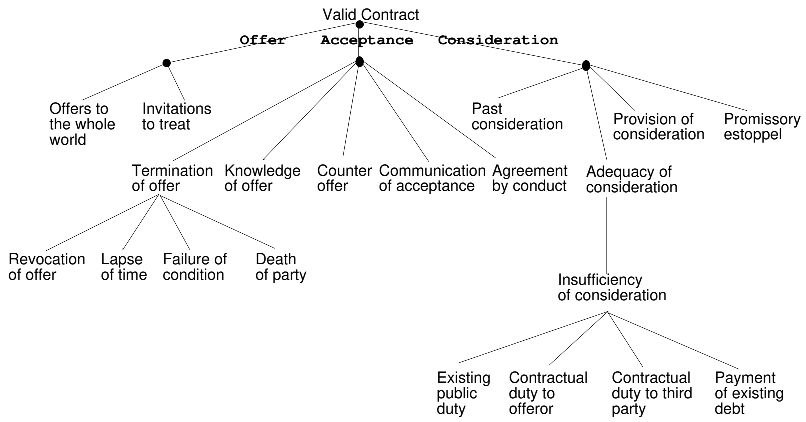


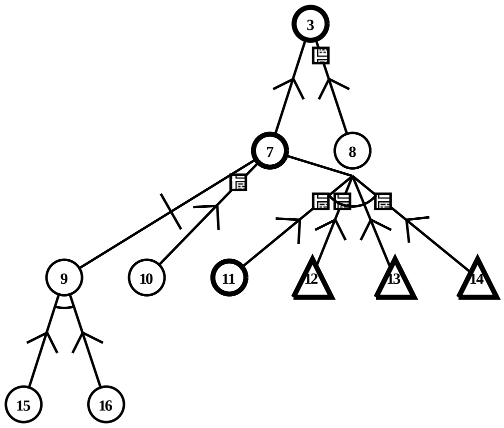
Figure 5.4: The argument packet hierarchy for the Contract Law domain.

5.2.1 Selecting an Argument Packet

The user may access the argument packets via either of two methods. Selecting the "Browse Argument Packets" option in the "General" menu results in the root argument packet for the current domain being displayed in the Argument Packet window. The user can then browse the argument packet hierarchy until the appropriate argument packet is displayed. Alternatively, clicking the middle mouse button on a node in the argument structure displays a menu of argument packets that may be applied to that node. The chosen argument packet is then displayed in the Argument Packet window, with nodes labelled with the node numbers they would be allocated if the argument packet was copied immediately into the argument structure.

5.2.2 Copying an Argument Packet

Once the desired argument packet has been located, the user must then choose the parts of the argument packet that are applicable to the current case. Many argument packets contain disjunctive inferences, only some of which are relevant to the case in hand. For example, in figure 5.5 it can be seen that acceptance is made (node 3) if either legal requirements for communication of acceptance are satisfied (node 7), or the requirement for communication of acceptance was waived (node 8). The user may select portions of the argument packet via the left mouse button—clicking the left mouse button on a given node will highlight that node along with all other nodes in the same supporting argument. The nodes in the supporting argument or rebuttal containing the parent of the selected node are then highlighted, with this process continuing recursively up the tree until the root of the argument packet is reached. For example, clicking on node 12 causes the subargument dealing with acceptance via post (nodes 3, 7, 11, 12, 13, and 14) to be highlighted, as shown in figure 5.5.



3. The offeree accepted the offer at time t2.

7. Legal requirements for communication of acceptance were satisfied at time t2.

8. The offeror expressly or implicitly waived the requirement of communication of acceptance.

9. The offeree used an inappropriate mode of communication.

10. The offeror received the communication at time t2.

11. The offeree posted a letter of acceptance at time t2.

12. The offeror did not specify a mode of communication other than via post.

13. The offeror and offeree did not intend for there to be no agreement until the offeror received the communication.

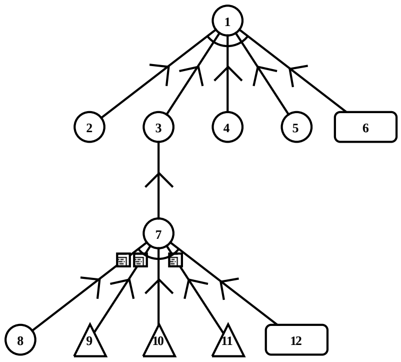
14. The letter of acceptance was properly stamped and addressed.

15. The offeror specified a particular mode of communication.

16. The offeree used a less expeditious method of communication.

Figure 5.5: Filtering the argument packet.

Having selected the relevant portions of the argument packet in this manner, the user can then copy the selected portions of the argument packet into the argument structure. A new instance of the selected subgraph is constructed, whose root is the node in the argument structure from which the user accessed the argument packet browser. Copies of any "hot buttons" to precedent cases attached to links in the selected subgraph are attached to the corresponding links in the new instance, providing implicit backing for the inferences represented by the newly constructed links. If the textual description associated with a new node instance contains references to previously defined variables, the values for the variables are automatically inserted into the templates. References to new variables are resolved by asking the user to provide the appropriate values. Warrant nodes are automatically generated for the inferences represented by conjunctive supporting arguments and rebuttals contained by the new argument packet instance. Warrant nodes are assigned textual descriptions in the form of an "if...then..." rule, where the antecedent of the rule is a conjunction derived from the textual descriptions for the conjuncts involved in the inference, and the conclusion is derived from the textual description for the parent node involved in the inference. For example, the warrant node generated for the supporting argument dealing with communication of acceptance via post (nodes 11—14) is assigned the textual description "If the offeree posted a letter of acceptance at time t2 and the offeror did not specify a mode of communication other than via post and the offeror and offeree did not intend for there to be no agreement until the offeror received the communication and the letter of acceptance was properly stamped and addressed, then legal requirements for communication of acceptance were satisfied at time t2." The warrant nodes have implicit backing in the form of the argument packet itself and the past precedents associated with the inferences.



1. A valid contract exists between the offeror (BNZ) and offeree (Mr Henderson).
2. The offeror (BNZ) made an offer to the offeree (Mr Henderson) at time t1 (16 May 1988).
3. The offeree (Mr Henderson) accepted the offer at time t2 (21 June 1988).
4. Valid consideration was provided in exchange for the offer.
5. The offeror (BNZ) and offeree (Mr Henderson) intended for the agreement to have legal consequences.
6. If the offeror made an offer to the offeree at time t1 and the offeree accepted the offer at time t2 and valid consideration was provided in exchange for the offer and the offeror and offeree intended for the agreement to have legal consequences then a valid contract exists between the offeror and offeree.
7. Legal requirements for communication of acceptance were satisfied at time t2 (21 June 1988).
8. The offeree (Mr Henderson) posted a letter of acceptance at time t2 (21 June 1988).
9. The offeror (BNZ) did not specify a mode of communication other than via post.
10. The offeror (BNZ) and offeree (Mr Henderson) did not intend for there to be no agreement until the offeror received the communication.
11. The letter of acceptance was properly stamped and addressed.
12. If the offeree posted a letter of acceptance at time t2 and the offeror did not specify a mode of communication other than via post and the offeror and offeree did not intend for there to be no agreement until the offeror received the communication and the letter of acceptance was properly stamped and addressed then legal requirements for communication of acceptance were satisfied at time t2.

Figure 5.6: Copying the argument packet into the argument structure.

When the portion of the argument packet for "Communication of Acceptance" selected in figure 5.5 is copied into a partially constructed argument structure for *Henderson v BNZ*, the argument structure that results is as shown in figure 5.6.

5.2.3 Adapting Argument Packets

In some situations, straightforward application of an argument packet is not possible, as the current case does not quite fit the pattern of inference encoded in the argument packet. In such a case, it may be appropriate to *broaden* the scope of the argument packet to fit the circumstances of the current case. In other situations, it may be appropriate to argue that an argument packet does not apply to the current case due to extenuating circumstances, either by *discrediting* the argument packet or *distinguishing* it from the current case. Broadening, distinguishing, and discrediting are each examples of argument strategies defined by Skalak and Rissland for dealing with situations where a strict interpretation of a rule is not desirable [Skalak 92]. They describe a number of *argument moves* that can be used to modify the rule so that it either supports the argument for the current case or does not rebut it. Broadening an argument packet can be accomplished by widening the scope of the warrant for one of its supporting arguments, or by narrowing the scope of one of its rebuttals. Discrediting an argument packet involves attacking one of the warrants upon which the argument packet is based. Finally, distinguishing the current case from the general pattern of inference contained in an argument packet can be achieved by arguing that one of the key facts in a supporting argument for the argument packet is not true of the current case.

For example, consider an argument packet representing a rule of the United States Federal tax law that determines whether or not a taxpayer is entitled to a deduction for expenses associated with maintaining a "home office"—a part of the home used for business purposes. The rule currently states that the taxpayer is entitled to a deduction if a portion of the home is used exclusively on a regular basis

1. As the principal place of business for any trade or business of the taxpayer,

2. As a place of business that is used by patients, clients, or customers in meeting or dealing with the taxpayer in the normal course of his trade or business, or

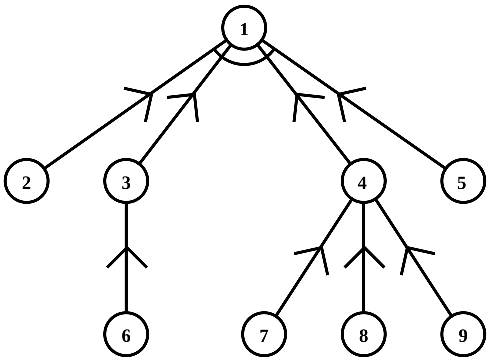
3. In the case of a separate structure that is not attached to the home, in connection with the taxpayer's trade or business.

Furthermore, if the taxpayer is an employee, then the exclusive use must be for the convenience of his or her employer.[1]

Suppose that the courts' current interpretation of "regular use" is at least 50% of the taxpayer's work time (i.e. all the cases in which a deduction was granted involved usage greater than 50%). The argument packet for the rule is displayed in figure 5.7.

Consider a case in which all conditions for the deduction are met except for the condition of "regular use"—the taxpayer in question is only using the home office for 40% of his or her work time. In order to convince the court that the taxpayer is entitled to the deduction, it is necessary to establish that 40% is enough to be considered regular use. By doing so we will in effect be broadening the rule by relaxing one of the conditions for entitlement. One strategy for accomplishing this is to use what Rissland and Skalak refer to as a "double negative" argument: that is, argue that the current case is so unlike the cases in which the deduction was not granted because the "regular use" condition was not satisfied, that the condition for "regular use" should be considered to be satisfied. For example, if all previous cases in which the condition for "regular use" was not established involved less than 20% of the taxpayer's work time, one could argue that 40% is clearly more substantial than 20% and that it would

[1] This rule has been adapted from [Skalak 92].



1. The taxpayer is entitled to the home office deduction.
2. The home office is used exclusively for business purposes.
3. The home office is used on a regular basis.
4. The home office is used for business purposes.
5. The use of the home office is for the convenience of the employer.
6. At least 50% of the taxpayer's work time is spent in the home office.
7. The home office is used as the principal place of business for the taxpayer.
8. The home office is used as a business meeting place by the taxpayer.
9. The home office is a separate structure used in connection with the taxpayer's business.

Figure 5.7: The argument packet for home office deductions.

be unfair to withhold the deduction on that basis. This argument may be countered by a "slippery slope" argument from the IRS, stating that 20% usage is clearly insufficient and that allowing the rule to be broadened from 50% usage to 40% usage makes a precedent that could eventually cause the condition to be reduced all the way to 20%. These arguments are displayed in figure 5.8.

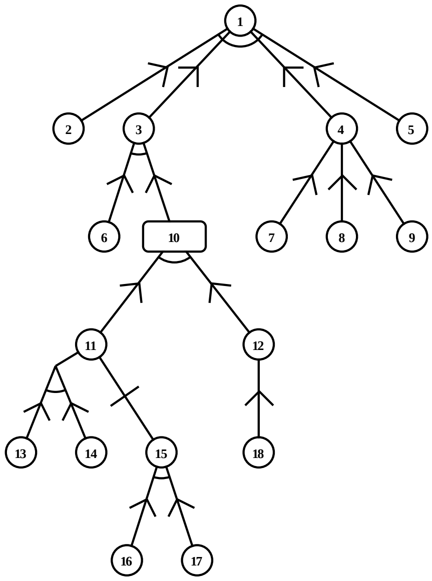
Another strategy for adapting argument packets is to discredit a rule. For example, in a case where the taxpayer has been spending half of his work time in the home office, the IRS might argue that the condition for regular use is already too broad: because of the difficulty in determining the point at which the amount of work time spent elsewhere becomes too much to allow a deduction, the rule should require all of the taxpayer's work time to be spent in the home office. This argument is displayed in figure 5.9.

One useful but as yet unimplemented feature would be to automatically incorporate the results of successful broadening and discrediting arguments into the argument packets themselves. Such a feature would facilitate incremental knowledge acquisition. By incrementally updating argument packets with the results of relevant cases, one could ensure that argument packets do not become obsolete through changes in the law.

5.3 Summary

This chapter has described how arguments may be constructed by adding nodes one by one to the argument or by instantiating *argument packets*—standard patterns of inference for the given domain of law. Argument packets facilitate the quick construction of complex arguments, while also providing a novel means of indexing documents relating to past cases.

A number of strategies have been described for arguing about the applicability of an argument packet to a particular case.



1. The taxpayer is entitled to the home office deduction.

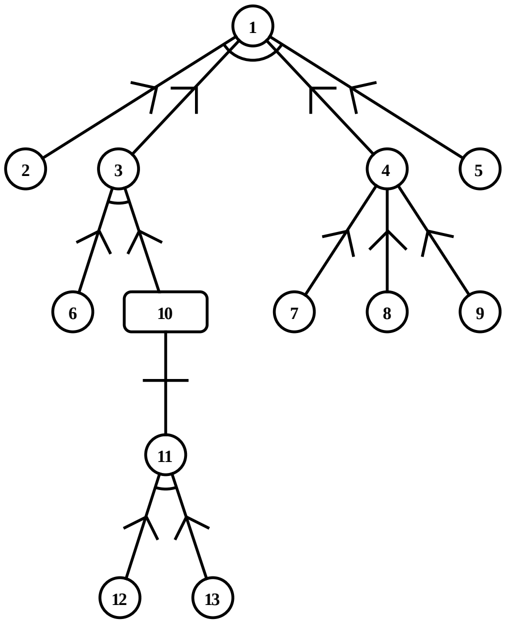
   ⋮

6. 40% of the taxpayer's work time is spent in the home office.

   ⋮

10. 40% is sufficient to be considered "regular use".
11. 40% is a legitimate broadening of 50%.
12. 50% use is sufficient to be considered regular use.
13. The cases where the deduction was not granted due to the condition for regular use not being satisfied all involved use of less than 20% of the taxpayer's work time.
14. 40% use is clearly more substantial than 20% use.
15. Allowing 40% use to be considered regular use would lead to an untenable rule.
16. 20% use is clearly untenable.
17. Subsequent claims by taxpayers would continue to lower the rate required for "regular use" until eventually 20% was reached.
18. 50% use appears in the existing argument packet.

