Foundations of Argumentation Technology
Summary of Habilitation Thesis

Thomas F. Gordon
Technische Universität Berlin
Fakultät IV – Informatik und Elektrotechnik
Franklinstr. 28/29
D-10587 Berlin, Germany

Abstract

This cumulative habilitation thesis summarizes sixteen scientific publications, written over a period of more than 20 years, on the topic of computational models of argument and argumentation technology. Most of this research has been driven by an analysis of task requirements in the legal domain, with the aim of developing software tools which help people to make and justify legal decisions in a more efficient, transparent and just way.

One of the results of this research is a high-level model of argumentation tasks (use cases) and their relationships, which has proven useful as a research roadmap in recent years. The tasks of this model are grouped into logical, dialectical and rhetorical layers. The logical layer, which encompasses classical logic but is broader, consists of tasks for modeling knowledge and argumentation schemes and using these to construct arguments. The dialectical layer consists of tasks for conducting and regulating dialogues in which arguments are exchanged to resolve issues and make decisions, depending on the type of dialogue. Methods for evaluating arguments to determine the acceptability of claims, by using proof burdens and standards to aggregate competing pro and con arguments, are at this level. Each type of dialogue has its own procedural rules (protocol). Finally, the rhetorical layer consists of tasks for effectively participating in dialogues, including the selection of argument moves (speech acts) and the presentation of arguments and justification of decisions in forms which are clear and understandable, taking into consideration the particular audience. Methods for diagramming or mapping arguments can be useful at this level.

From the beginning, this research has been founded on premises derived from the field of argumentation in philosophy and legal theory:

- Argumentation is a synthetic, modeling activity, in which theories and arguments are constructed together in a purposeful, goal-driven way.
- The space of potential arguments in realistic application domains is not only infinite but not even recursively enumerable. No procedure for generating all possible arguments systematically exists, typically.
- Although classical logic sets the standard for normative models of valid inference, about the necessary consequences of propositions and what it means for propositions to be inconsistent, further normative models are needed for practical reasoning and decision-making in the face of incomplete or inconsistent information and limited resources for problem solving.
- For practical reasoning it is useful to represent knowledge as defeasible rules with exceptions, where these rules represent not what is probably true in the domain, empirically, but rather encode heuristics about how to solve problems and make decisions in the domain, so as to minimize risks of errors.
- Argumentation typically takes places in dialogues. Procedural norms regulate these dialogues, including the allocation of proof burdens and standards. Which claimed propositions are acceptable at the end of a dialogue depends not only relations between these claims and

assumptions about the rules and facts of the domain, but also on the *actual* moves which were made by the parties during the dialogue, in accordance with these procedural rules.

A series of prototype argumentation support systems has been developed during the course of this research: Oblog, the Argument Construction Set (ACS), the Pleadings Game, Zeno and, most recently, Carneades. All of these systems are represented in the submitted papers and put into relationship with one another, showing how they fit into a larger research program.

Contents

Acknowledgments

The work reported here was conducted in collaboration with many colleagues over the years. I would like to take this opportunity to thank Katie Atkinson, Stephan Ballnat, Trevor Bench-Capon, Trevor Bench-Capon, Thomas Christaller, Gerhard Brewka, Anne Gardner, Olivier Glassey, Matthias Grabmair, Carole Hafner, Joachim Hertzberg, Rinke Hoekstra, Nikos Karacapilidis, Marc Lauritsen, Ann Macintosh, Oliver Märker, Thorne McCarty, Jonas Pattberg, Henry Prakken, Gernot Richter, Gerald Quirchmayr, Giovanni Sartor, Torsten Schaub, Barbara Schmidt-Belz, Gerd Schürmann, Erich Schweighofer, Volker Tschammer, Angi Voss, Hans Voss, Doug Walton, and Adam Wyner.

I will always to be grateful to Herbert Fiedler, for his support at the very beginning of my career, at the GMD, and would also like to mention Bernard Schlink, for his having introduced me to Herbert Fiedler.

Wolfgang Bibel has always been a source of inspiration to me. I would like to thank him for his continual encouragement and support, both during the writing of my doctoral thesis and up until the present day.

Bernd Mahr deserves special mention for his supervision and support of this habilitation thesis and for giving me this opportunity. It has been a pleasure working with him.

Last but not least, I would like to thank my wife, Ines, our two children Dustin and Caroline, and my dear my mother, Gloria. Without their love and support I wouldn't have appreciated the true value of my professional accomplishments.

Chapter 1

Introduction

Argumentation is the process of critically analyzing claims, solving practical problems and making decisions by constructing, aggregating and evaluating arguments pro and con statements at issue, typically in the context of dialogues. Argumentation is studied in several fields, including philosophy, law, communications studies, informatics and computer science. Within computer science, argumentation is usually studied as an artificial intelligence topic.

I use the term "argumentation technology" in a way analogous to "information technology". Just as information technology (IT), now often called information and communications technology (ICT), is technology for storing, processing, and communicating information, argumentation technology (AT) is technology for supporting tasks in argumentation processes and dialogues, such as constructing, aggregating, and evaluating arguments, visualizing dependencies among arguments, managing commitments and applying dialogue protocols.

Argumentation technology is developed for use in decision-support systems (DSS) [163], which are broadly defined as any kind of (computer) system for supporting decision-making activities, both by individuals and within groups and organizations. There are many kinds of decision-support systems [102], using full text information retrieval, relational databases, spreadsheets, rule-based systems and other technologies. Argumentation technology is orthogonal to such technologies, since the entities of argumentation systems, such as evidence, arguments, and dialogues, can be represented in many ways using these and other technologies. The methods, processes and tools of argumentation technology are specified at another layer of abstraction and mapped to textual, relational or other representations when constructing decision-support systems.

Argumentation technology uses and builds upon information technology. The concepts of information and communication are broad enough to cover all the elements of argumentation processes, such as the assertion of a claim, the asking of a question or the putting forward of an argument. One issue is whether argumentation technology is an inherent part of the field of informatics, or rather an application domain for informatics. Informatics is a scientific field which studies natural and artificial systems which store, process and communicate information, as well as an engineering field which develops and evaluates information technology.[1] Applied informatics is the subfield of informatics which uses information technology to develop information systems and tools, such as domain-specific programming languages, for other fields, such as medicine, business administration, or government, in interdisciplinary cooperation with experts from these fields. If argumentation

[1]Informatics and computer science are closely related, if not identical, fields. Computer science may be a somewhat narrower field, as it is focused on theories, methods and technology for computing mathematical functions, and studies information processing mainly to extent it can be understood as computation. In principal, the fields of informatics and computer science do not depend on electronics. In practice, however, informatics and computer science both of course make extensive use of electronics, including telecommunications networks, and at some universities the study of electronic computing machinery is considered to be a part of the field of informatics, called technical informatics, or the departments for electronics and informatics have been merged.

is considered to be mainly a field outside of informatics, such as philosophy, then argumentation technology might be viewed as part of the field applied informatics. Arguably, however, argumentation technology is best viewed as fully within the scope of the field of informatics, rather than an application field. Informatics has always shared with philosophy, as part of its core subject matter, an interest in the normative study of rational, systematic methods for thinking and problem solving. Logic, including formal and mathematical logic, has always been one of the core subjects of philosophy, before it also became part of computer science, via applied mathematics. In the 1950s, the scope of the topic of what we now call informatics was expanded to include research on artificial intelligence, with the goal of modeling, simulating and even constructing intelligent artifacts. The modern view of artificial intelligence [149] recognizes the social, communicative dimensions of intelligence and aims to model not only individuals but also groups of deliberating and negotiating *intelligent agents*. And in the early 1960s, Carl Adam Petri in his dissertation [128] argued for a paradigm shift in the way we view computers, away from the computation, inference and artificial intelligence views of computers towards a view which emphasized the potential of computers for realizing a distributed infrastructure, not just a medium, providing comprehensive support for communication processes:

> Now is the time to shift our view of computers from communications medium to negotiation medium, from knowledge processing to interest processing. *Carl Adam Petri, 1962 [128]*

Thus, I take the view that argumentation should be considered a core topic of theoretical and practical informatics, rather than a application domain. Argumentation technology builds upon research on formal logic, automatic theorem proving and logic programming and extends the scope of formal, normative models of reasoning and decision-making to cover a much broader class of practical problems. Here are some examples of the kind of practical problems, small and large, we want to support with argumentation technology:

1. Deciding whether or not to buy a new car and if so, which make and model.
2. Designing a domain-specific programming language.
3. Deciding whether or not a citizen is entitled to some social service, such as housing support or unemployment benefits.
4. Deciding whether or not to publish a cartoon depicting Mohammed, after threats to cartoonists in Denmark.
5. Deciding how to reform the tax system, possibly by eliminating income taxes and relying solely on value-added taxes.
6. Deciding when and how to return troops home from the Iraq war.
7. Deciding how to reduce greenhouse gases.