Figure 5.8: Broadening the condition for regular use.



1. The taxpayer is entitled to the home office deduction.
2. The home office is used exclusively for business purposes.
3. The home office is used on a regular basis.
4. The home office is used for business purposes.
5. The use of the home office is for the convenience of the employer.
6. At least 50% of the taxpayer's work time is spent in the home office.
7. The home office is used as the principal place of business for the taxpayer.
8. The home office is used as a business meeting place by the taxpayer.
9. The home office is a separate structure used in connection with the taxpayer's business.
10. 50% is sufficient to be considered "regular use".
11. 50% use is too broad.
12. Allowing 50% use means that the home office is not required to be the sole office.
13. Allowing taxpayers to spend work time elsewhere and still get the deduction is wrong, as it is too hard to determine the point at which the amount of time spent elsewhere becomes too much to allow a deduction.

Figure 5.9: Discrediting the existing condition for regular use.

Chapter 6

Argument Views

The analysis of complex cases can result in complicated argument structures that consist of hundreds of nodes and links. Such argument structures are difficult to understand and edit because the argument structure cannot be fully displayed in the editing window, and the size of the argument structure makes it hard to focus on one particular portion or aspect of the argument at a time.

Similar problems have confronted relational database designers in the past, as tables in large databases can contain hundreds of thousands of records that are used by a diverse range of users. The solution that they developed was that of *relational views*. A view is a means of transforming data in such a way as to highlight information that is of interest or to hide information that is not of interest [Elmasri 89]. A relational view may allow the user to only see a subset of the underlying data, or it may contain "virtual" data derived from underlying data. The mapping between the underlying data and the contents of the view is provided by a static query.

Argument views operate in a similar fashion to relational views—an argument view may hide part of the argument structure or it may transform the argument structure to produce new information. The mapping between the underlying argument structure and the displayed argument structure is provided by static algorithms. However argument views differ from relational views in that they are not transparent—they are explicitly activated by the user.

There are two classes of argument views: views that reduce the complexity of large arguments; and views that derive useful information from the argument structure. Section 6.1 describes views that reduce the complexity of the argument. Section 6.2 describes views that derive useful information from the argument structure.

6.1 Views For Dealing With Complex Arguments

The views described in this section allow the user to deal with complex argument structures by hiding the portions of the argument structure that are not of current interest. This may involve hiding groups of related nodes or hiding all nodes of a particular class.

6.1.1 Imploding and Exploding Portions of the Argument Structure

One means of dealing with complexity is to hide groups of related nodes by collapsing or *imploding* a group of nodes down to a single node. This allows the user to more easily focus upon the portions of the argument structure that are of current interest. The remaining node can be *exploded* at a later time in order to reveal the hidden substructure.

Imploding a substructure is initiated by choosing the "Implode Substructure" option in the "View" menu and then clicking on the root of the substructure to be imploded. If the node belongs to more than one substructure, then the user must choose the substructure that is to be imploded from a menu of possible substructures. The substructure is then imploded and the fill pattern of the root node changed to indicate that it is the root of an imploded substructure. Exploding an imploded substructure is initiated by choosing the "Explode Hidden Substructure" option in the "View" menu and then clicking on the root of the imploded substructure.

There are four types of substructures that can be imploded and exploded: argument subtrees, instances of argument packets, user-defined substructures, and substructures associated with strengthening and weakening links.

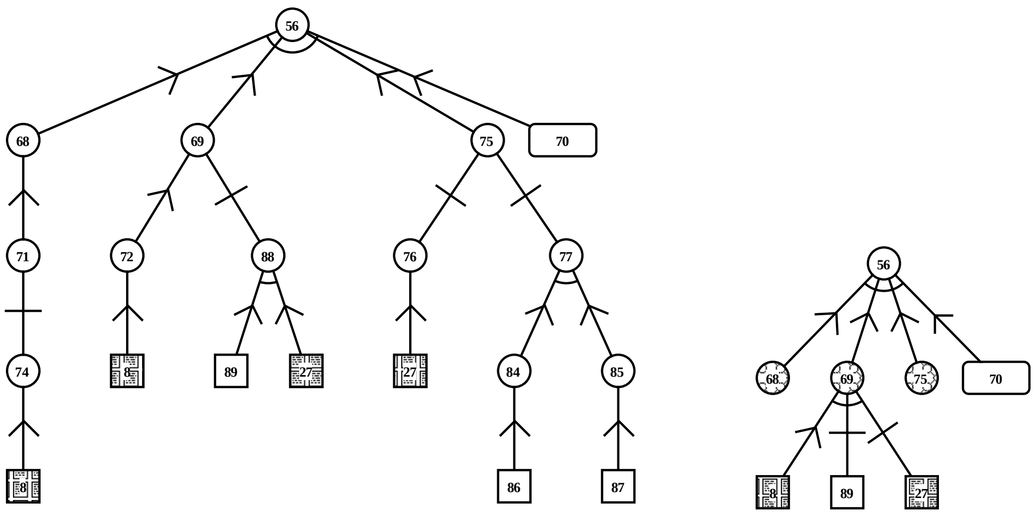
Imploding Argument Subtrees

Two views exist that allow the user to collapse an argument subtree by clicking on the root of the subtree. The first view collapses the entire subtree down to a single node, whereas the second view only collapses the "body" of the subtree, by directly connecting the leaf nodes to the root and hiding all nodes and links in between. Figure 6.1 shows two argument structures. The first argument structure displays the entire subargument relating to provision of consideration under the doctrine of estoppel in *Henderson v BNZ*. The second argument structure displays the same subargument after imploding the entire subtrees representing the behaviour of the bank causing Mr Henderson to believe something (node 68) and the bank's behaviour being unconscionable (node 75), and the body of the subtree representing Mr Henderson acting on the belief to his detriment (node 69).

In addition to hiding part of the argument structure, the second view also shows how the leaf nodes indirectly impact on the root. The types of the new links are determined based upon the types of the links on the path from the root to the leaves—an odd number of rebutting or weakening links results in a rebutting link being created, while an even number results in a supporting link being created. For example, the physical evidence provided by the repayment options letter (node 8) indirectly supports the claim that Mr Henderson suffered a detrimental effect due to his reliance on the belief that the bank would accept an offer from his mother (node 69), as it provides evidence that Mr Henderson made weekly interest payments under the terms of the third option (node 72). However, the physical evidence provided by the testimony of the bank manager (node 89) indirectly rebuts node 69, as it provides evidence that Mr Henderson was already making the weekly interest payments and made no fresh commitment to continue paying them. Note that the view only shows how the leaf nodes *individually* impact on the root—as the view ignores the way in which the leaf nodes *combine* to support the root, nothing can be concluded about the total level of support for the root.

A special case arises when the subtree being imploded contains a node that has multiple parents and one of the parents does not belong to the subtree. In these situations the "external" parent is linked to the root of the subtree when the subtree is imploded, in order to indicate that it is connected to a node in the imploded substructure. For example, node 5 in figure 6.2 is connected to nodes 3 and 9. If the subtree rooted at node 3 is imploded, then node 9 is connected to node 3 as shown in figure 6.2.

A useful feature of this view is that it can be used in a "nested" fashion. For example,



8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, presented to the court.
56. Consideration was provided under the doctrine of promissory estoppel.
68. The behaviour of the offeror (BNZ) was such that it led the offeree (Mr Henderson) to believe something.
69. The offeree (Mr Henderson) acted on that belief to his or her detriment.
70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.
71. The bank represented to the offeree that if Mrs Henderson went to the trouble of making an offer of $30,000, it would accept the offer as repayment of his entire debt.
72. Mr Henderson was making $200 weekly payments under the terms of the agreement.
74. The most the bank could be said to be representing was that it would "consider" any offer by Mrs Henderson.
75. It was unconscionable for the bank to reject Mrs Henderson's offer after her incurring expense in relying on their promise.
76. Mrs Henderson offered less than the specified sum of $30,000.
77. The bank's representation of option three was being taken to mean something quite different than what was intended.
84. The District Office intended that the bank would still receive the proceeds of the mortgagee sale.
85. The offeree was aware that any offer would have to be agreed upon by the District Office.
86. Testimony of District Office staff.
87. Testimony of the offeree (Mr Henderson).
88. Mr Henderson was already making the payments and made no fresh commitment to do so.
89. Testimony of the bank manager.

Figure 6.1: Collapsing argument subtrees.

having imploded part of the subargument relating to estoppel, the user could then collapse the entire estoppel subargument rooted at node 56. When the user wishes to re-examine the subargument dealing with estoppel, the nested nature of the subtrees allows the user to incrementally unfold the entire subargument. Alternatively, the user could choose to explode the subargument in its entirety.

Imploding Instances of Argument Packets

This view allows the user to collapse an instance of an argument packet down to a single node by clicking on any node in the argument packet. The view finds the root and the leaves of the argument packet, and hides the "body" of the argument packet by connecting the children of the leaves (i.e. the nodes immediately outside the argument packet instance) directly to the root. The new links created by the view are given types according to the links on the path from the root to the children of the leaves, in the same manner as for argument subtrees. For example, figure 6.3 shows the effects of imploding the instance of the argument packet for

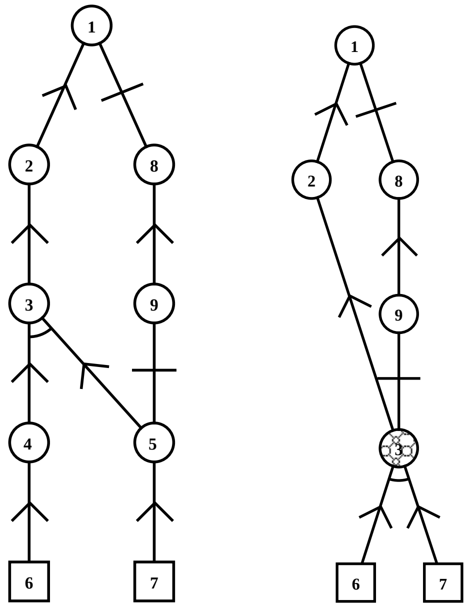


Figure 6.2: Imploding a subtree with links to outside nodes.

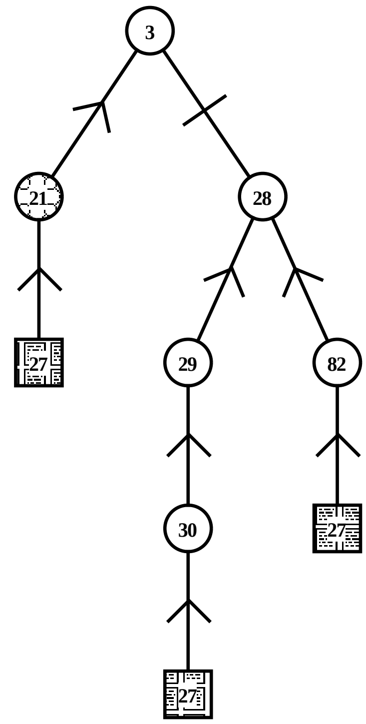
"Communication of Acceptance" in the sub-argument relating to acceptance for *Henderson v BNZ*.

If a node in the argument packet instance being imploded has a parent node that does not belong to the argument packet instance, this is indicated by a link from the root of the imploded packet to the external parent node.

Defining New Substructures

If the user attempts to explode a node that is not the root of an imploded substructure, the system will inform the user that there is no hidden substructure to be exploded. However, the user may still explode the node providing that he or she provides the missing substructure. The user can add nodes to the root until the hidden substructure is fully defined. The new substructure can then be imploded and exploded as if it was an instance of an argument packet.

This view is useful when it is discovered that a link between two nodes actually represents a more complex relationship. The user can explode the node at the head of the link, add more structure underneath the node, and then move the link so that it is attached to one of the leaf nodes in the new structure. For example, the user might decide that the support relationship between the claim that the repayment options letter contained an offer from the BNZ to the Hendersons (node 7) and the physical evidence of the letter itself (node 8) is actually an indirect relationship, involving an intermediate fact. The user can add the new substructure by exploding node 7, adding a new node in support of node 7 (node 93), and moving the link between nodes 7 and 8 to node 93. The newly created substructure can be imploded to show the original relationship, as shown in figure 6.4.

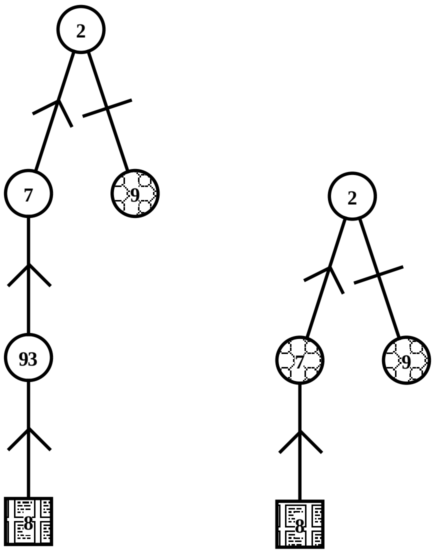


3. The offeree (Mr Henderson) accepted the offer at time t2 (21 June 1988).
21. Legal requirements for communication of acceptance were satisfied at time t2 (21 June 1988).
27. Letter from Mrs Henderson's solicitor, presented to the court.
28. The "acceptance" was really a counter offer that destroyed the original offer.
29. The "acceptance" was phrased in such a way as to be inconsistent with the offer.
30. The letter specified a repayment sum of \$28,000—less than the \$30,000 specified by the bank.
82. The letter concluded "...If such an arrangement can be reached would you kindly advise."

Figure 6.3: Imploding argument packets.

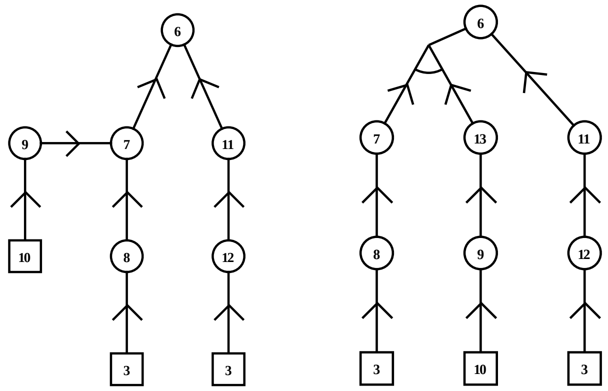
Exploding Strengthening and Weakening Links

Strengthening and weakening links are links that affect the strength of an inference. Analysis of examples has revealed that inferences that are affected by strengthening or weakening links appear to be based upon a hidden assumption (see section 3.2.1). Strengthening links support the hidden assumption, while weakening links rebut the assumption. The view allows the user to explode a strengthening or weakening link by clicking on the node attached to the tail end of the link. When the link is exploded for the first time, the system automatically creates a blank node to represent the hidden assumption. The user can then give the node an appropriate description and add more structure to the argument underlying the assumption. The resulting structure can later be imploded back to the original strengthening or weakening link via the node that the user originally clicked upon to explode the link. An example of exploding a strengthening link in the argument subtree relating to offer for *Henderson v BNZ* is displayed in figure 6.5.



2. The offeror (BNZ) made an offer to the offeree (Mr Henderson) at time t1 (16 May 1988).
7. The offeror (BNZ) sent a letter containing an offer to the offeree (Mr Henderson) at time t1 (16 May 1988).
8. Letter specifying repayment options, as presented to the court.
9. The offer was merely an invitation to treat.
93. The third repayment option contained in the letter was an offer from the offeror (BNZ) to the offeree (Mr Henderson).

Figure 6.4: Defining and imploding a new substructure.

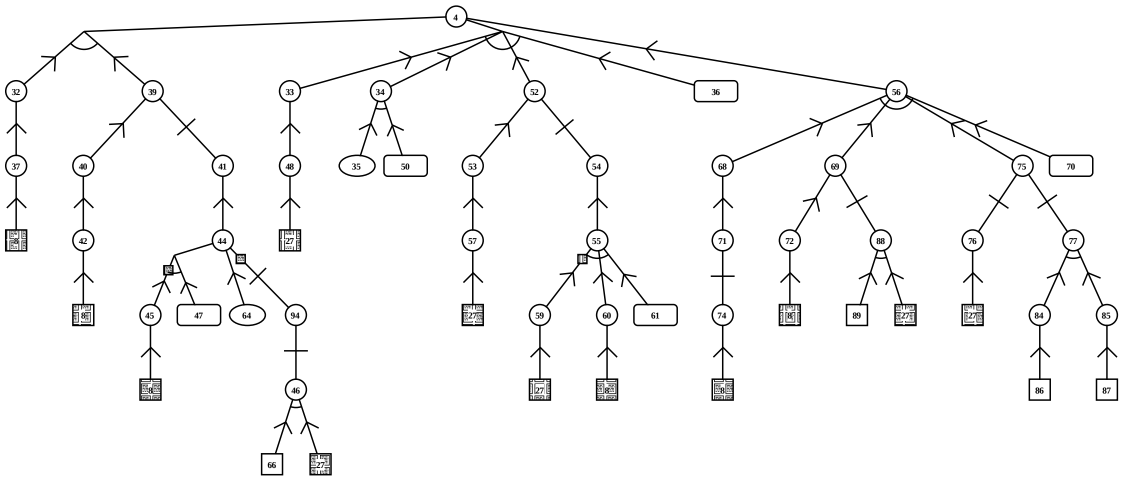


3. Letter specifying repayment options, as presented to the court.
6. The option in question—the third option—was merely an invitation to treat.
7. The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.
8. The third option did not specify a definite repayment sum, but said that it would “consider” a reasonable figure.
9. The bank manager (Mr McAuley) indicated that an opportunity would be given to increase the repayment sum.
10. Testimony of the offeree (Mr Henderson).
11. The third option required someone other than the offeree (Mr Henderson) to take some action.
12. The third option required Mrs Henderson to “. . . make an offer . . .”
13. The bank was inviting Mrs Henderson to commence negotiations.

Figure 6.5: Exploding the strengthening link in *Henderson v BNZ*.

6.1.2 Displaying Warrant Nodes

Warrant nodes are created either explicitly by the user or implicitly when an argument packet is instantiated. By default, the system does not display warrant nodes. However, the user can toggle the display status of warrant nodes by selecting the "Toggle Warrants" option in the Views menu. Figures 6.6 and 6.7 show the sub-argument relating to consideration for *Henderson v BNZ*, with warrants displayed using the graphical styles used to represent the Wigmore notation and Toulmin's notation respectively.



36. If consideration was provided by a third party and the consideration was adequate and the offeree was able to rely on the consideration provided by the third party then valid consideration was provided in exchange for the offer.

47. If the consideration was interest payments for an existing debt then the consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with the offeror.

50. The offeree is able to rely on consideration provided by a third party if the date of the agreement is after 31 March 1983.

61. If the consideration was the part payment of a debt and the offer was the forbearance of the balance of the debt then the consideration was insufficient because it was merely a repayment of existing debt owed by the offeree to the offeror.

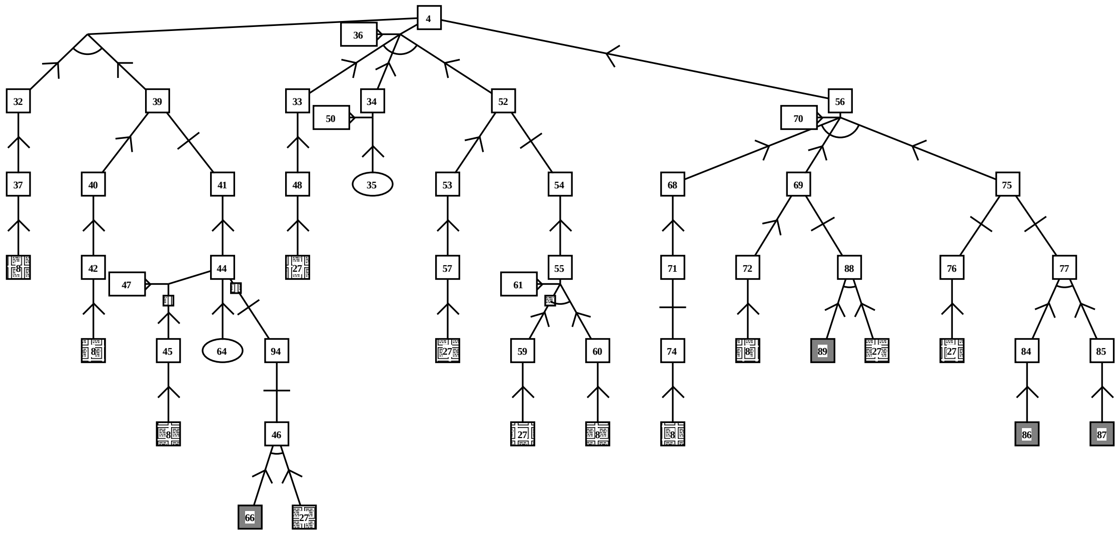
70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.

Figure 6.6: Displaying warrants using the Wigmore notation.

A second view that may be usefully applied to warrants is one that makes explicit the backing for a warrant, as described in section 5.2.2. The view would allow the user to toggle between representing backing for warrants implicitly via document links attached to links in the supporting argument or rebuttal containing the warrant, or by explicitly attaching backing nodes in support of the warrant node. This view has not yet been implemented.

6.2 Views For Deriving Additional Information About An Argument

Three views have been defined that allow the user to derive additional information from an argument. The first two views enable the user to evaluate an argument in terms of how well it is supported and whether there exist any weaknesses to be eliminated or opportunities to be exploited. The third view allows the user to evaluate the effects of making a hypothetical modification to the argument.



36. If consideration was provided by a third party and the consideration was adequate and the offeree was able to rely on the consideration provided by the third party then valid consideration was provided in exchange for the offer.

47. If the consideration was interest payments for an existing debt then the consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with the offeror.

50. The offeree is able to rely on consideration provided by a third party if the date of the agreement is after 31 March 1983.

61. If the consideration was the part payment of a debt and the offer was the forbearance of the balance of the debt then the consideration was insufficient because it was merely a repayment of existing debt owed by the offeree to the offeror.

70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.

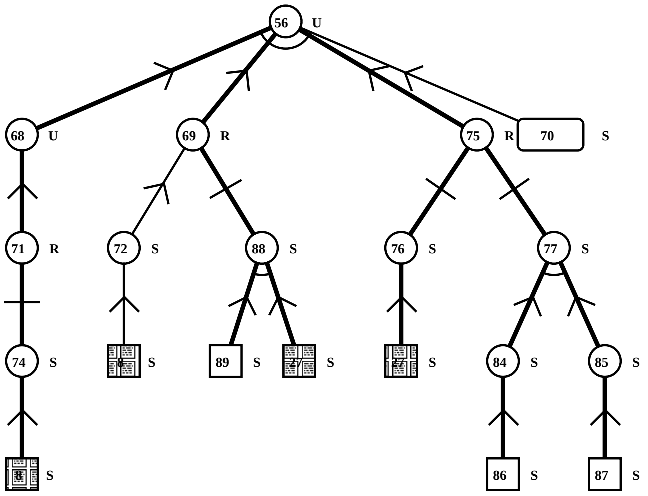
Figure 6.7: Displaying warrants using Toulmin's notation.

6.2.1 Calculating Support For A Node

This view uses the semantics defined in section 3.1.2 to determine the level of support for a node selected by the user. If the selected node is unsupported or rebutted, the view also highlights the chain of argument that caused the node to become unsupported or rebutted. Analysing the highlighted chain of argument will enable the user to determine the changes that need to be made to the argument in order to change the status of the selected node to supported. The changes will typically involve rebutting nodes that are part of the highlighted chain of argument, and adding new supporting arguments for unsupported nodes in the argument. For example, figure 6.8 shows that the claim that consideration was provided in *Henderson v BNZ* under the rule of estoppel (node 56) is unsupported.

This is because only one of the conjuncts in the supporting argument for node 56 is supported (the estoppel rule itself, node 70). All the other conjuncts are unsupported or rebutted, and the highlighted subarguments indicate why. For example, the claim that the behaviour of the BNZ caused the Hendersons to believe something (node 68) is unsupported, as the only supporting argument for the claim is rebutted. In particular, the bank did not represent to the Hendersons that if Mrs Henderson went to the trouble of making an offer, the bank would accept it (as claimed by node 71): the bank only said that it would "consider" any offer (node 74).

The extent to which the subargument is highlighted shows that the estoppel argument is deeply flawed—the argument is rebutted in four separate places (nodes 74, 76, 77 and 88). To repair this argument for estoppel, Henderson's lawyer would have to both discredit these rebuttals and provide supporting arguments for nodes 71 and 75. The magnitude of the changes required suggests that it is not worth pursuing the estoppel subargument further.



8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, presented to the court.
56. Consideration was provided under the doctrine of promissory estoppel.
68. The behaviour of the offeror (BNZ) was such that it led the offeree (Mr Henderson) to believe something.
69. The offeree (Mr Henderson) acted on that belief to his or her detriment.
70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.
71. The bank represented to the offeree that if Mrs Henderson went to the trouble of making an offer of $30,000, it would accept the offer as repayment of his entire debt.
72. Mr Henderson was making $200 weekly payments under the terms of the agreement.
74. The most the bank could be said to be representing was that it would "consider" any offer by Mrs Henderson.
75. It was unconscionable for the bank to reject Mrs Henderson's offer after her incurring expense in relying on their promise.
76. Mrs Henderson offered less than the specified sum of $30,000.
77. The bank's representation of option three was being taken to mean something quite different than what was intended.
84. The District Office intended that the bank would still receive the proceeds of the mortgagee sale.
85. The offeree was aware that any offer would have to be agreed upon by the District Office.
86. Testimony of District Office staff.
87. Testimony of the offeree (Mr Henderson).
88. Mr Henderson was already making the payments and made no fresh commitment to do so.
89. Testimony of the bank manager (Mr McAuley).

Figure 6.8: Calculating the support for the estoppel sub-argument.

6.2.2 Finding Threats and Opportunities

Threats and opportunities arise through the use of secondary nodes, as described in section 3.1.1. This view calculates the current and potential levels of support for a node selected by the user, and highlights any secondary nodes which, if rebutted, would raise or lower the level of support for the selected node. A secondary node whose rebuttal would raise the level of support for the selected node represents an *opportunity* to be exploited. A secondary node whose rebuttal would lower the level of support for the selected node represents a *threat* to be eliminated. The view labels each node in the subtree rooted at the selected node with its current level of support, followed by a plus if the potential level of support for the node is higher than its current level of support, a minus if the potential level of support for the node is lower than its current level of support, or both if the level of support for the node is both potentially higher and potentially lower than its current level of support. The chain of argument that would be affected by a change in the status of a secondary node is indicated by changing the line style of the links in the affected chain of argument to dashes.

For example, when constructing the sub-argument relating to consideration, the counsel for Mr Henderson may make two assumptions: that the opposing counsel will be able to prove that the weekly interest payments made by Mr Henderson were insufficient consideration (node 41); and that he or she can prove that the consideration provided by Mrs Henderson was adequate (node 52). Assumptions such as these will often be made in the early stages of argument construction when there has not been time to collect all of the evidence relating to a claim, but the lawyer is reasonably sure that enough evidence to support the claim will be found.

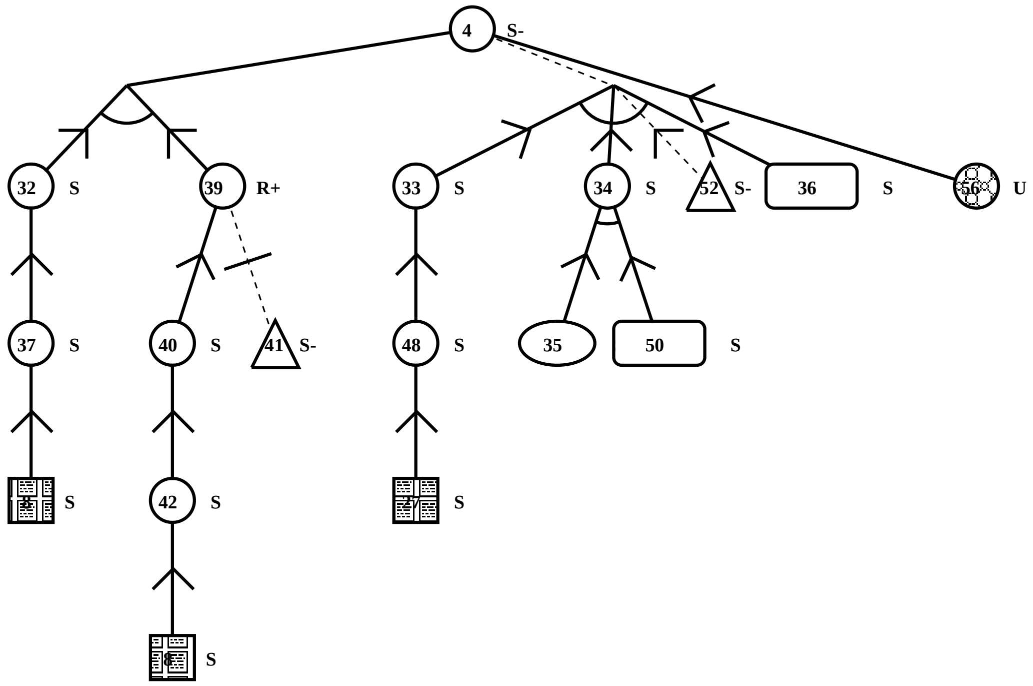
Figure 6.9 displays the result of making such assumptions upon the level of support for the consideration sub-argument. The first assumption creates an opportunity to increase the level of support for the subargument relating to provision of consideration by Mr Henderson. If Mr Henderson's lawyer can successfully rebut the assumption that the weekly interest payments are insufficient consideration (node 41), then the rebuttal of the claim that the interest payments are adequate consideration (node 39) will fail (indicated by the "+" sign after the **R**), and hence the claim will become supported. However, this will have no impact on the overall claim that consideration was provided (node 4), as it is already supported by the subargument relating to the provision of consideration by Mrs Henderson. Thus in the current state of the argument, any change in the status of node 41 will only affect node 39, indicated by the dashed line between nodes 41 and 39.