What these problems have in common, is that they are all poorly-defined, or what Rittel and Weber called 'wicked' problems [144]. There is no definitive formulation of the problem. There is no stopping rule, i.e. no criteria for deciding whether or not a solution has been found. Solutions are not true or false, but only better or worse. Each problem is essentially unique, so it is difficult to learn from experience. The solution space is typically not recursively enumerable. In [69] I identified some additional features of such problems. There is usually both too much and not enough information available. The resources which may be expended in constructing a solution, such as time and money, are limited. The expected value of the solution is not high enough to warrant the development of computer model or *knowledge-base* of the problem space. Opinions differ about the truth or relevance of the available information. Arguments can be made both pro and con each issue. Inferences are defeasible; further information may require the retraction of provisional conclusions. Value judgments are at least as important as facts or knowledge. Multiple parties, with conflicting interests, are affected. Conflicts of interests are inevitable. No solution is likely to be optimal for all interested parties.

Practical problems of this kind typically occur in the context of what Carl Hewitt has called 'open systems' [100]. Open systems consist of numerous, distributed components with information coming from many sources, processed concurrently. Information may enter the system at any time. Control and decision-making are decentralized. Information coming into the system from the outside, or from different parts of the system, may be inconsistent. The information available to one part of the system may be unknown or unavailable to other parts of the system, for example to protect privacy. Open systems require methods for "coping with conflicting, inconsistent and partial information" [100, p. 271]. Procedural norms for decision-making dialogues ("due process") are needed to allow for "the consideration of multiple inconsistent microtheories" [100, p. 285.].

Existing information technologies provide poor support for solving such problems. Algorithms, i.e. effective procedures, are by definition only available for decidable problems. Conversely, a problem is decidable only if there is an effective procedure for determining membership in the set of solutions. Moreover, the correctness of the output of an algorithm depends on the correctness of its input: garbage in, garbage-out. Heuristic search and automated theorem proving methods are somewhat more general, but work only for semi-decidable problems, i.e. well-defined problems for which it is possible to systematically generate and test all possible solutions [149, pg. 62] Databases can provide efficient retrieval of large amounts of information, but the correctness and completeness of the data can be problematic, depending on the quality and trustworthiness of the organizational processes which provide and maintain the content of the database. This problem is exacerbated in the context of distributed databases, such as the World Wide Web. Knowledge-based systems typically require the poorly-defined problem to first be converted into a well-defined problem [31]. This conversion process is possible only by making strong assumptions which may not hold in the case at hand. Moreover, knowledge-base systems suffer from the problem of the knowledge-acquisition bottleneck and thus are typically not cost-effective for one-of-a-kind problems, where the development costs of the knowledge-base cannot be distributed for over many problems.

1.1 Argumentation in Philosophy

My goal is to develop a computational model of argumentation which can serve as a foundation for decision-support systems. It is important for this work to be informed by the state-of-the-art in argumentation theory in the field of philosophy, to avoid reinventing the wheel, take advantage of the insights and results from philosophers who have been working on the topic for centuries, and to avoid using terms in ways inconsistent with their meaning in philosophy, at least not without good reason.

The modern study of argumentation probably began with Toulmin's "The Uses of Argument" in 1958 [161] and Pereleman and Olbrechts-Tyteca's "The New Rhetoric" [127] . Toulmin was one of the first philosophers in the twentieth century to reflect on the limitations of mathematical logic as a model of rationality in the context of everyday discourse and practical problems. By the 1950s, logic had become more or less synonymous with mathematical logic, as invented by Boole, De Morgan, Pierce, Frege, Hilbert and others, starting in the middle of the nineteenth century. Interestingly, Toulmin proposed legal argumentation as a model for practical reasoning, claiming that normative models of practical reasoning should be measured by the ideals of jurisprudence. The logic of practical reasoning, he wrote is "generalized jurisprudence" [161, p. 7].

By the 1970s, work in philosophy and related disciplines on normative models of natural language argumentation, as it occurs in political debates, the media and legal disputes, developed into the field of argumentation, also known as 'informal logic' [164, 165, 89, 173]. The 'informal' in 'informal logic' is not intended to preclude formal or mathematical models of argumentation. Rather, the name reflects the goal of the field to extend the scope of the field of logic beyond deductive and inductive arguments, where the inferences sanctioned by the logic depends only on the logical form of the premises and conclusion, to also cover conventional inference patterns in natural language texts, where the conclusions sanctioned by the patterns, called 'argumentation schemes' [173] are only presumptively true, rather than necessarily true, and depend on interpreting the meaning of terms and predicates used in their premises, situated in the context of the particular dialogue, and not just their form.

The ancient Greek philosophers, in particular Aristotle, distinguished at least three normative sciences: logic, rhetoric and dialectic. Logic is the study of forms of inference, exemplified by Aristotle's theory of syllogism, which served as the main model of deductive logic until propositional logic was invented in the 19th century. Rhetoric is the study of the art of persuasion. Rhetoric is also a normative field of study, because it develops models of effective communication, which serve as normative standards for evaluating the quality of actual speeches and texts. And dialectic is the study of dialogues in which arguments and counter-arguments are exchanged to resolve disputes or controversies. The core topics studied by dialectic are various kinds of opposition and conflict and rational methods for resolving these conflicts using argumentation. Three kinds of opposition studied by dialectic are:

1. Opposing arguments — various forms of attack relations among arguments, such as arguments pro and con some statement, now often called *rebuttals*.

2. Opposing interests — conflicts among the interests of stakeholders affected by the outcome of the dispute or the solution of the problem.

3. Opposing ideas — conflicts among the alternative proposals for resolving the conflict or solving the problem, when they are logically inconsistent or it is physically impossible to implement or execute both of them.

Logic and rhetoric survived, but dialectic was largely forgotten until Kant and, especially, Hegel, in the 18th and 19th centuries. Hegel developed a dialectical method for resolving conflicts between two conflicting ideas, the 'thesis' and the 'antithesis', where the solution integrates aspects of both to produce a 'synthesis'. Marx, presumably influenced by Hegel, made dialectic a central part of his philosophy of 'dialectical materialism', where it referred to the class struggle. Thus, Marx's interpretation of dialectic focused on conflicts of interest. Perhaps due in some part to Marx, dialectic since his time has been viewed with great skepticism and has had some vocal critics, such as Karl Popper, especially in the United States and Great Britain, where it is viewed as being incompatible with the still dominant philosophy of positivism, which in computer science implicitly serves as the philosophical foundation for most work in artificial intelligence, knowledge engineering and, most recently, the so-called 'semantic web' [22].

Perhaps because dialectic is somewhat controversial, 'informal logic' and 'argumentation' are now usually used to refer to the subject. But the subject matter of the field of argumentation is broader, covering all of logic, dialectic and rhetoric, as a normative science for all aspects of practical reasoning, conflict resolution and problem-solving in dialogues.

One of the leading contemporary philosophers of argumentation is Douglas Walton. In recent years, my efforts to construct computational models of argument has been strongly influenced by Walton's work, in particular his "The New Dialectic" [169]. Walton's recent textbook, "Fundamentals of Critical Argumentation" [173], is a good starting point for getting an overview of the modern philosophy of argumentation. Since Walton's philosophy underpins much of my work, its main elements are summarized in the rest of this section. Let us begin with the fundamental concept of an argument. An argument links a set of statements, the premises, to another statement, the conclusion. The premises may be labelled with additional information, about their role in the argument. Aristotle's theory of syllogism, for example, distinguished major premises from minor premises. The basic idea is that the premises provide some kind of support for the conclusion. If the premises are accepted, then the argument, if it is a good one, lends some weight to the conclusion.

The goal of argumentation is often described as discovering or determining the truth of some claim, where a claim is a statement which has been asserted by some party in the dialogue. When the claim is about a factual or theoretical issue, this may make sense, at least as an ideal. However, when the issue being discussed is about what action to take in order to solve some practical problem, this characterization of the goal of argumentation is more problematical. If, for example, the plan of a city to build the airport is being subjected to public review, one would not ordinarily characterize this as being an issue of truth or falsity. The question is not whether the plan is true, but whether it is good, acceptable or well-advised.

For this reason, the goal of argumentation is to determine the acceptability of claims, rather than their truth. In the case of factual claims, ideally only true claims would be acceptable. Given complete information and unlimited reasoning resources, argumentation should conclude that a factual statement is acceptable if and only if it is true. But in practice, resources will typically be limited and we will often have to decide whether or not to accept claims with less than complete certainty about their truth. Consider criminal cases, to take a familiar example, where a person can be convicted of having committed a crime when the evidence is conclusive "beyond reasonable doubt". Although this is a high standard of proof, it does not require complete certainty.

Arguments provide reasons for accepting their conclusion; the conclusion need not be a logical consequence of the premises. Logical consequences are necessary, by virtue of their form, irrespective of their content. Arguments, in contrast, are substantive and 'defeasible'. They are substantive because they depend not only on the form of the premises, but also their content and acceptability. And they are defeasible because their conclusions are only plausible or presumptive, not certain, and may be defeated in various ways by additional information, for example by revealing implicit premises which turn out to be untenable or by bringing forward better counterarguments.[2]

As just suggested, some premises of arguments may be implicit. For the sake of efficiency, the norms of argumentation do not require all premises to be made explicit, at least not immediately. For example, premises which are thought to be common knowledge, or otherwise already accepted by the other participants, are typically left implicit. "Socrates is a man, therefore Socrates is mortal" to use a standard example, is a perfectly understandable argument, even though the major premise "All men are mortal." has been omitted. Implicit premises can be revealed and possibly challenged during the dialogue as necessary.

There are many different kinds of arguments and much research has gone into discovering and classifying various patterns of argument, based on an analysis of the structure and content of arguments reconstructed from natural language texts. These patterns of argument have come to be called "argumentation schemes". Although they are the result of empirical research, they also have a normative side. They are a useful tool both for guiding the reconstruction of arguments put forward by others, after the fact, so as to open them up to critical analysis and evaluation, as well supporting the construction or invention of new arguments to put forward in a dialogue.

Argumentation schemes generalize the concept of an inference rule to cover plausible as well as deductive and inductive forms of argument. Argumentation schemes are conventional patterns of argument, historically rooted in Aristotle's "Topics" [158]. Unlike inference rules, argumentation schemes may be domain dependent. Each scheme comes with a set of "critical questions" for evaluating and challenging arguments which use the scheme. For example, the scheme for argument from expert opinion includes a critical question about whether the expert is biased. Argumenta-

[2]The concept of defeasibility in argumentation was imported from the field of law, where is means [26] being "subject to being defeated, annulled, revoked or undone upon the happening of a future event or the performance of a condition subsequent, or by a conditional limitation." In the field of artificial intelligence, this property of argumentation is known as 'nonmonotonicity', a term borrowed from mathematics. In calculus, a function f is monotonic if and only if $f(x) > f(y)$ when $x > y$. In logic, a consequence relation, $\models$, is monotonic if and only $C_{\models}(\Gamma) \supset C_{\models}(\Delta)$ if $\Delta \subset \Gamma$, where $C_{\models}$ is the deductive closure of a set of formulas given the $\models$ consequence relation.

tion schemes are useful for several purposes, including reconstructing and classifying arguments, criticizing arguments, and as templates for making new arguments.

Since argumentation schemes may be domain dependent, there are an unlimited number of such schemes. Domain dependent schemes, in fields such as the law, may evolve along with the knowledge of some domain. Some schemes, however, are general purpose. Walton and his colleagues have taken on the project of collecting and classifying general purpose schemes. To date their collection contains about 60 schemes. Examples include Argument from Expert Opinion, mentioned previously, Argument from Popular Opinion, Argument from Analogy, Argument from Correlation to Cause, Argument from Consequence, Argument from Sign and Argument from Verbal Classification.

When evaluating arguments put forth in a dialogue, one issue is the 'validity' of the argument. An invalid argument has no weight, i.e. provides no support for its conclusion. But how shall validity be defined? In classical deductive logic, an inference is valid if and only if the conclusion is necessarily true if the premises are true. This conception of validity is too stringent for arguments, since in general these only provide plausible support for their conclusions. Nonmonotonic logics strengthen the consequence relation to support consequences which are only plausible. Consequences in nonmonotonic logics are defeasible: it may be that some consequence of a set of premises is not a consequence of some superset of these premises. That is, additional information may require plausible conclusions to have to be retracted.

Nonmonotonic logics retain however the relational approach of argument validity of classical logic: whether or not an argument is valid depends only on the relationship between the set of premises and the conclusion. Walton's theory of argumentation, however, takes a more contextual, procedural view of argument validity: an argument is valid if and only it furthers the goals of the dialogue in which it is put forward. From this perspective, the validity of an argument can depend on the state and history of the dialogue. To give a practical example: an argument in favor of some proposal made during the brainstorming phase of a deliberation might be valid during the process of selecting some of these brainstorming ideas for a more in-depth evaluation in the next phase of the deliberation, but not valid in this later phase if the particular proposal had not been selected. To sum up: from a dialectical perspective, whether or not an argument is valid depends on how and when it is used in a dialogue, not merely on the relation between its premises and conclusion.

Whether or not an argument has been used properly or furthers the goals of the dialogue, depends on the type of dialogue. Walton has developed a taxonomy of dialogue types, as illustrated in Figure 1.1.

Persuasion dialogues debate the truth of some statement. One party, the proponent, claims that some statement is true. The other party, called the respondent, challenges this claim. There are several subtypes of persuasion dialogues. In a 'dispute', the respondent not only challenges the proponent's claim, but also claims some opposing, contradictory statement to be true. The roles in a dispute are symmetric. The proponent and respondent each have a burden of proof, for their respective claims. More common, however, is the 'dissent' form of persuasion dialogue, in which the respondent only doubts the proponent's claim, but makes no claim of his own. In a dissent, the proponent has the burden of proof and must produce the stronger arguments. The arguments of the respondent need only be strong enough to cast doubt on the proponent's claim.

Although the dialogue types are usually described as involving two parties, they can be generalized to any number of parties. More important than the number of participants is their roles in the dialogue. Several participants could share a role.

An information seeking dialogue has the goal of seeking advice. The starting point is not the assertion of some claim, as in persuasion dialogue, but rather the asking of a question. Expert consultations, for example with medical doctors or lawyers, are a subtype of information seeking dialogues.