On the other hand, the secondary node representing the second assumption (node 52) has created a threat to the support for node 4. If node 52 was rebutted, the supporting argument relating to provision of consideration by Mrs Henderson would be rebutted, thus changing the level of support for node 4 to unsupported (hence the "-" sign after the **S**). The threat is indicated by the dashed line running from node 52 to node 4.

6.2.3 Highlighting Nodes Affected By The Testimony of a Given Witness

A strategy often used in court is to attack the credibility of a witness. If the argument of one side substantially depends on the testimony of a particular witness, then casting doubt upon the credibility of the witness severely weakens their argument. A view has been defined that shows the effects of pursuing such a strategy. The user indicates the witness by clicking on an evidence node representing testimony by the witness. The view then evaluates the level of support for the argument, assuming that nodes representing testimony by the witness (evidence nodes that have the same description or template variables as the selected node) are currently supported but potentially rebutted. The view displays the chain of argument that would be affected by discrediting the witness in a similar manner to the view for finding threats and opportunities.

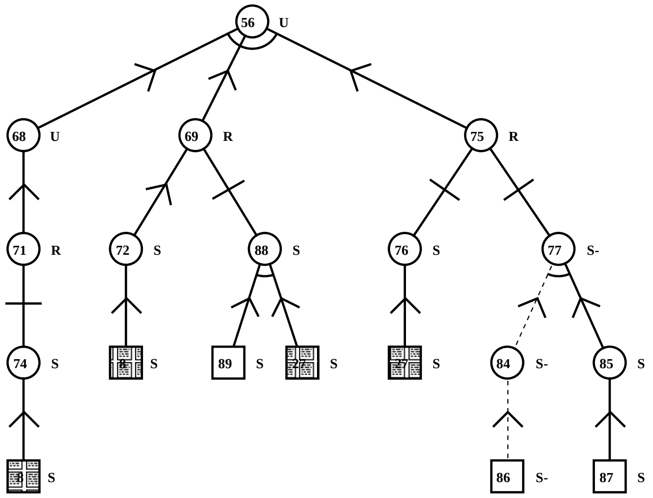
For example, figure 6.10 shows that the support for the claim that consideration was provided under the rule of estoppel (node 56) would be unaffected by any undermining of the testimony of the BNZ District Office staff as to the intentions of the bank with regard to the third option (node 86). The view shows that the chain of argument that would be affected by changing the status of node 86 merely consists of the claim that the District Office staff intended option three to require that the bank still receive the proceeds of a mortgagee sale of



4. Valid consideration was provided in exchange for the offer.
8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, presented to the court.
32. The offeree (Mr Henderson provided the consideration.
33. Consideration was provided by a third party (Mrs Henderson).
34. The offeree (Mr Henderson) was able to rely on the consideration provided by the third party (Mrs Henderson).
35. The date is after 31 March 1983.
36. If consideration was provided by a third party and the consideration was adequate and the offeree was able to rely on the consideration provided by the third party then valid consideration was provided in exchange for the offer.
37. Under the terms of the repayment options letter, the offeree (Mr Henderson) had to make interest payments of $200 per week in the interim.
39. The consideration provided by Mr Henderson was adequate.
40. The consideration provided by Mr Henderson had some intrinsic value.
41. The consideration provided by Mr Henderson was insufficient.
42. The consideration provided by Mr Henderson had a monetary value.
48. Under the terms of the agreement, Mrs Henderson would make a payment of $28,000 to the BNZ.
50. The offeree is able to rely on consideration provided by a third party if the date of the agreement is after 31 March 1983.
52. The consideration provided by the third party (Mrs Henderson) was adequate.
56. Consideration was provided under the doctrine of promissory estoppel.

Figure 6.9: Finding Threats and Opportunities.

the property (node 84), and the more general claim that the bank's representation of option three was being taken to mean something quite different from what was intended (node 77). However, the status of node 75 (the claim that it was unconscionable for the bank to reject Mrs Henderson's offer) would remain unchanged. Although the rebuttal to node 75 provided by node 77 would fail, node 75 is also rebutted by the fact that Mrs Henderson offered less than the sum specified by the bank (node 76). If this separate rebuttal was not present, doubts cast on the testimony of the bank staff would change the level of support for node 75 to unsupported. However, even if node 75 was supported, it would not change the level of support for the claim that consideration was provided under the rule of estoppel (node 56).



8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, presented to the court.
56. Consideration was provided under the doctrine of promissory estoppel.
68. The behaviour of the offeror (BNZ) was such that it led the offeree (Mr Henderson) to believe something.
69. The offeree (Mr Henderson) acted on that belief to his or her detriment.
70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.
71. The bank represented to the offeree that if Mrs Henderson went to the trouble of making an offer of $30,000, it would accept the offer as repayment of his entire debt.
72. Mr Henderson was making $200 weekly payments under the terms of the agreement.
74. The most the bank could be said to be representing was that it would "consider" any offer by Mrs Henderson.
75. It was unconscionable for the bank to reject Mrs Henderson's offer after her incurring expense in relying on their promise.
76. Mrs Henderson offered less than the specified sum of $30,000.
77. The bank's representation of option three was being taken to mean something quite different than what was intended.
84. The District Office intended that the bank would still receive the proceeds of the mortgagee sale.
85. The offeree was aware that any offer would have to be agreed upon by the District Office.
86. Testimony of District Office staff.
87. Testimony of the offeree (Mr Henderson).
88. Mr Henderson was already making the payments and made no fresh commitment to do so.
89. Testimony of the bank manager (Mr McAuley).

Figure 6.10: Displaying the effects of undermining the District Office staff's testimony.

The supporting argument for node 56 (and thus node 56 itself) would remain unsupported due to nodes 68 and 69 not being supported. This suggests that challenging the testimony of the bank staff will not gain much for Mr Henderson, as there are several other weaknesses in the argument that must be dealt with as well.

Chapter 7

Other Issues

Graphical styles provide the link between the conceptual layer and the presentation layer of the argument representation. Section 7.1 describes the graphical styles editor: a subsystem that enables users to switch between graphical styles and to create new styles by editing existing styles. Section 7.2 briefly discusses a number of issues relating to the implementation of the system.

7.1 The Graphical Styles Editor

Graphical styles, as currently implemented, specify how each node and link type should be displayed graphically. Invoking the graphical styles editor results in a window being displayed containing lists of the node and link types defined by the current graphical style, as shown in figure 7.1. The graphical representations of the currently selected node and link types are displayed underneath the appropriate list. The menu bar attached to the window allows the user to load and save graphical styles (via the "File" menu), create new node or link types, or edit the graphical representations of existing types (via the "Node" and "Link" menus).

When a new node or link type is created, the user is prompted to enter the name of the new type along with a brief description of its purpose. The appropriate style editor is then loaded, allowing the user to edit the default graphical representation assigned to the type. If the user wishes to edit an existing node or link type, the appropriate style editor is loaded with the current graphical representation assigned to the chosen type.

7.1.1 The Node Styles Editor

The *Node Styles Editor* allows the user to edit the graphical representation of instances of a selected node type. The editor consists of a window displaying the current graphical representation assigned to the node type and a list of attributes that can be modified by the user to change the way in which instances of the node are presented graphically, as displayed in figure 7.2.

Pop-up menus and other input devices enable the user to change the value for an attribute, with the changes automatically propagating to the displayed instance. The attributes that can be modified are as follows:

- *Node type*: the shape that is used to display instances of the node type. The available shapes include ovals, rectangles, roundtangles (rectangles with rounded corners),

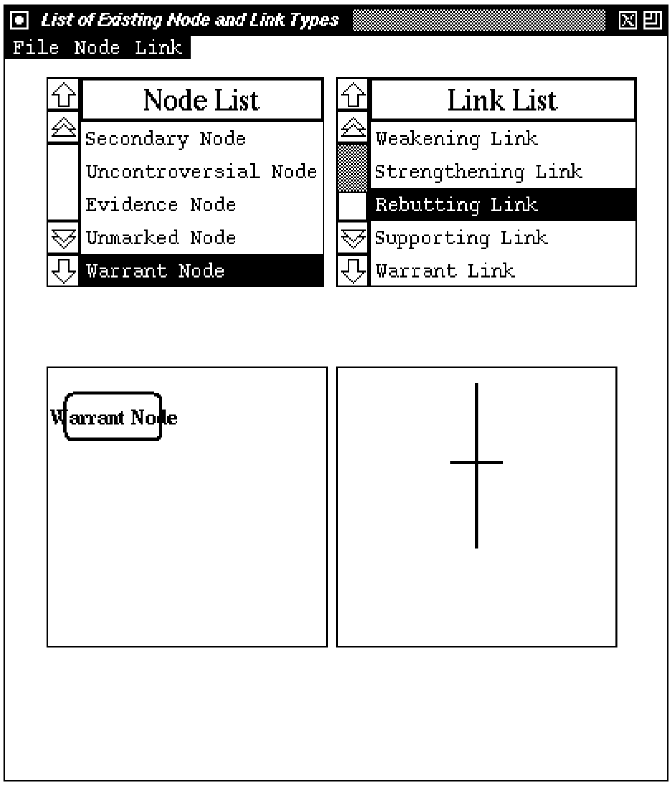


Figure 7.1: The Graphical Styles Editor.

triangles, angles (triangles with one side missing) and general polygons.

- *Node Width*: the width of the displayed node.
- *Node Height*: the height of the displayed node.
- *Fill Style*: the shading used to fill the interior of the displayed node.
- *Node Outline*: the type of line used to enclose the displayed node.
- *Rotation*: the angle of rotation that should be applied to the node before it is displayed. This attribute can be used to convert rectangles into diamonds, for example.
- *Label Position*: the position of the label relative to the displayed node.
- *Font Family*, *Font Face* and *Font Size*: the font used to display the label.

When the desired changes have been made to the graphical representation, the "Exit" button can be used to exit the editor.

7.1.2 The Link Styles Editor

The *Link Styles Editor* allows the user to edit the way in which instances of a given link type are presented graphically. Like the Node Styles Editor, the Link Styles Editor consists of a window displaying the current graphical representation assigned to the link type and a list

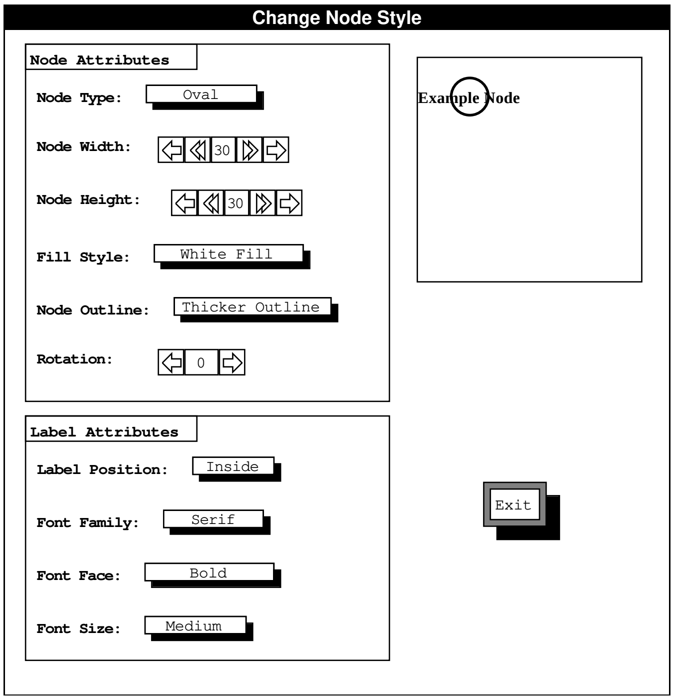


Figure 7.2: Options in the Node Styles Editor.

of attributes that can be modified to change the graphical representation, as shown in figure 7.3.

The attributes that can be modified are as follows:

- *Orientation*: whether the link will be used to connect nodes vertically or horizontally.
- *Line Style*: the type of line used to display the link.
- *Accessories*: this pop-up menu allows the user to add "accessories" to the link. Links start out by default as ordinary lines, to which the user can attach arrowheads, bars, circles (filled or unfilled), crosses or text labels. The user must indicate where on the link the accessory is to be attached.
- *Attachment Point*: for horizontally oriented links, whether the link attaches a node directly to its parent, or whether it attaches to the branching point of the supporting argument or rebuttal containing the link. For example, the warrant link type defined in the graphical style based upon the Toulmin notation is a horizontally oriented link that always attaches to the appropriate branching link. Strengthening and weakening links, on the other hand, attach directly to the parent node.

When the desired changes have been made to the graphical representation, the "Exit" button can be used to exit the editor.

7.2 Implementation Details

A prototype of the system has been developed using *CMU Common Lisp* running on a *Sparc 10* workstation. Graphical support is provided by the *Garnet* toolkit developed by Carnegie

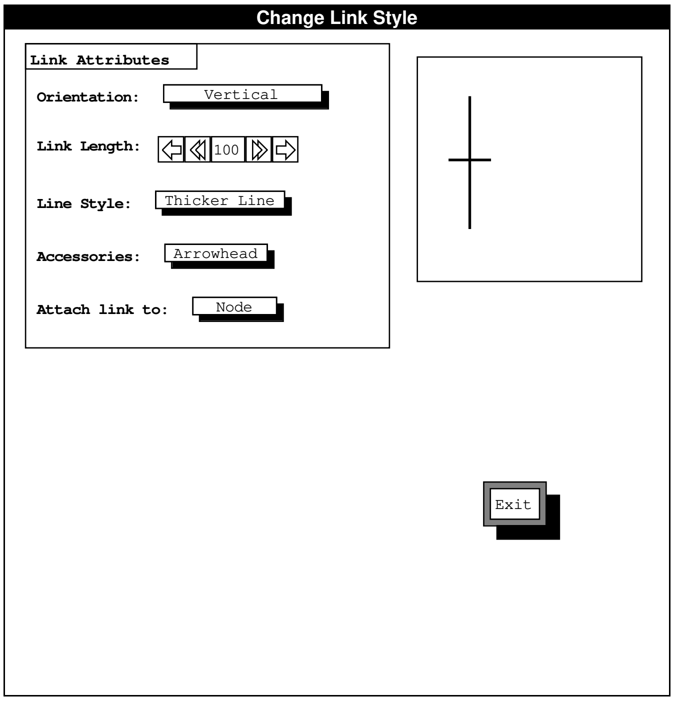


Figure 7.3: Options in the Link Styles Editor.