The goal of negotiation dialogues is to make a 'deal', i.e. to reach an agreement on how to exchange such things as goods, services or money. The starting point is neither a question nor a

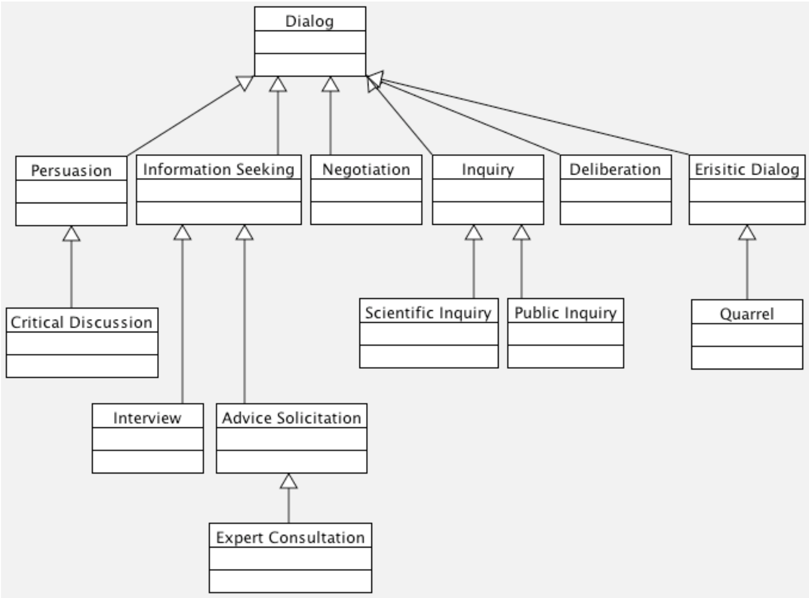

Figure 1.1: Taxonomy of Dialogue Types

claim, but rather an offer. This can be accepted by the other party or modified in a counteroffer.

An inquiry is a methodical investigation of some matter, to explain or understand some observations or data. Scientific inquiries try to explain natural phenomena by developing hypotheses and constructing, evaluating and comparing scientific theories. Public inquiries investigate such things as accidents or crimes. The starting point of an inquiry consists of the observations in need of explanation. These observations are not being called into question, unlike the claim of a persuasion dialogue. The question is not whether these observations are true, but how best to explain them.

Deliberation dialogues are about choosing some course of action which takes into account the interests of multiple stakeholders. In a deliberation, one of the first tasks is to identify the stakeholders and their interests. They may not all be participants in the dialogue, at least not initially. And it may not be practical for every stakeholder to take part in the dialogue personally. Stakeholders may need to be represented by others. A common mistake in deliberation is for participants to make and try to defend specific proposals at too early a stage in the dialogue. It is usually better to first spend time trying to identify the stakeholders and understand their interests. Brainstorming may come next, in which ideas are freely collected but participants are not supposed to commit themselves yet to particular proposals.

So-called 'eristic' dialogues, from the ancient Greek word meaning wrangle or strife, is an emotional kind of dialogue in which the participants vent their anger, frustration or other deep feelings. Eristic dialogues are considered by some to be irrational and to have no other goal than to "argue for the sake of argument". Walton's view, however, is that such dialogues can serve a positive, 'cathartic' function and that they are, like the other kinds of dialogues in his typology, guided by norms, even if these norms are quite relaxed compared to the other dialogue types. For example, the basic civility norms requiring participants to do such things as take turns and give each other a fair opportunity to express their views, remain in force.

Actual dialogues may be mixtures of these various types and a dialogue may shift from one type to the other and back. For example, during a negotiation a salesman may make some claims about the product that might be called into question by the customer, causing a temporary shift to a persuasion dialogue. Similarly, in a deliberation, once the stage has been reached to evaluate specific proposals, each such evaluation could take the form of a persuasion dialogue.

Dialogue types are defined along several dimensions: the purpose or goal of the dialogue, the roles of the participants, the speech acts available, the termination criteria, a process model and a 'protocol' for regulating this process. Dialogue types in argumentation theory are normative models of communication. If argumentation dialogues are viewed as games, then the participants are its players, the speech acts its moves, and the protocol defines it rules.

Speech acts are uses of natural language in dialogues, such as asking questions, making claims, putting forward arguments or counterarguments, making concessions or retracting claims. The protocol defines the pre- and postconditions of these speech acts, to regulate when a speech act may be made and, if it is allowed, with what effect. This may depend on the stage of the process and the state of the dialogue, taking into consideration the prior history of the dialogue, i.e. what has already been said.

In addition to defining the preconditions and postconditions of speech acts, the protocol will include rules regulating such things as termination conditions (When is the dialogue finished?), commitments rules (When does a party become committed to some statement?), proof standards (How are the arguments pro and con some statement to be balanced, weighed or otherwise aggregated for each issue?), and finally the distribution of the "burden of proof". There are various kinds of proof burdens to consider: the "burden of questioning" regulates whether some statement can be assumed to be true so long as it has not been called into question; the "burden of production" regulates which party is responsible for producing arguments or evidence suggesting that some presumption may not hold; and the "burden of persuasion" regulates which party must have the stronger arguments when the time comes to make a decision. Usually the same party will have both the burden of production and the burden of persuasion. But this is not always the case. In criminal law, for example, the defense has the burden of production for any exceptions to crimes, such as self-defense in murder cases, but the prosecution has the burden of persuasion, even for such exceptions. Thus, to continue with the murder example, the prosecution has the burden of persuading the court that the killing was not done in self-defense, once the defendant has produced sufficient evidence to meet his burden of production.

1.2 Argumentation Tasks

Models, also computational ones, are abstractions designed for particular tasks. Details of the domain irrelevant for the task are abstracted away in the model. Computational models of argumentation have been developed for various argumentation tasks, so it will be useful for classifying these models and understanding their strengths and limitations to first try to identify and structure the main tasks or, to use the software engineering term [52], 'use cases', of argumentation. Figure 1.2 summarizes the results of my analysis of these use cases.

This use-case analysis is based on my prior work with Brewka [34], but also informed by subsequent work by Prakken, Bench-Capon and others [133, 17]. I now distinguish the following three layers, inspired by classical Greek philosophy:

- The logical layer is responsible for constructing or generating arguments by applying inference rules and argumentation schemes to axioms from models of evidence, rules, concepts, cases and other kinds of "knowledge bases". Classical logic sets the universally accepted standard for valid inferences about the *necessary* consequences of a set of propositions, including what it means for a set of propositions to be inconsistent. Argumentation schemes extend the universally valid formal inference rules of classical logic with domain-specific and contextual rules for deriving weaker conclusions, which are only presumptively true.

- The dialectical layer is responsible for structuring, evaluating, comparing and aggregating arguments which have been put forward during a dialogue, and informing participants about

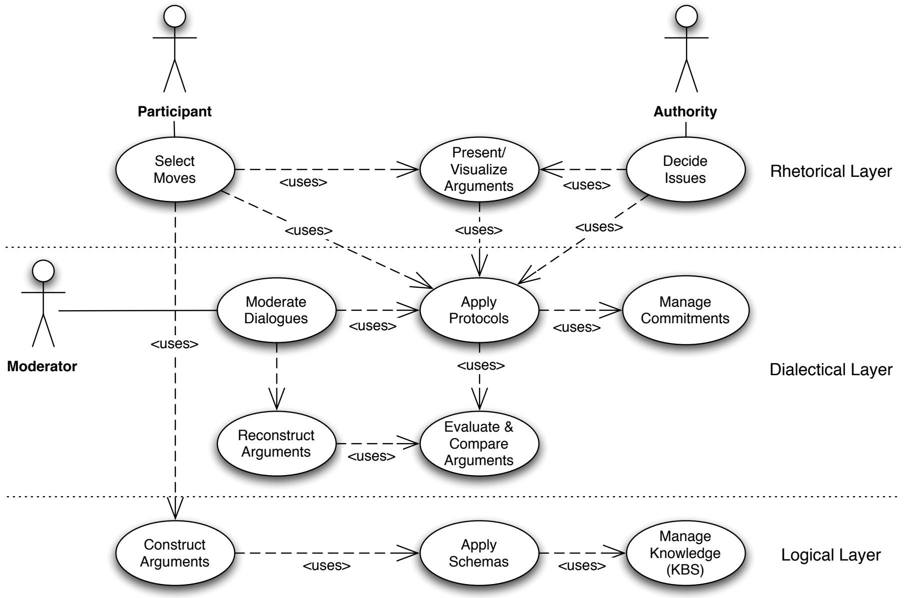

Figure 1.2: Argumentation Use Cases

the acceptability of statements and validity of arguments at each stage of the dialogue. I also include at this layer the task of 'reconstructing' arguments by interpreting natural language texts. The dialectical layer is also responsible for regulating and supporting the process of argumentation, guiding and facilitating the dialogue, to help assure it achieves its normative goals. The facilitation tasks of moderators and mediators are included here. One of these tasks is to help participants to obey procedural rules, i.e. the argumentation protocol for the applicable dialogue type. This task in turn requires keeping track of the commitments of the participants in the dialogue, from their claims, assertions and retractions.

- Finally, the rhetorical layer is responsible for helping participants to 'play the game' well. Whereas the dialectical layer supports the normative goals of the dialogue as a whole, this layer provides a private advisor to each participant, to help them protect and further their own interests. Tasks here include selecting among arguments which could be made and presenting these arguments clearly and persuasively, taking into consideration the intended audience, perhaps using argument visualization techniques. The decision-making task of authorities has also been put in this layer. This task includes the creation of a sufficient justification of the decision from the arguments which have been put forward.

The actors in this use-case diagram are roles, not individuals. In particular dialogues, one person may have more than one role, some roles may be combined or some roles may need to be distinguished further. For example, in lawsuits a judge may have the moderator role during the trial but share the authority role with a jury.

Philosophers may disagree about where to draw the borders between logic, dialectic and rhetoric. Some may prefer to move one use case or the other to some other layer in my diagram. I do not make strong claims about the differences between logic, dialectic and rhetoric. Although inspired by Greek philosophy, it is not necessary for the purposes of this study to try to perform a scholarly or historical analysis of how the meanings of there terms have been used in philosophy. My goal here has been only to try to identify common argumentation tasks and their dependencies, by performing a use-case analysis, applying software engineering methods.

1.3 Application Scenarios

Argumentation technology is potentially useful in any domain requiring solutions to practical problems to be found, justified or explained, especially when multiple parties with conflicting interests or opinions are involved.[3] Here I just want to provide a brief overview of some applications, to help demonstrate the practical relevance of research on argumentation technology which motivates my work on this topic.

Argumentation technology is useful for capturing and utilizing "design rationales", i.e. a representation of the reasoning behind the design of an artifact. There has been considerable research on capturing design rationales in the fields of software engineering [104] and human-computer interaction (HCI) [154]. Design rationales can be useful for several tasks, such as facilitating maintenance by helping to identify design changes required as requirements or assumptions change, or helping to maintain consistency of designs, by keeping track of dependencies among components.

In computer science, research on multi-agents systems conceives of distributed systems as intelligent, communication agents, which collaborate by negotiating and deliberating [145]. Some application scenarios for multi-agent have included load balancing of power requirements in electrical grid networks, routing messages in telecommunications networks, autonomous interactions between personal digital assistants (PDAs), for example to schedule meetings, and e-commerce applications, such as the composition of web services to perform complex tasks, including the negotiation of contracts for these services.

One of the earliest applications envisioned for argumentation technology was in the field of urban planning. Rittel, who invented the concept of 'wicked problems' and promoted the use of argumentation as the preferred way to address these problems [144], was an urban planning professor. More recently, Rittel's ideas have been adapted to help citizens to participate in urban planning, providing feedback to professional planners, using the World Wide Web. This idea has been generalized to other kinds of issues where the increased participation of citizens in planning processes promises to improve the quality and acceptability of decisions, under the banner of 'e-participation', as a part of 'e-democracy' [119].

Related to e-participation and e-democracy is the idea of using information and communications technology to improve *governance* in both the private ('corporate governance') and public sectors. The basic idea of governance is to 'steer' an organization with the aim of articulating and achieving its goals. That is, governance is a broad concept covering many forms of management, leadership and administration. I have developed a model of governance [70, 71, 72, 73], further elaborating the life-cycle model of Macintosh [119], with its agenda setting, issue analysis, policy development, implementation and monitoring stages.

All tasks in the governance life-cyle require argumentation dialogues among stakeholders about legal or regulatory issues. It is no accident that much work on the modern theory of argumentation has been conducted by legal philosophers, such as Pereleman [125, 126], philosophers with a great interest in the legal domain [161, 172], or interdisciplinary work by lawyers and computer scientists in the field of artificial intelligence and law, such as [7, 136, 166, 122, 134, 115, 92, 18], not to mention my own work on the Pleadings Game [67, 68].

The implementation phase of the governance life-cycle, in the case of public governance, is about public administration, for example of tax or social welfare legislation. This encompasses not only the modification of administration procedures and organizational measures, but increasingly also the development of computational models of the regulations and computer systems based on these models for supporting dialogues about rights and obligations between public agencies, businesses and citizens. These dialogues are not always understood or characterized as involving an exchange of arguments between citizens or business and government, but viewing them this way

[3] There are of course methods for solving practical problems which do not make use of argumentation, such as those based on decision theory, but in this section we are only concerned with application scenarios for argumentation technology, not a comparison of argumentation technology with other approaches to solving practical problems.

opens up the possibility of applying argumentation technology to improve the quality, transparency, accountability, acceptability and efficiency of these interactions.

Alternative dispute resolution (ADR) has become the name covering all procedures for resolving legal conflicts outside of the traditional judicial process, such as arbitration and mediation. Argumentation technology exists, also as commercial products, for helping mediators and other third-parties to visualize arguments and negotiate settlements. Some of this is collaborative software ('groupware') for supporting negotiation and deliberation dialogues online, typically on the Internet using the World Wide Web.

Argument visualization technology, such as Araucaria [141], Compendium [151], Rationale [10] and ArguNet [24], is being increasingly used in critical thinking and philosophy courses [97] and for education in other fields, such as law [3, 36].

Finally, argumentation technology is being used to support scientific discourses. For example, Horn has created argument maps of 'great debates' in science, such as the issue in artificial intelligence about whether or not it is possible to construct computers which can be said to think [103]. And Buckingham-Shum and his colleagues at the Knowledge Media Institute have developed a web-based system, ClaimMaker [114, 155], for collaboratively constructing and browsing, much like a Wiki, an index of scientific claims and argumentative relations among claims, with links to the original scientific sources.

Chapter 2

Contributions of Submitted Publications

Sixteen publications, out of a body of more than fifty publications, have been selected for inclusion in this cumulative thesis. They include publications from the beginning of my research career up until the recent past, over a period of more than 20 years. The publications are presented in temporal order, from the earliest to the most recent.