Mellon University. The majority of the diagrams presented in this dissertation were generated automatically using Garnet functions.

7.2.1 Argument Display

The nodes and links in the argument are laid out using a modified version of Walker's node positioning algorithm for trees [Walker 90]. The modifications relate to the positioning of horizontal links and warrant links—nodes connected by such links should be positioned on the same level even though one node is the "child" of the other. The graphical representations of the nodes and links are drawn on the screen using objects provided by *Opal*, a subsystem of Garnet.

7.2.2 Storing and Retrieving Arguments

Argument structures are currently stored by using the Garnet *write record* function to write the root node for the argument structure to a file. Argument structures are retrieved by using the Lisp *load* function to load the appropriate file.

7.2.3 Document Interface

Documents are displayed using NCSA Mosaic [Braverman 93a]. When the user first attempts to display a document, a mosaic process is spawned and its process identifier kept. The display of further documents is handled by instructing the existing mosaic process to load the new document. This involves writing the filename of the new document to a specific file and sending an interrupt to the mosaic process [Braverman 93b].

Chapter 8

Conclusions

This dissertation describes a prototype system that facilitates the tasks of case preparation and analysis, and evidence management. The system enables users to construct and evaluate arguments, and to store and retrieve documents associated with cases. The system embodies a new model of argument representation, and exemplifies a new framework for designing software for litigation support. Section 8.1 summarises the significant features of the model of argument representation. Section 8.2 describes how the system supports the tasks of case preparation, case analysis, and evidence management. Section 8.3 briefly outlines a number of possible extensions that could be made to the system. The chapter concludes with a brief summary of the project.

8.1 Argument Representation

The model of argument representation presented in this dissertation provides a powerful yet flexible mechanism for representing the structure and content of a legal argument. Significant features of the model include

- A conceptual representation with a well-defined syntax and a clear semantics based on support for propositions.
- Explicit representation of several different types of facts: evidence facts, secondary facts, uncontroversial facts, warrant facts, and unmarked facts.
- An extension that allows propositions to contain variables.
- The elimination of several weaknesses of the Wigmore notation. In particular, a well-defined semantics for strengthening and weakening relationships is provided.

The representation focuses on the structure of the argument as opposed to its content: information about the argument structure is represented formally, while argument content is represented informally. Although the argument model presented in this dissertation has been specifically tailored to the legal domain, it should be easily applicable to arguments in other domains with only minor modifications, such as renaming evidence facts to “grounds”.

8.2 Applications of the System

There are two major applications of the system: *case preparation and analysis* and *evidence management*.

8.2.1 Case Preparation and Analysis

The system supports case preparation and analysis by enabling lawyers to construct and evaluate arguments in support of their clients, and by allowing students to construct and evaluate the arguments of historical cases. Facilities for argument construction and evaluation include the following.

- An argument editor is provided that enables the user to construct, edit, and browse arguments via a graphical user interface. Arguments are displayed in the argument editor using one of a number of predefined *graphical styles*. A graphical styles editor is provided enabling the user to switch graphical styles, and to create new styles by editing the existing styles.

- The construction of arguments is simplified by the provision of a set of *argument packets* encoding standard patterns of inference extracted from the decisions of past cases. Argument packets may be instantiated and copied into the argument structure. The resulting structure contains warrants with implicit backing in the form of the argument packet, along with any precedent cases linked to the argument packet. A set of argument packets has been developed for the domain of contract law.

- Browsing of arguments is facilitated by the provision of a set of *argument views* that enable the user to summarise portions of the argument structure. The views provide a means of focusing on specific portions of the argument. Users can implode substructures that are not of current interest, and explode those that are of interest.

  Of particular interest is a view for exploding strengthening and weakening substructures, which provides a means of making explicit the implicit assumptions upon which the substructures are based. Assumptions such as these often prove to be points of weakness in the argument, and provide opportunities for rebuttal.

- Argument evaluation is facilitated by the provision of several other argument views. One view uses the semantics of the argument representation to determine the level of support for a given fact in the argument, highlighting the source of any rebuttals adversely affecting the support for the fact. A second view enables the user to discover any threats and opportunities that may be present in the argument. A third view demonstrates how the semantics of support for arguments can be used to analyse hypothetical situations, by displaying the effects that undermining the testimony of a witness has upon the status of the argument.

8.2.2 Evidence Management

The system supports evidence management by enabling the user to store and retrieve documents relating both to physical evidence and to relevant past precedents and statutes. The system provides a novel alternative to full text retrieval that uses the structure of the argument to index the documents related to the argument:

- Documents relating to physical evidence may be attached to the evidence nodes that represent the contents of the documents.
- Documents relating to past precedents and statutes may be attached to the links that represent the inferences for which the rules expressed in the documents provide backing. Argument packets provide a means of retrieving the documents for past precedents that are relevant to the case.

To some extent, the tasks of case preparation and evidence management overlap. Retrieving the text of past precedents whose facts are similar to the case at hand enables lawyers to find backing for their arguments. The discovery of new evidence can also lead to modifications to the argument. Conversely, constructing an argument for the case enables the lawyer to determine what issues are important and therefore which past precedents are applicable, and what evidence needs to be found. The system recognises the inter-related nature of case preparation and evidence management by allowing the lawyer to retrieve and attach documents to an argument as it is constructed.

8.3 Future Work

A number of extensions to the system are possible, and are described below.

8.3.1 Reasoning With Arguments

The conceptual representation does not currently enable the user to express how strongly one fact is supported or rebutted by another. This makes it difficult to properly evaluate the argument in terms of how convincing it would be to a judge or jury. An argument that is logically sound (supported) but based upon very weak premises may be less convincing than an argument that is rebutted by a subargument that is itself based on weak premises. There are a number of methods that could be used to express and reason about the strength of arguments. Clark's model of argument emphasised simplicity by using a simple multiplicative calculus to derive the strength of a claim from the strengths of its grounds and warrants [Clark 91]. A more complex yet more powerful alternative is that of Bayesian reasoning [Robertson 91].

8.3.2 Automating the Argument Packet Interface

At present, the application of argument packets to argument structures is almost completely directed by the user. Unfortunately, this assumes a certain amount of domain knowledge on the part of the user, as he or she must be able to recognise situations in which a given argument packet is applicable—all the system does to aid the user is to list the argument packets that could be used to flesh out a given node. However, the question arises—how can the system determine which argument packets are applicable to the given case? One possibility would involve representing the *content* of argument structures and argument packets in more detail. At present, facts are simply represented by strings of characters that may contain references to variables. An alternative would be to represent facts formally—this would allow the system to better match the content of argument packets with the facts of the case. However, this raises additional problems: if possible, the user should not have to deal with facts at the level of the formal representation, but instead should be permitted unrestricted text input. Unfortunately, this seems to require an algorithm for translating natural language sentences entered by the user into the formal representation for the corresponding facts. Developing and implementing such an algorithm is too hard a problem at present. Furthermore, developing a formal knowledge base for the legal domain is itself a challenging task. (For one effort in this direction, see [Restatement 81].)

8.3.3 Storing Arguments

The argument packet hierarchy provides one means of retrieving relevant precedents. An alternative basis for retrieval could be provided by storing the argument associated with a case along with the text of the court's decision in the case database. Case-based reasoning techniques could then be used to match the argument for the current case with those stored in the case database in order to find relevant past precedents. Again, this raises the difficult issue of the extent to which argument content should be represented.

8.3.4 The User Interface

Some aspects of the user interface need tidying up. The tree-drawing algorithm used for positioning nodes in the argument structure on the screen needs to be replaced by an algorithm that is able to handle a wide variety of methods for laying out arguments. The general speed of the user interface could also be improved by re-implementing the prototype using a language such as C.

8.3.5 Better Graphical Styles

The current implementation of graphical styles is rather inflexible. It is impossible to accurately represent notations that do not lay out arguments in a top-down fashion, or whose syntax does not exactly mirror that of the conceptual representation. At present, graphical styles merely consist of a list of graphical representations for each node and link type. What is needed is a more flexible representation for graphical styles, that is able to accurately map the information contained in the conceptual representation to other notations such as the Toulmin notation. This requires a careful analysis of the existing notations and their mapping to the conceptual representation, in order to determine the type of information that must be able to be represented by graphical styles.

8.3.6 Generating the Text of Arguments

The system currently displays the contents of the argument in the form of a list of text descriptions for each node. A useful extension involves using the argument structure and description list to generate a "spoken" argument (i.e. one that could be presented in court) for one side in the dispute, citing supporting arguments and precedents and dealing with any rebuttals. This could be presented in the form of a document whose sentences are linked to the portions of the argument structure to which they relate, in a similar manner to the relationships between documents and arguments in Euclid [Smolensky 87].

8.4 Summary

This dissertation has described a representation of argument structure that is richer than previous representations. The representation has been used in the development of a prototype system to support litigators in the tasks of case preparation and analysis, and evidence management. The system goes beyond previous systems for litigation support, but much work remains to be done.

Bibliography

[Anderson 91] Terence Anderson and William Twining, "Analysis of Evidence", Little, Brown & Co., Boston, 1991.

[Ashley 87] Kevin D. Ashley and Edwina L. Rissland, "Compare and Contrast, A Test of Expertise", in *Proceedings AAAI-87*, American Association for Artificial Intelligence, Seattle, August 1987.

[Ashley 88] Kevin D. Ashley and Edwina L. Rissland, "Waiting on Weighting: A Symbolic Least Commitment Approach", in *Proceedings AAAI-88*, American Association for Artificial Intelligence, Minneapolis, 1988.

[Berners-Lee 93] Tim Berners-Lee and Daniel Connolly, "Hypertext Markup Language (HTML)", Document URL http://info.cern.ch/hypertext/WWW/Markup/HTML.html on the *World Wide Web.*

[Bing 78] J. Bing, "Legal Information Retrieval Systems: The Need for and Design of Extremely Simple Retrieval Strategies", in *Computer and Law Journal*, Volume 1, Number 2, 1978.

[Birnbaum 80] Lawrence Birnbaum, "Representing Argument Structure", unpublished, 1980.

[Blair 85] D. C. Blair and M. E. Maron, "An Evaluation of Retrieval Effectiveness for a Full-Text Document Retrieval System", in *Communications of the ACM*, Volume 28, Number 3, 1985.

[Blanchard 92] Blanchard J, *Henderson v BNZ (transcript of judgment)*, High Court, Whangarei, 10 December 1992. (M 45/89)

[Branting 91] L. Karl Branting and Bruce W. Porter, "Rules and Precedents as Complementary Warrants", in *Proceedings AAAI-91*, American Association for Artificial Intelligence, 1991.

[Braverman 93a] Alan M. Braverman, "Using NCSA Mosaic", Document URL http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/d2-userguide.html on the *World Wide Web.*

[Braverman 93b] Alan M. Braverman, "Using Mosaic by Remote Control", Document URL http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/remote-control.html on the *World Wide Web.*

[Clark 90] Peter Clark, "Representing Knowledge As Arguments: Applying Expert System Technology to Judgemental Problem-solving", in *Research and Development in Expert Systems VII*, Cambridge University Press, 1990.

[Clark 91] Peter Clark, "A Model of Argumentation and Its Application in a Cooperative Expert System", PhD Dissertation, Department of Computer Science, University of Strathclyde, Glasgow, 1991.

[Dung 93] Phan Minh Dung, "On the Acceptability of Arguments and Its Fundamental Role in Nonmonotonic Reasoning and Logic Programming", in *Proceedings IJCAI-93*, Chambery, France, 1993.

[Elmasri 89] Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Benjamin Cummings Publishing, 1989.

[Foakes 84] *Foakes v Beer*, (1884) 9 App Cas 605.

[Gardner 87] Anne Gardner, "An Artificial Intelligence Approach to Legal Reasoning", MIT Press, 1987.

[Ginsberg 93] Matt Ginsberg, "Essentials of Artificial Intelligence", Morgan Kaufmann, 1993.

[Greenleaf 93] Graham Greenleaf and Andrew Mowbray, "Information Technology in Complex Criminal Trials", Report to the Australian Institute of Judicial Administration, May 1993.

[Hafner 78] Carole D. Hafner, "An Information Retrieval System Based on a Computer Model of Legal Knowledge", PhD Dissertation, University of Michigan, 1978.

[Hafner 87] Carole D. Hafner, "Conceptual Organisation of Case Law Knowledge Bases", in *Proceedings of the First International Conference on Artificial Intelligence and Law*, ACM Press, 1987.

[Hammond 86] Kristian D. Hammond, "CHEF: A Model of Case-based Planning", in *Proceedings AAAI-86*, American Association for Artificial Intelligence, Philadelphia, Pa., August 1986.

[Jones 91] Eric K. Jones, "The Flexible Use of Abstract Knowledge in Planning", PhD Dissertation, Yale University, 1991.

[Kass 89] Alex Kass, "Adaptation-Based Explanation: Extending Script/Frame Theory to Handle Novel Input", in *Proceedings IJCAI-89*, Detroit, Michigan, 1989.

[McCarty 81] L. Thorne McCarty and N. S. Sridharan, "The Representation of an Evolving System of Legal Concepts: II. Prototypes and Deformations", in *Proceedings IJCAI-81*, William Kaufmann, California, 1981.

[McCarty 83] L. Thorne McCarty, "Intelligent Legal Information Systems: Problems and Prospects", in *Rutgers Computers and Technology Law Journal*, Volume 9, Number 2, 1983.

[McGuire 81] Rod McGuire, Lawrence Birnbaum and Margot Flowers, "Opportunistic Processing in Arguments", in *Proceedings IJCAI-81*, William Kaufmann, California, 1981.

[NZLR 72] *R v Thomas*, [1972] NZLR 34.

[Perelman 69] CH. Perelman and L. Olbrechts-Tyteca, "The New Rhetoric: A Treatise on Argumentation", Notre Dame Press, 1969.

[Restatement 81] "Restatement of the Law, Second: Contracts 2d", St. Paul:American Law Institute Publishers, 1981.

[Rissland 84] Edwina L. Rissland, Eduardo M. Valcarce and Kevin D. Ashley, "Explaining and Arguing With Examples", in *Proceedings AAAI-84*, American Association for Artificial Intelligence, 1984.

[Rissland 85] Edwina L. Rissland, "AI and Legal Reasoning", in *Proceedings IJCAI-85*, Los Angeles, 1985.

[Rissland 89] Edwina L. Rissland and David B. Skalak, "Combining Case-Based and Rule-Based Reasoning: A Heuristic Approach", in *Proceedings IJCAI-89*, Detroit, 1989.

[Rissland 90] Edwina L. Rissland, "Artificial Intelligence and Law: Stepping Stones to a Model of Legal Reasoning", in *Yale Law Journal*, Volume 99, Number 8, 1990.

[Robertson 90] Bernard Robertson, "John Henry Wigmore and Arthur Allan Thomas—An example of Wigmorian Analysis", in *VUW Law Review*, Volume 20, Victoria University of Wellington, 1990.

[Robertson 91] Bernard Robertson and G A Vignaux, "Extending the Conversation About Bayes", in *Cardozo Law Review*, Volume 13, Benjamin N. Cardozo School of Law, New York, 1991.

[Routen 92] Tom Routen, "Reusing Formalisations of Legislation in a Tutoring System", in *Artificial Intelligence Review*, Volume 6, 1992.

[Sartor 91] Giovanni Sartor, "The Structure of Norm Conditions and Nonmonotonic Reasoning in Law", in *Proceedings of the Third International Conference on Artificial Intelligence & Law*, ACM Press, 1991.

[Skalak 92] David B. Skalak and Edwina L. Rissland, "Arguments and Cases: An Inevitable Intertwining", in *Artificial Intelligence and Law*, Volume 1, Number 1, 1992.

[Smolensky 87] Paul Smolensky, Brigham Bell, Barbara Fox, Roger King and Clayton Lewis, "Constraint-based Hypertext for Argumentation", in *Hypertext '87 Proceedings*, ACM Press, 1987.

[Toulmin 58] Stephen Toulmin, "The Uses of Argument", Cambridge University Press, 1958.

[Toulmin 79] Stephen Toulmin, Richard Rieke and Allan Janik, "An Introduction to Reasoning", Macmillan Publishing, 1979.

[Walker 90] John Q. Walker II, "A Node-positioning Algorithm for General Trees", in *Software—Practice and Experience*, Volume 20, Number 7, July 1990.

[Wigmore 13] John H. Wigmore, "The Problem of Proof", in *Illinois Law Review*, Volume VIII, Number 2, June 1913.

[Williams 91] *Williams v Roffey Bros.*, [1991] 1 QB 1.

[Wilson 92] Eve Wilson, "Guiding lawyers: mapping law into Hypertext", in *Artificial Intelligence Review*, Volume 6, 1992.

Appendix A

Argument Packets for Contract Law

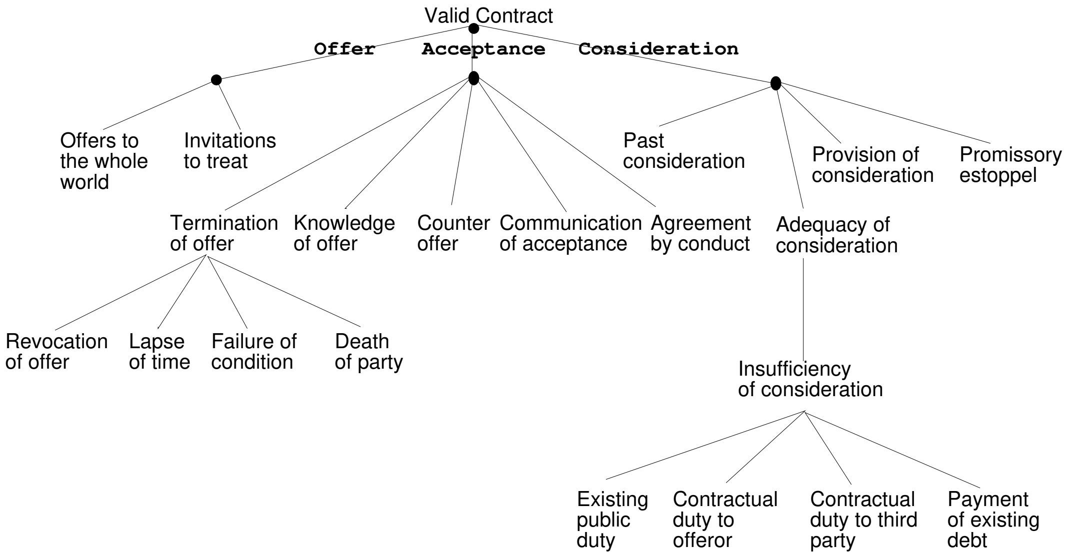
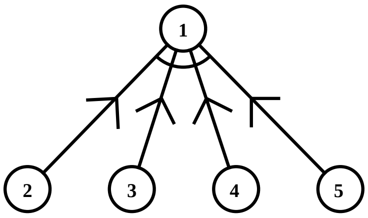
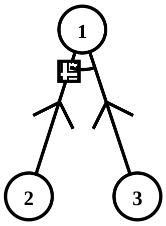


Figure A.1: The argument packet hierarchy for the Contract Law domain.



1. A valid contract exists between the offeror and the offeree.
2. The offeror made an offer to the offeree at time t1.
3. The offeree accepted the offer at time t2.
4. Valid consideration was provided in exchange for the offer.
5. The offeror and offeree intended for the agreement to have legal consequences.

Figure A.2: The argument packet for contracts.

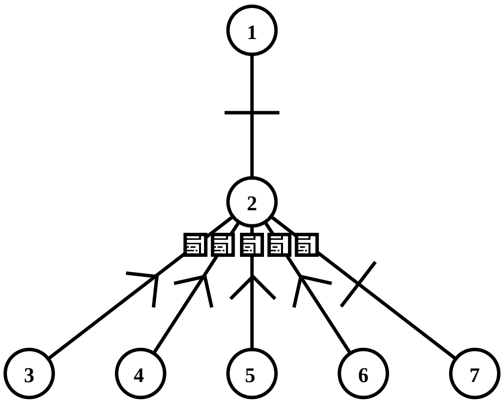


1. The offeror made an offer to the offeree at time t1.
2. The offer was made to the whole world.
3. The offeree is part of the whole world.

Cases:

2 → 1: *Carlill v Carbolic Smoke Ball Co.*

Figure A.3: The argument packet for offers to the whole world.



1. The offeror made an offer to the offeree at time t1.
2. The offer was merely an invitation to treat.
3. The offer was merely a request for bids.
4. The offer was merely a statement of intention.
5. The offer was merely a preliminary statement of price.
6. The offer merely consisted of a display of priced goods.
7. The offer contained an express or implied promise relating to the consideration of competing bids.

<u>Cases:</u>

$3 \rightarrow 2$: *Payne v Cave*

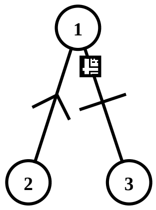
$4 \rightarrow 2$: *Harris v Nickerson*

$5 \rightarrow 2$: *Harvey v Facey, Clifton v Palumbo, Gibson v Manchester City Council*

$6 \rightarrow 2$: *Pharmaceutical Society of Great Britain v Boots Cash Chemists (Southern) Ltd, Fisher v Bell*

$7 + 2$: *Harvela Investments Ltd v Royal Trust Co. of Canada Ltd, Markholm Construction Ltd v Wellington City Council, Blackpool Fylde Aero Club Ltd v Blackpool Borough Council*

Figure A.4: The argument packet for invitations to treat.

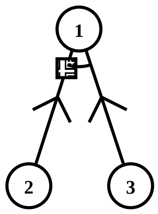


1. The offeree accepted the offer at time t2.
2. Acceptance was made by fulfilling the terms of the offer.
3. At time t2, the offeree had no knowledge of the offer.

<u>Cases:</u>

$3 + 1$: *R v Clarke, Williams v Carwardine*

Figure A.5: The argument packet for knowledge of offer.

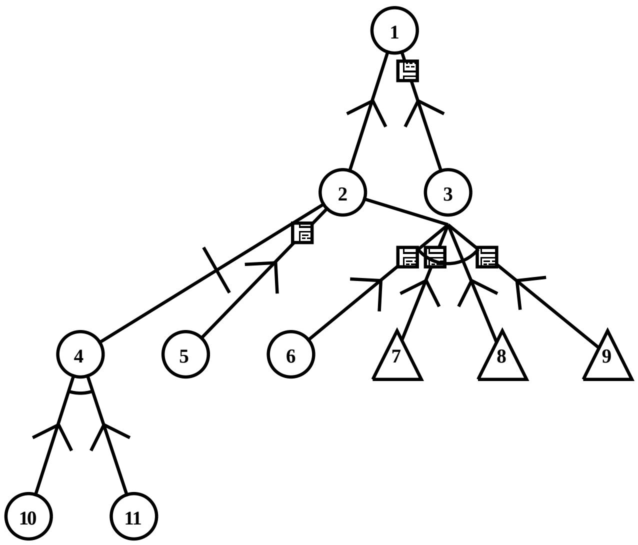


1. The offeree accepted the offer at time t2.
2. The offeror and offeree subsequently acted as if a contract existed.
3. The first time that the offeror and offeree behaved in such a manner was at time t2.

Cases:

2 → 1: *Brogden v Metropolitan Railway Co.*

Figure A.6: The argument packet for agreement inferred by conduct.



1. The offeree accepted the offer at time t2.
2. Legal requirements for communication of acceptance were satisfied at time t2.
3. The offeror expressly or implicitly waived the requirement of communication of acceptance.
4. The offeree used an inappropriate mode of communication.
5. The offeror received the communication at time t2.
6. The offeree posted a letter of acceptance at time t2.
7. The offeror did not specify a mode of communication other than via post.
8. The offeror and offeree did not intend for there to be no agreement until the offeror received the communication.
9. The letter of acceptance was properly stamped and addressed.
10. The offeror specified a particular mode of communication.
11. The offeree used a less expeditious method of communication.

Cases:

3 → 1: *Carlill v Carbolic Smoke Ball Co.*

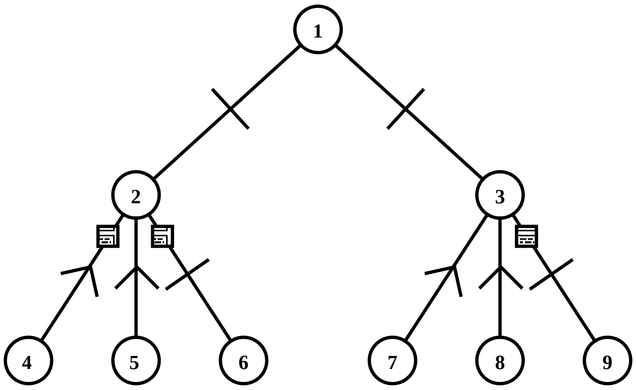
5 → 2: *Entores Ltd v Miles Far East Corporation*

6 → 2: *Adams v Lindsell, Byrne v Van Tienhoven*

7 → 2: *Holwell Securities Ltd v Hughes*

9 → 2: *Re London and Northern Bank*

Figure A.7: The argument packet for communication of acceptance.



1. The offeree accepted the offer at time t2.
2. Prior to time t2, the offeree made a counter offer to the offeror, destroying the original offer.
3. The "acceptance" was really a counter offer that destroyed the original offer.
4. The offeree made reference to a new term in a communication to the offeror.
5. The offeree made a communication to the offeror that was inconsistent with the offer.
6. The communication was merely a request for information.
7. The "acceptance" made reference to a new term.
8. The "acceptance" was phrased in such a way as to be inconsistent with the offer.
9. The communication was an acceptance, together with a request to consider different terms.

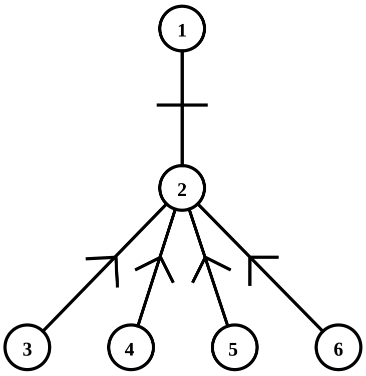
<u>Cases:</u>

4 → 2: *Hyde v Wrench*

6 + 2: *Stevenson v MacLean*

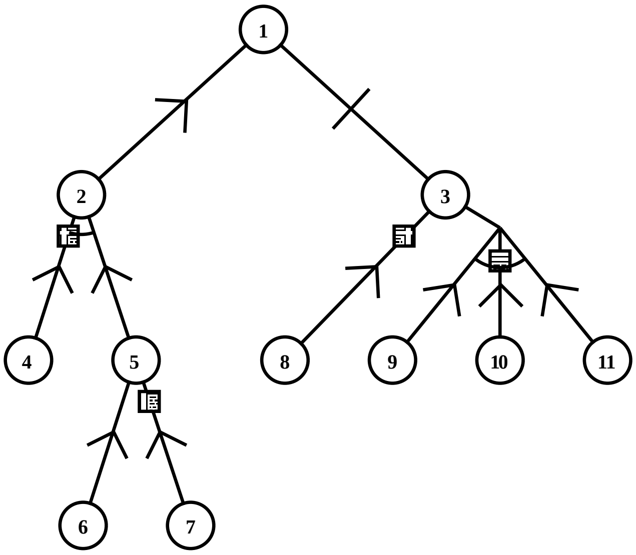
9 + 3: *Powierza v Daley*

Figure A.8: The argument packet for counter offer.



1. The offeree accepted the offer at time t2.
2. The offer was terminated by the offeror before time t2.
3. The offeror revoked the offer before time t2.
4. The offer lapsed before time t2.
5. A condition subject to which the offer was made was broken before time t2.
6. The offeror or offeree died before time t2.

Figure A.9: The argument packet for termination of offer.



1. The offeror revoked the offer before time t2.
2. The revocation was communicated to the offeree before time t2.
3. The offeror was not able to revoke the offer before time t2.
4. The offeree received the communication before time t2.
5. The revocation was communicated by the offeror or by a third party.
6. The revocation was communicated by the offeror.
7. The revocation was communicated by a third party.
8. A separate contract prevented the offeror from revoking the offer before time t2.
9. The offer was in the form of a promise by the offeror in return for an act by the offeree.
10. The offer implied that the offeror would not revoke the offer if the offeree began performing the act within a reasonable time.
11. The offeree had begun performing the act.

Cases:

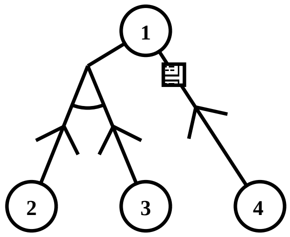
$4 \rightarrow 2$: *Routledge v Grant, Byrne v Van Tienhoven*

$7 \rightarrow 5$: *Dickinson v Dodds, Cartwright v Hoogstoel*

$8 \rightarrow 3$: *Great Northern Railway Co. v Witham*

$10 \rightarrow 3$: *Errington v Errington, Luxor (Eastbourne) Ltd v Cooper*

Figure A.10: The argument packet for revocation of offer.

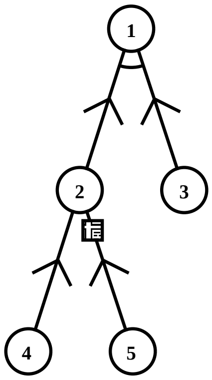


1. The offer lapsed before time t2.
2. The offer stated that it was open until time t3.
3. Time t3 is before time t2.
4. Acceptance was not made within a reasonable time.

<u>Cases:</u>

4 → 1: *Ramsgate Victoria Hotel Co v Montefore, Loring v City of Boston*

Figure A.11: The argument packet for lapse of time.