Each section presents a single publication and includes an analysis of the contribution of the publication to the whole body of my research on argumentation technology. Taken together, the submitted papers reveal a continuous, sustained research program on argumentation technology founded on some basic premises which distinguish this body of work from most research on computational models of argument:

- Argumentation is a synthetic, modeling activity, in which theories and arguments are constructed together in a purposeful, goal-driven way.
- The space of potential arguments in realistic application domains is not only infinite but not even recursively enumerable. No procedure for generating all possible arguments systematically exists, typically.
- Although classical logic sets the standard for normative models of valid inference, about the necessary consequences of propositions and what it means for propositions to be inconsistent, further normative models are needed for practical reasoning and decision-making in the face of incomplete or inconsistent information and limited resources for problem solving.
- For practical reasoning it is useful to represent knowledge as defeasible rules with exceptions, where these rules represent not what is probably true in the domain, empirically, but rather encode heuristics about how to solve problems and make decisions in the domain, so as to minimize risks of errors.
- Argumentation typically takes places in dialogues. Procedural norms regulate these dialogues, including the allocation of proof burdens and standards. Which claimed propositions are acceptable at the end of a dialogue depends not only relations between these claims and assumptions about the rules and facts of the domain, but also on the actual moves which were made by the parties during the dialogue, in accordance with these procedural rules.
- For helping to assure that computational models of argument are realistic and practically relevant, it is useful to adopt a requirements-driven research strategy, by building and validating prototype tools and applications in real application domains, such as the law.

2.1 The Role of Exceptions in Models of the Law

"The Role of Exceptions in Models of the Law" [58] was the first of my publications to note that legal rules are typically formulated in legislation and regulations as a general rule subject to exceptions stated separately in other rules, and to consider the consequences of reformulating such rules as material implications of classical logic and using deduction to draw legal conclusions. Two arguments were made. The first cited mainly Bing [25] to support the thesis that legal reasoning is not primarily deductive, but rather a modeling process of shaping an understanding of the facts, based on evidence, and an interpretation of the legal sources, to a construct a theory for some legal conclusion. Hart [99] had noted that concepts are 'open-textured' and cannot be defined finally in advance of the process of analyzing particular cases:

> ... we have no way of framing rules of language which are ready for all imaginable possibilities. However complex our definitions may be, we cannot render them so precise that ... for any given case we can say definitely that the concept either does or does not apply to it. ... Hence there can be no final and exhaustive definitions of concepts, even in science.... We can only redefine and refine our concepts to meet the new situations when they arise.

Notice that Hart's claim is not limited to the legal domain. He explicitly points out that concepts are open-textured, "even in science".

Bing analyzed the process of recognizing and evaluating a legal problem, during interviews between attorneys and clients. The attorney must construct the legally relevant facts from the client's story and find sources of legal norms, in statutes or cases for example, which can be interpreted so as to construct a model of the legal norms which, together the constructed facts, can be used to construct arguments in favor of the client's claims. I wrote:

> The attorney is not a passive listener to the client's version of his tale. Rather, the attorney creates a version of the facts compatible with his conceptualization of the law and attempts to illicit facts from the client which may be relevant in terms of this conceptualization. Similarly, knowledge of the law is required to guide the search for relevant legal texts. A reciprocal process occurs here. The attorney's knowledge of the law guides his selection of cases and statutes, but his subsequent reading of these sources may extend, modify or update his knowledge of the law (which, in turn, may require him to modify his characterization of the facts or discover additional information). This process repeats until the attorney believes he can make out a satisfactory case for his client, or that no case can be made. Furthermore, later negotiations with opposing parties may require further investigation and research which in turn require him to modify his arguments or characterization of the facts. As for the construction of legal arguments, they are constructed provisionally throughout this process, using the model which is built up from the attorney's previous understanding of the law together with the modifications resulting from his interpretation of the legal sources he has consulted. Only by constructing and evaluating these provisional arguments can the attorney decide whether further legal research and factual investigation is necessary.

Thus already in this, one of my earliest publications, I viewed reasoning in particular cases as a modeling process in which arguments and counter-arguments are constructed and exchanged in dialogues, which goes beyond the mere deductive application of rules to facts.

Now, there is a difference between the task of finding or constructing arguments in the process of making decisions, and the task of justifying such decisions after they have been reached. Some adherents of classical logic, such as Alchourron and Bulygin [2], have argued that the process of justifying, if not finding, decisions is "essentially deductive" or at least "can be reconstructed as a logical inference" [2, p. 9]. I do not dispute this view, per se, but only the conclusion with regard to *how* to use classical logic to build expert or knowldege-based systems which one might be tempted to draw from this, namely that rules can be adequately modeled as material implications in classical logic, in advance of the process of analyzing particular cases, and used to generate justifications for decisions.

My second argument in “The Role of Exceptions in Models of the Law” was a novel, economic argument against this idea of modeling rules as material implications in classical logic, inspired by Posner’s economic theory of the law [131]. The utility of a set of rules should outweigh the costs of building, maintaining and using the rules. In dialogues, the relation of costs to benefits may be different for each of the parties. Gathering evidence to prove facts necessary to establish the antecedents of rules consumes resources. Rules should be organized and structured in such a way as to allow a quick and inexpensive preliminary analysis of a situation. In some sense the result of this analysis should be ‘probably’ correct. Further effort and expense should lead to a deeper, more complete analysis which is still more likely to be correct. But whether or not additional resources should be expended depends on a risk assessment of the cost of making an error against the cost of this additional effort. Rules should be structured so as to allocate the costs of fact finding to the party who is able to discover the required evidence at lowest cost and to minimize the amount of damage caused by a mistaken decision due to incomplete information. The rules should be structured so as to facilitate dialogue protocols which can regulate presumptions and the burden of proof to allocate the costs and benefits of further research and investigation among the parties in a fair way.

These requirements for rules were then argued to not be met when rules are reconstructed as material implications, rather than preserved in their traditional form as general rules subject to exceptions, since such a logical reconstruction of the rules abstracts away control information which is vital for meeting these requirements.

2.2 Oblog-2

In “Oblog-2: A Hybrid Knowledge Representation System for Defeasible Reasoning” [59], I developed a hybrid knowledge representation formalism and inference engine supporting reasoning with both taxonomies of terms and rules subject to exceptions. The system was designed to support the modeling conception of reasoning with rules. I already took the view at this time that reasoning with rules is a modeling process in which deduction has an important albeit subordinate role to play. The terminological component of Oblog-2 was inspired by KL-ONE [28] and research on inheritance hierarchies [49, 162]. Oblog-2 allowed a taxonomy of ‘types’ to be represented along with binary relations on these types, called ‘attributes’, similar to the concepts and roles of description logic, respectively. Types were represented by their *genus*, i.e. their parent type in the taxonomy, and *differentia*, i.e. the attributes which distinguish instances of the type from instances of the parent type. These were not definitions, since they expressed only sufficient conditions, not necessary conditions. Rules were modeled as Horn clauses indexed by attribute (predicate) and type. If more than one rule was applicable to some entity, the type taxonomy was used to prefer the most specific rule, following the principal of *lex specialis*, resulting in a kind of nonmonotonic inference. A kind of “certainty factor”, comparable to certainty factors in production rules systems like EMYCIN [123], was computed for each inference, using the number of exceptional rules in the taxonomy below the type associated with each rule used.

Oblog-2 suffered from a number of problems and limitations. Its terminological component, unlike KL-ONE [28] at the time and description logic [11] much later, was not able to infer subsumption relations between types from their descriptions. Rather these needed to be asserted explicitly in the taxonomy. Specificity was the only principal used to resolve conflicts among rules. Other ordering principles, such as preferring newer rules, were not supported. It was not possible to represent explicit exceptions (‘unless’ clauses in rules), exclusionary rules or priority rules. However this was a limitation it shared with most, if indeed not all other approaches to nonmonotonic reasoning at the time. Oblog-2, like 'pure' Horn clause logic, supported no kind of negation operator. Prolog's negation as failure was not used, since it is based on the assumption that all relevant facts are known and have been asserted. In our intended applications, exactly the opposite assumption applies: that the information available is incomplete. Finally, although I already had begun to view practical reasoning as a model construction process, Oblog-2 did not explicitly construct arguments from rules or provide much support for searching the space of models. Users would have needed to manually explore the space of models by defining taxonomies and rules and using the inference engine to test their consequences.