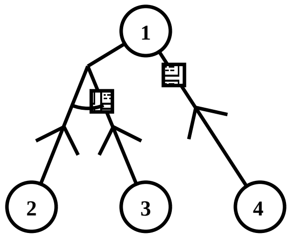


1. A condition subject to which the offer was made was broken before time t2.
2. The condition was expressly or implicitly stated in the contract.
3. The condition was broken before time t2.
4. The condition was expressly stated in the contract.
5. The condition was implicitly stated in the contract.

<u>Cases:</u>

5 → 2: *Financing Ltd v Stimson*

Figure A.12: The argument packet for failure of subjective condition.



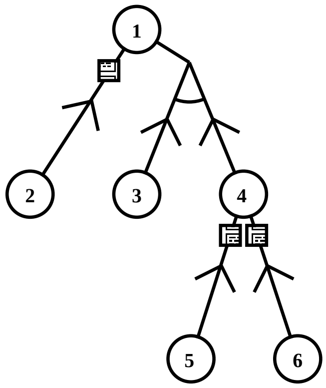
1. The offeror or offeree died before time t2.
2. The offeror died before time t2.
3. The offeree learned of the death of the offeror before time t2.
4. The offeree died before time t2.

<u>Cases:</u>

3 → 1: *Bradbury v Morgan, Re Whelan, Coulthart v Clementson*

4 → 1: *Reynolds v Atherton, Re Irvine*

Figure A.13: The argument packet for death of offeror or offeree



1. Valid consideration was provided in exchange for the offer.
2. Consideration was provided by the offeree.
3. Consideration was provided by a third party.
4. The offeree was able to rely on the consideration provided by the third party.
5. The offeree and the third party are joint promisees.
6. Time t2 is after 31 March 1983.

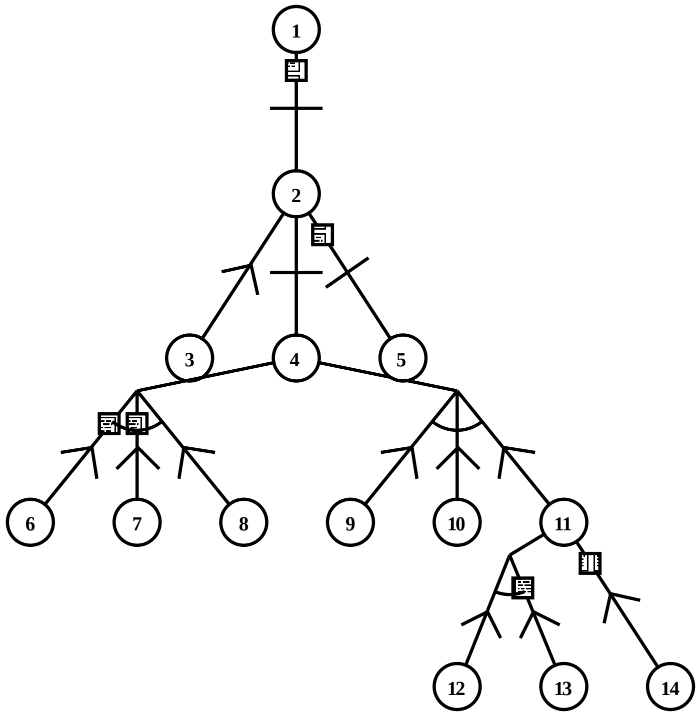
<u>Cases:</u>

2 → 1: *Tweddle v Atkinson*

5 → 4: *Coulls v Bagot's Executor and Trustee Co. Ltd, McEvoy v Belfast Banking Co. Ltd*

6 → 4: *Contracts (Privity) Act 1982*

Figure A.14: The argument packet for provision of consideration.



1. Valid consideration was provided in exchange for the offer.
2. The consideration was provided prior to and independently of the offer.
3. The consideration was provided prior to the offer being made.
4. The consideration and the offer were related.
5. The offer was a promise to make testamentary provision.
6. The consideration was a service provided at the request of the offeror.
7. The offeror and offeree understood that the service was to be remunerated by a payment or some benefit.
8. Had the offeror promised in advance to remunerate the offeree, the promise would have been legally enforceable.
9. The consideration was the waiving of a previous debt owed by the offeror to the offeree.
10. The offer was a promise to pay the previous debt.
11. The promise is legally enforceable due to statute.
12. The debt had been barred.
13. After the debt had been barred, the offeror acknowledged the debt.
14. The promised payment was in the form of a bill of exchange.

Cases:

2 ┼ 1: *Roscorla v Thomas, Re McArdle*

5 ┼ 2: *Law Reform (Testamentary Promises) Act 1949*

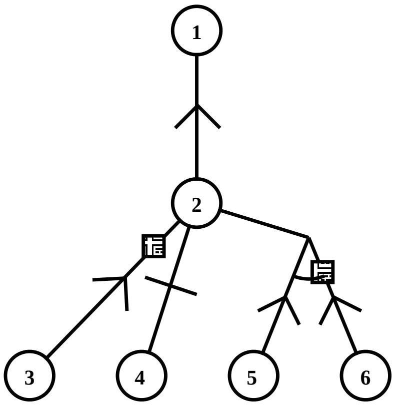
6 → 4: *Lampleigh v Braithwait*

7 → 4: *Stewart v Casey, Casey v Commissioner of Inland Revenue, Teo Tipene v Mere te Puni*

13 → 11: *Limitation Act 1950*

14 → 11: *Bills of Exchange Act 1908*

Figure A.15: The argument packet for past consideration.



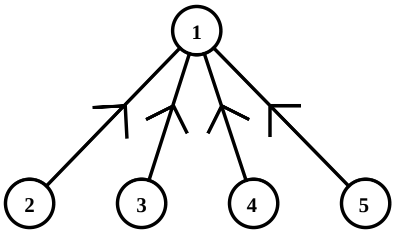
1. Valid consideration was provided in exchange for the offer.
2. The consideration was adequate.
3. The consideration had some intrinsic value.
4. The consideration was insufficient.
5. The consideration involved the surrender of a claim to a legal right.
6. The offeree honestly believed that the claim could succeed.

Cases:

3 → 2: *Thomas v Thomas, Re Murphy, Chappell & Co. Ltd v Nestle Co. Ltd, Pollway Ltd v Abdullah*

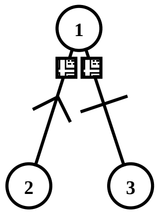
6 → 2: *Couch v Branch Investments Ltd, Paulger v Butland Industries Ltd*

Figure A.16: The argument packet for adequacy of consideration.



1. The consideration was insufficient.
2. The consideration was insufficient because the offeree was already legally bound to provide it due to an existing public duty.
3. The consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with the offeror.
4. The consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with a third party.
5. The consideration was insufficient because it was merely a repayment of existing debt owed by the offeree to the offeror.

Figure A.17: The argument packet for insufficiency of consideration.



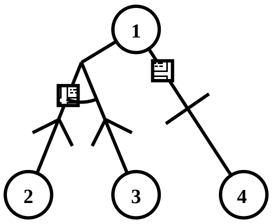
1. The consideration was insufficient because the offeree was already legally bound to provide it due to an existing public duty.
2. The offeree was already legally bound to provide the consideration because of an existing public duty.
3. The consideration provided by the offeree exceeded that which the offeree was legally bound to provide.

Cases:

2 → 1: *Collins v Godefroy*

3 ⊣ 1: *Glasbrook Brothers v Glamorgan County Council, Black White and Grey Cabs Ltd v Reid, Ward v Byham*

Figure A.18: The argument packet for insufficiency of consideration due to an existing public duty.



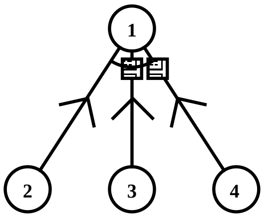
1. The consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with the offeror.
2. The offeree was already legally bound to provide the consideration under the terms of a previous contract with the offeror.
3. The offeree took advantage of the offeror's position.
4. The consideration provided by the offeree resulted in the offeror obtaining an additional benefit.

Cases:

2 → 1: *Stilk v Myrick, Cook Island Shipping Co. Ltd v Colson Builders Ltd*

4 ⊣ 1: *Moyes v Groves Ltd v Radiation (NZ) Ltd, Hartley v Ponsonby, Williams v Roffey Bros. & Nicholls (Contractors) Ltd*

Figure A.19: The argument packet for insufficiency of consideration due to an existing contractual duty to the offeror.



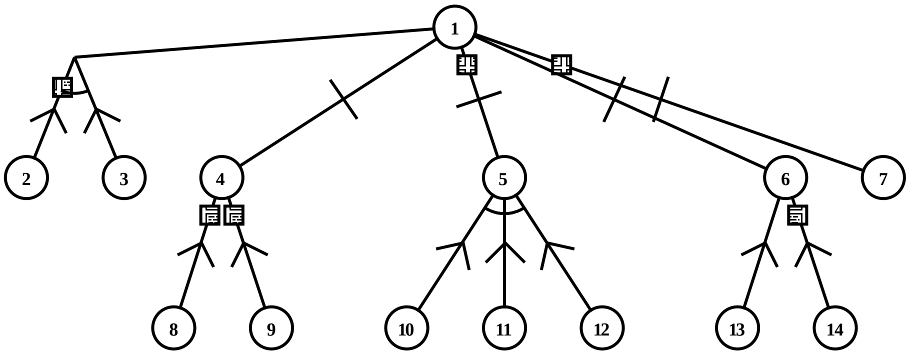
1. The consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with a third party.
2. The offeree was already legally bound to provide the consideration under the terms of a contract with a third party.
3. The offeror did not obtain an additional benefit from the provision of consideration by the offeree.
4. The offeree did not suffer an additional detriment by providing the consideration.

Cases:

3 → 1: *New Zealand Shipping Co. v AM Satterthwaite & Co. (The Eurymedon)*

4 → 1: *Shadwell v Shadwell, Scotson v Pegg*

Figure A.20: The argument packet for insufficiency of consideration due to an existing contractual duty to a third party.



1. The consideration was insufficient because it was merely a repayment of existing debt owed by the offeree to the offeror.
2. The consideration was the part payment of a debt.
3. The offer was the forbearance of the balance of the debt.
4. The part payment was made in a different fashion to that specified by the debt.
5. The part payment was sufficient consideration under the Judicature Act 1908.
6. There was uncertainty surrounding the amount of the debt.
7. Recovery of the entire debt would result in a fraud being committed.
8. Payment was made earlier than required.
9. Payment was made in a different form to that specified by the debt.
10. The offeror had received the part payment.
11. The offeror acknowledged that the part payment was in satisfaction of the entire debt.
12. The acknowledgement was made in writing and signed by the offeror or his or her agent.
13. The claim is for an unliquidated sum.
14. The amount of the debt is in dispute.

<u>Cases:</u>

2 → 1: *Pinnel's Case, Foakes v Beer*

5 $+$ 1: *Judicature Act 1908*

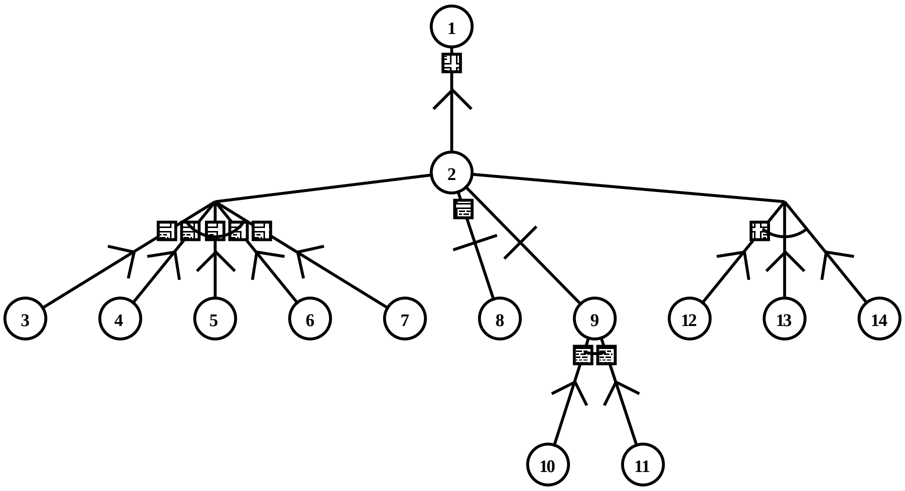
7 $+$ 1: *Welby v Drake*

8 → 4: *Pinnel's Case*

9 → 4: *Sibree v Tripp, D and C Builders Ltd v Rees*

14 → 6: *Homeguard Products (NZ) Ltd v Kiwi Packaging Ltd*

Figure A.21: The argument packet for part payment of debt.



1. Valid consideration was provided in exchange for the offer.
2. Consideration was provided under the doctrine of promissory estoppel.
3. The offer took the form of a promise not to enforce certain contractual rights.
4. The offeree relied on the promise to his or her detriment.
5. The offeror and offeree were in a contractual relationship.
6. The promise was clear and unambiguous.
7. The offeree had acted equitably.
8. The promise released a debt or part of a debt that had already fallen due for payment.
9. The enforcement of the rights was resumed subsequent to the promise being made.
10. The offeror had given the offeree notice that he or she would resume enforcing the rights.
11. The offeree had not relied on the promise to such an extent that resuming the enforcement of the rights would be inequitable.
12. The behaviour of the offeror was such that it led the offeree to believe something.
13. The offeree acted on that belief to his or her detriment.
14. The behaviour of the offeror was unconscionable.

<u>Cases:</u>

2 → 1: *Hughes v Metropolitan Railway Co., Central London Property Trust Ltd v High Trees House Ltd*

3 → 2: *Coombe v Coombe, McCathie v McCathie*

4 → 2: *Commissioner of Inland Revenue v Morris, P v P*

5 → 2: *McCathie v McCathie*

6 → 2: *John Burrows Ltd v Subsurface Surveys Ltd*

7 → 2: *Re Goile, D and C Builders Ltd v Rees*

8 ⊣ 2: *Homeguard Products Ltd v Kiwi Packaging Ltd, Foakes v Beer*

10 → 9: *Tool Metal Manufacturing Co. Ltd v Tungsten Electric Co. Ltd*

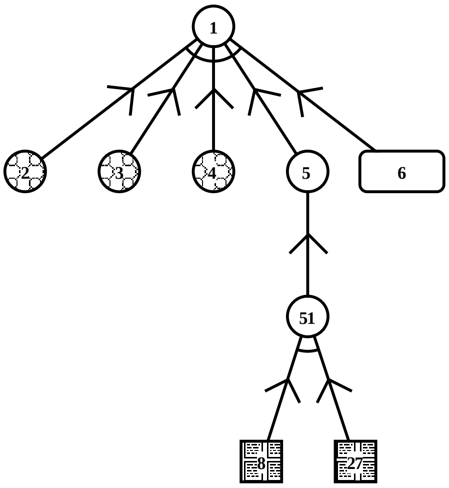
11 → 9: *Emmanuel Ayodeji Ajayi v RT Briscoe (Nigeria) Ltd*

12 → 2: *Waltons Stores (Interstate) Ltd v Maher, McDonald v Attorney General*

Figure A.22: The argument packet for promissory estoppel.