2.3 Some Problems with Prolog as a Knowledge Representation Language for Legal Expert Systems

In "Some Problems with Prolog as a Knowledge Representation Language for Legal Expert Systems" [60], I put forward some of further arguments against modeling rules as material implications, more specifically in the form of Horn clauses as used by the logic programming language Prolog [38]. Again, legal reasoning was claimed to be a modeling process in which theories are constructed, rather than principally a deduction task, this time citing a paper by the director of my research institute at the time, Fiedler [50], rather than Bing's somewhat earlier work [25].[1]

Fiedler wrote [50]:

> In effect, the task of the judge essentially includes the choice, shaping and logical construction of the appropriate legal rules as well as the pertinent statements of facts in mutual interdependence. It is true that the resulting fabric of the judgment and its reasons has the function of deductively connecting the decision to the rules of law and the facts of the case. Nevertheless the process of decision-making is not reduced to the application of deductive logic to given premises, but essentially consists in constructing a logical fabric, which at the same time has the qualities of an adequate model and a stringent deduction. In terms of modern methodology, judicial decision-making will have to be qualified as a process of model-construction or 'modelling'.

Even as an account of how legal decisions are justified, deduction is not sufficient, since the universal norms of logic are but one constraint on the power of judges to construct theories, and proofs from these theories, among other constraints. In particular, statements in the theory about the facts of the case and the applicable law must be shown to be sufficiently supported by evidence and a reasonable interpretation of legal sources, respectively. The problem of classifying the *primary facts*, about the events of the case, under the *secondary facts*, about the antecedents of the applicable rules, called the "problem of classification" or, in Europe, the 'subsumption problem', cannot be solved by some kind of formal matching or unification process, but rather may itself require the construction or revision of some theory of the law. Theories of the law and facts are constructed in an iterative process, searching for theories which lead to some desired result. Thus, whereas deduction in classical logic is an objective task which does not depend on the interests or goals of some agent, theory construction is a *purposeful* activity, conducted by agents pursuing practical goals.

One of the main new arguments I put forward in this paper related to the normative function of rules. The paper addressed only the legal domain, but in retrospect these arguments can be generalized to any domain in which rules are used to guide action by encoding policies about

[1] In retrospect, both Bing and Fielder were presumably influenced by the Engisch [47], a German legal philosopher, who spoke twenty years earlier of the way one's attention must shift back and forth ("Hin und Her-Wandern des Blickes") between the evidence and legal sources when trying to subsume facts under legal terms. See also Rawls' "reflective equilibrium" theory of moral reasoning, which claims that general moral principals and judgments about the morality of specific acts are constructed together, in an iterative process of mutual adaptation [139].

risks, such as the "business rules" of companies. The first argument was based on the observation that rules are not intended primarily for resolving disputes after they have arisen, or justify legal decisions, but rather to guide behavior so as to prevent and avoid such disputes in the first place. Rules can guide behavior only to the extend that it is possible for individuals to *learn* the rules. That is, rules must be structured in such as way as to take the cognitive capabilities of humans into account. Complex rules with many antecedents are difficult to remember. The traditional structure of general rules with separate exceptions makes it easier for people to at first learn the basic rules and, over time, also become familiar with the exceptions. Since the rules are designed to guide behavior so as to minimize risks, being able to remember only the basic rules, even though these are incomplete, is better and safer than not being able to remember any of the rules at all. Furthermore, it was argued that structuring rules as general rules and exceptions makes it possible to modify some norms by adding additional exceptions, without having to change the basic rules, thus avoiding the need to unlearn the basic rules.

Other arguments in the paper were more specific objections to the use of Horn clause logic for modeling legal rules, as suggested by an influential case study done at Imperial College, in which Prolog was used to model the British Nationality Act [152]. The simple depth-first control strategy of Prolog and its model of negation as failure do not reflect the way presumptions and burden of proof are used in the legal procedures to control reasoning in the face of incomplete information or evidence about the facts of the case. Since the structure of general rules and exceptions is abstracted away during the process the modeling the rules using Horn clauses, it becomes difficult, if not impossible to explain any conclusions derived from the rules in terms of the original legal sources. I wrote:

> An expert system, especially a legal expert system, must be capable of explaining its reasoning in terms users can appreciate. To achieve this transparency, it is helpful if the computer model and the legal sources upon which the model is based have a similar structure.

This idea was echoed a couple of years later by Routen [146], who cited a later paper of ours [62], on the importance of nonmonotonicity for legal reasoning, discussed below, which reiterated these arguments. Routen was a colleague of Bench-Capon. Together they published a journal version of this work [147], which led to Bench-Capon's later publications on the 'isomorphic modeling' of legislation [14, 15]. See also Karpf's work on the isomorphic modeling of statutory law [105], which was published at bit earlier than Bench-Capon's but may have been an independent development.

2.4 The Argument Construction Set

In "The Argument Construction Set" [61], I presented the basic design of my first interactive system for supporting the construction and evaluation of arguments. This design was implemented by Schweichhart [150] for his German diploma thesis, which I supervised. At the time, there was a controversy within the field of artificial intelligence and law, initiated by Leith [111, 112] in response to a pilot project in which Prolog was used to model the British Nationality Act [152]. Leith argued that, contrary to Hart's thesis [99], that there no clear legal cases and thus no 'clear rules'. If there are no clear rules, then how can rule-based systems be useful for legal reasoning? My contribution to this debate was to argue, based on my view of legal reasoning as a modeling and theory construction process, that rule-based systems can be useful for generating arguments, even in hard cases. The conclusions generated by applying such rules are only presumptively true and can be rebutted or undercut by other arguments. I called my approach a "constructive model" of decision-making, to contrast it with the deductive model:

> In legal expert systems based on this constructive model of legal decision making, knowledge bases could be prepared in advance by legal experts, just as is done in

> conventional expert systems based on the deductive model, but one must be careful to design the system so as to prevent the experts from usurping the responsibility of the lawyer deciding the case for the ultimate content of the model of the law used.

In this paper I also contributed further arguments against the utility of classical logic as a model of practical reasoning, due to decidability and computational complexity issues. Propositional logic is decidable but intractable. Full first-order logic is undecidable. Some subsets of first-order logic, such as Horn clause logic, are semi-decidable, but a normative model of practical reasoning requires a tractable decision procedure. The goal of argumentation protocols is to define rational procedures for making decisions within given resource limitations, even in the face of incomplete or inconsistent information.

I reiterated my position that determining the truth of propositions is not the primary goal of decision-making. Rather the goal of decision-making is to choose actions, taking into account risks and values or utilities:

> ... In the context of common sense reasoning, truth is not the principal goal, but rational, considered decisions. Decision-making occurs in time and space and is subject to limited resources, in particular limited information. When faced with a decision to make, if a relevant proposition cannot be deduced from available information, then other forms of reasoning must be applied. If the proposition is usually true in contexts similar to the current one, then an agent may assume it to be true in this case as well. That is, some form of probabilistic reasoning may take place. Even if the proposition is probably true, however, an agent may choose to believe that it is not true, as he must consider the consequences of making a wrong decision. The relative risks of error must be weighed.

Consider Walton's example, which he attributes to Carneades, the ancient Greek philosopher, about walking through a cave and seeing something which looks like a snake [171, p. 52]. Even if it is probably just a coiled rope, it may be safest to assume it is a snake and decide to walk carefully around it. In the paper I used the following legal example. Due to the care usually taken by the police in crime investigations, accused persons are probably guilty. But since civilized societies consider it worse to punish an innocent person than to let a guilty person go free, defendants in criminal cases are presumed innocent until proven guilty beyond a reasonable doubt.