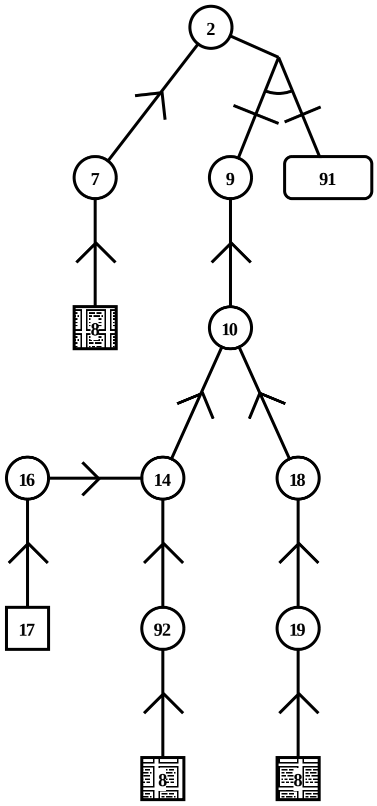
Appendix B

The Argument Structure for Henderson v BNZ



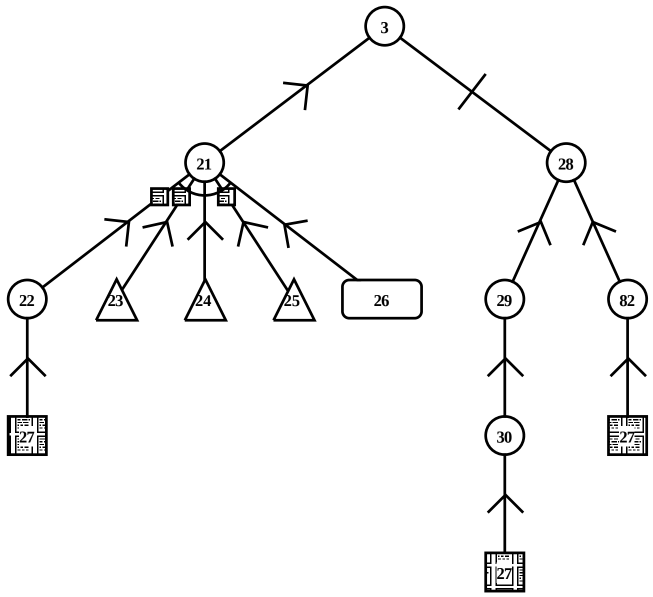
1. A valid contract exists between the offeror (BNZ) and the offeree (Mr Henderson).
2. The offeror (BNZ) made an offer to the offeree (Mr Henderson) at time t1 (16 May 1988).
3. The offeree (Mr Henderson) accepted the offer at time t2 (21 June 1988).
4. Valid consideration was provided in exchange for the offer.
5. The offeror (BNZ) and offeree (Mr Henderson) intended for the agreement to have legal consequences.
6. If the offeror made an offer to the offeree at time t1 and the offeree accepted the offer at time t2 and valid consideration was provided in exchange for the offer and the offeror and offeree intended for the agreement to have legal consequences then a valid contract exists between the offeror and offeree.
8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, as presented to the court.
51. The wording of option three and the letter from Mrs Henderson's solicitor makes it clear that the parties intended the agreement to have legal consequences.

Figure B.1: The overall argument in *Henderson v BNZ*.



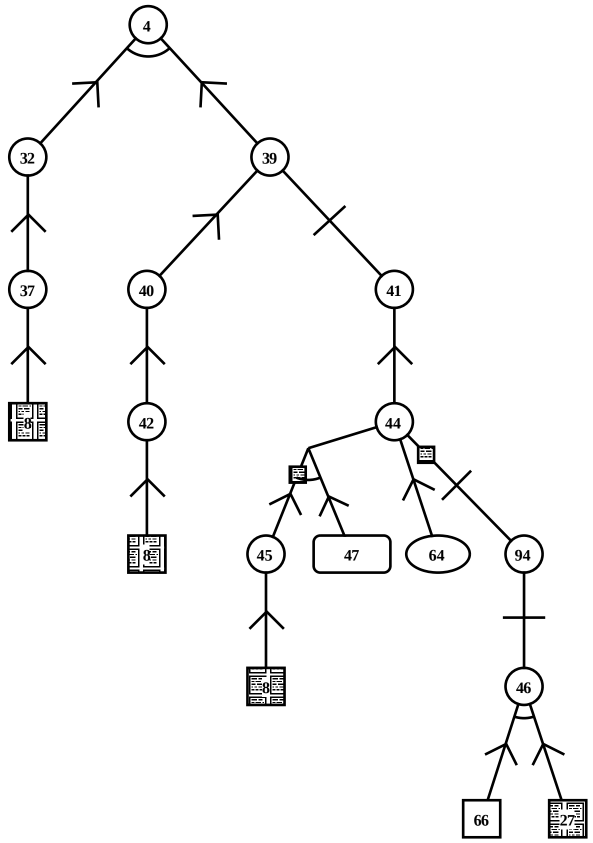
2. The offeror (BNZ) made an offer to the offeree (Mr Henderson) at time t1 (16 May 1988).
7. The offeror (BNZ) sent a letter containing an offer to the offeree (Mr Henderson) at time t2 (16 May 1988).
8. Letter specifying repayment options, as presented to the court.
9. The letter was merely an invitation to treat.
10. The option in question—the third option—was merely an invitation to treat.
14. The bank was only saying that it would consider any repayment sum that was proposed by Mrs Henderson.
16. The bank manager (Mr McAuley) indicated that an opportunity would be given to increase the repayment sum.
17. Testimony of the offeree (Mr Henderson).
18. The third option required someone other than the offeree (Mr Henderson) to take some action.
19. The third option required Mrs Henderson to "... make an offer ..."
91. An invitation to treat is not an offer.
92. The third option did not specify a definite repayment sum, but said that it would "consider" a reasonable figure.

Figure B.2: The subargument relating to offer in *Henderson v BNZ.*



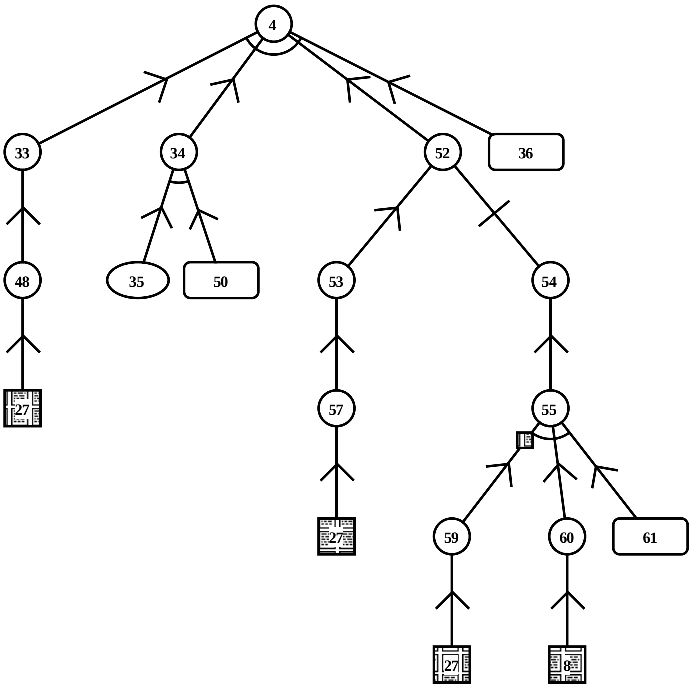
3. The offeree (Mr Henderson) accepted the offer at time t2 (21 June 1988).
21. Legal requirements for communication of acceptance were satisfied at time t2 (21 June 1988).
22. The offeree (Mr Henderson) posted a letter of acceptance at time t2 (21 June 1988).
23. The offeror (BNZ) did not specify a mode of communication other than via post.
24. The offeror (BNZ) and offeree (Mr Henderson) did not intend for there to be no agreement until the offeror received the letter of acceptance.
25. The letter of acceptance was properly stamped and addressed.
26. If the offeree posted a letter of acceptance at time t2 and the offeror did not specify a mode of communication other than via post and the offeror and offeree did not intend for there to be no agreement until the offeror received the letter of acceptance and the letter of acceptance was properly stamped and addressed, then legal requirements for communication of acceptance were satisfied at time t2.
27. Letter from Mrs Henderson's solicitor, presented to the court.
28. The "acceptance" was really a counter offer that destroyed the original offer.
29. The "acceptance" was phrased in such a way as to be inconsistent with the offer.
30. The solicitor's letter specified a repayment sum of $28,000—less than the $30,000 specified by the offeror.
82. The letter concluded "...If such an arrangement can be reached would you kindly advise."

Figure B.3: The subargument relating to acceptance in *Henderson v BNZ.*



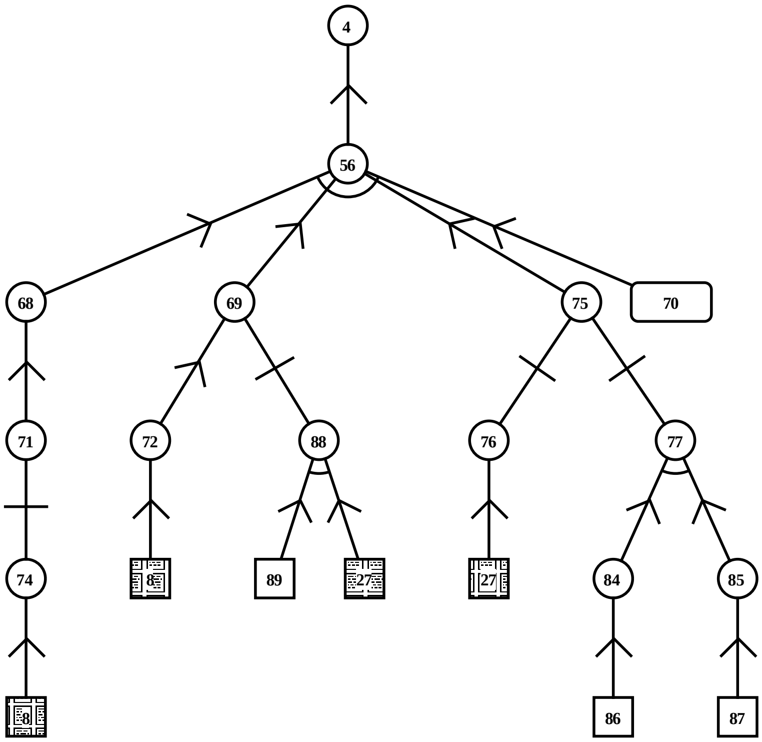
4. Valid consideration was provided in exchange for the offer.
8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, as presented to the court.
32. The offeree (Mr Henderson) provided the consideration.
37. Under the terms of the repayment options, the offeree (Mr Henderson) had to make repayments of $200 per week in the interim.
39. The consideration was adequate.
40. The consideration had some intrinsic value.
41. The consideration was insufficient.
42. The consideration had a monetary value.
44. The consideration was insufficient because the offeree (Mr Henderson) was already legally bound to provide it due to a previous contract with the offeror (BNZ).
45. The consideration was the payment of interest on a debt owed by the offeree (Mr Henderson) to the offeror (BNZ).
46. The offeree (Mr Henderson) was already making the interest payments and he made no fresh commitment to continue paying.
47. If the consideration was interest payments for an existing debt then the consideration was insufficient because the offeree was already legally bound to provide it due to a previous contract with the offeror.
64. Mr Henderson's obligation to pay the interest payments was a very small matter in relation to his obligation to pay the entire debt.
66. Testimony of the bank manager (Mr McAuley).
94. The consideration provided by the offeree (Mr Henderson) resulted in the offeror (BNZ) obtaining an additional benefit.

Figure B.4: The subargument relating to provision of consideration by Mr Henderson in *Henderson v BNZ.*



4. Valid consideration was provided in exchange for the offer.
8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, as presented to the court.
33. Consideration was provided by a third party (Mrs Henderson).
34. The offeree (Mr Henderson) was able to rely on the consideration provided by the third party (Mrs Henderson).
35. Time t2 (21 June 1988) is after 31 March 1983.
36. If consideration was provided by a third party and the offeree was able to rely on the consideration provided by the third party and the consideration was adequate then valid consideration was provided in exchange for the offer.
48. Under the terms of the agreement, the third party (Mrs Henderson) would make a payment of $28,000 to the offeror (BNZ).
50. The offeree is able to rely on consideration provided by a third party if the date of the agreement is after 31 March 1983.
52. The consideration provided by the third party (Mrs Henderson) was adequate.
53. The consideration had some intrinsic value.
54. The consideration was insufficient.
55. The consideration was insufficient because it was merely a repayment of existing debt owed by the offeree (Mr Henderson) to the offeror (BNZ).
57. The consideration had a monetary value of $28,000.
59. The consideration was the part payment of a debt.
60. The offer was the forbearance of the balance of the debt.
61. If the consideration was the part payment of a debt and the offer was the forbearance of the balance of the debt then the consideration was insufficient because it was merely a repayment of existing debt owed by the offeree to the offeror.

Figure B.5: The subargument relating to provision of consideration by Mrs Henderson in *Henderson v BNZ*.



4. Valid consideration was provided in exchange for the offer.
8. Letter specifying repayment options, as presented to the court.
27. Letter from Mrs Henderson's solicitor, presented to the court.
56. Consideration was provided under the doctrine of promissory estoppel.
68. The behaviour of the offeror (BNZ) was such that it led the offeree (Mr Henderson) to believe something.
69. The offeree (Mr Henderson) acted on that belief to his or her detriment.
70. If the behaviour of the offeror was such that it led the offeree to believe something and the offeree acted on that belief to his or her detriment and the behaviour of the offeror was unconscionable, then consideration was provided under the doctrine of promissory estoppel.
71. The bank represented to the offeree that if Mrs Henderson went to the trouble of making an offer of $30,000, it would accept the offer as repayment of his entire debt.
72. Mr Henderson was making $200 weekly payments under the terms of the agreement.
74. The most the bank could be said to be representing was that it would "consider" any offer by Mrs Henderson.
75. It was unconscionable for the bank to reject Mrs Henderson's offer after her incurring expense in relying on their promise.
76. Mrs Henderson offered less than the specified sum of $30,000.
77. The bank's representation of option three was being taken to mean something quite different than what was intended.
84. The District Office intended that the bank would still receive the proceeds of the mortgagee sale.
85. The offeree was aware that any offer would have to be agreed upon by the District Office.
86. Testimony of District Office staff.
87. Testimony of the offeree (Mr Henderson).
88. Mr Henderson was already making the payments and made no fresh commitment to do so.
89. Testimony of the bank manager (Mr McAuley).

Figure B.6: The subargument relating to provision of consideration under the rule of estoppel in *Henderson v BNZ*.