Finally, I reviewed several nonmonotonic logics and considered their suitability for practical reasoning. I came to the conclusion that research in the field of nonmonotonic logics had not sufficiently investigated the *process* by which decisions can be made under conditions of limited information, time and other resources:

> Despite this interest in computation, however, researchers who have taken a logical approach to the problem of nonmonotonic reasoning have generally not shown interest in the process of reasoning with incomplete and changing information over time. Rather they have generally focused their attention on the traditional model-theoretic and proof-theoretic issues of logic.

This was when I began to shift my attention away from modeling practical reasoning as an inference relation and began thinking seriously about procedural, dialogical approaches. But I didn't quite go this far in the Argument Construction Set (ACS). Rather, the ACS was a modeling environment for constructing and modifying arguments, represented as 'jutifications' in a reason maintenance system and exploring the effects of these arguments on the acceptability of statements. The ACS addressed the argument comparison and evaluation use case of the dialectial layer, but not the use of argumentation dialogues and protocols or other procedural tasks.

It was in "The Argument Construction Set" that I first made a serious attempt to develop a system supporting the construction and exploration of arguments from rules, in accordance with our modeling view of practical reasoning. Dependencies among arguments were managed using a justification-based reason maintenance system, based on Doyle's Truth Maintenance System [43]. Such reason-maintenance systems compute whether a proposition is 'in' or 'out', given a set of 'justifications'. In Doyle's semantics 'justifications' are viewed as constraints and a proposition is considered to be 'in' so long as the constraints are satisfied; our system viewed justifications as arguments and deemed a proposition to be 'in' only if a defeasible proof of the proposition could be constructed from the given arguments. Rules in the Argument Construction Set (ACS) had the form of Horn clauses extended with an 'unless' operator, where "unless p" was interpreted to mean that p is not 'in' given the arguments considered thus far in the modeling process.

Arguments could be constructed from alternative interpretations of the valid rules. Each argument constructed from a rule r included a premise stating that r is valid. The (defeasible) consequences of some set of rules could be explored by first assuming, for each rule r in set, the proposition 'valid r' to be in, using the reason maintenance system. The ACS included a "tactical reasoner" for searching the space of rule sets, where each state in the space is a set of rules assumed to be valid, looking for states in which some goal proposition is in or out.

The ACS offered some advantages compared to Oblog-2, but also some disadvantages. The ACS had an 'unless' operator and a negation operator, but negation was not handled well. Negation was not interpreted as failure, but both p and $\neg p$ could be in simultaneously. p being in did not imply $\neg p$ being out, or vice versa. Using the unless operator, the ACS supported one form of explicit exceptions, but it still did not support exclusionary rules or priority rules. And, unlike Oblog-2, the ACS did not use specificity to order rules, or arguments from rules. The ACS was mainly intended as a tool helping an individual to explore and evaluate arguments. It was not a dialogical model of argument, although I did already anticipate extending the ACS to support argumentative dialogues, helping multiple users to "engage in a debate or dialog regarding some issue" [61, p. 20].

The ACS was implemented by Schweichhart as part of his Diploma thesis [150], which I supervised.

2.5 The Importance of Nonmonotonicity for Legal Reasoning

In "The Importance of Nonmonotonicity for Legal Reasoning" [62], I reiterated many of the above arguments of these earlier publications, but put them into the context of work on nonmonotonic logics, influenced by the work of a colleague at the time, Gerhard Brewka, who was in the process of writing a book on the subject [33], based on his PhD thesis [32]. I noted that classical logic does not take into account the "pragmatic aspects of reasoning about problems in time, where it may be necessary to make a decision based on incomplete knowledge of the facts, only to discover at a later time that our assumptions had been incorrect." Recalling the normative purpose of the law, which "presupposes that ordinary persons are able to learn and apply the law with some success", I countered the possible argument that laws could be organized in a comprehensible, hierarchical way, without resorting to general rules with exceptions, by hiding complex combinations of antecedents behind the definitions of abstract concepts. This approach creates a dilemma: either some technical term must be introduced, which is not meaningful to the target community, some common term must be given a technical meaning which deviates from its usual meaning, or references to paragraph or section numbers, e.g. "unless § 23 applies", must be used. All of these approaches are inferior to the use of general rules with exceptions, if our goal is to make it easier for people to learn the basic rules. I also emphasized a fundamental and perhaps obvious requirement of any logic for making practical decisions, namely that it must provide some rational guidance for actually and conclusively making the decision, even in the face of incomplete or conflicting evidence. Using classical logic, it may be that neither a proposition nor its complement are entailed by some theory of the case. Classical logic provides no rational means to prefer one or the other alternative in such cases. Structuring knowledge using general rules with exceptions always provides a default decision, which can be used as a fallback plan if necessary. For example, in criminal law, the accused is presumed innocent unless proven guilty. So, if evidence sufficient for proving guilt beyond a reasonable doubt is lacking, the judge must decide to acquit the defendant.

2.6 Issue Spotting in a System for Searching Interpretation Spaces

In "Issue Spotting in a System for Searching Interpretation Spaces" [63], I presented the successor to the ACS. It used de Kleer's Assumption-Based Truth Maintenance System (ATMS) [41], instead of the Doyle-style reason maintenance system developed for the ACS. The disadvantage of the ATMS, and the reason I initially chose not to use it, is that it manages dependencies only among strict arguments, i.e. arguments in which the conclusion is necessarily true if the premises are true, and not just presumptively true. This limitation of the ATMS was overcome here using Poole's assumption-based approach to default reasoning [130], which was later extended by Bondarenko et al. into an assumption-based model of argumentation [27]. Each strict argument $p_1, ..., p_n \Rightarrow c$ gets translated into a defeasible argument of the form $\delta, p_1, ..., p_n \Rightarrow c$, where δ is a proposition expressing the assumption, called a 'hypothesis' by Poole, that the argument is consistent with the facts. This argument is undercut by an argument of the form $q_1, ..., q_n \Rightarrow \neg\delta$, whenever every proposition in $q_1, ..., q_n$ is entailed by the facts. An 'extension' in Poole's theory is a maximum set of assumptions which is consistent with some theory of the domain and the facts. For any proposition at issue, one can check whether it is entailed by the theory in one or more extensions. The greater the number of extensions entailing the proposition, the more confidence one can have that it is true, since it depends on fewer assumptions. De Kleer's ATMS provides a tool for implementing this model of argumentation, since it can be used to compute the extensions entailing a proposition. The scheme for argument from rules realized in this system included not only a premise stating that the rule was valid, as in the scheme used for the ACS, but also a premise stating the assumption, called δ above, that the argument constructed by instantiating the scheme is consistent with the facts. I later applied this system to design a system for assembling legal documents [65].

2.7 An Abductive Theory of Legal Issues

In "An Abductive Theory of Legal Issues" [64], I developed a computational model of Hart's concept of 'clear cases' which did not depend on Hart's positivistic conception of legal reasoning [99]. Although Hart recognized the open-texture of concepts, his theory of legal reasoning was none the less positivistic, because he claimed it was possible to uniquely identify and define the currently valid legal rules, using a secondary 'rule of recognition'. For Hart, a case was clear if it can be decided by applying these rules to the facts of the case. Otherwise the case is not clear, but 'hard'.[2] Deduction in Hart's view is sufficient in clear cases. I revised Hart's theory of clear cases, removing its dependency on positivistic assumption, by redefining clearness relative to a set of competing interpretations and theories of the facts of the case and the relevant legal sources. These theories can be constructed during the analysis of the case, in accordance with my modeling and theory construction view of practical reasoning.

In Hart's positivist theory of the law, judges only have discretion to extend the set of valid rules when deciding unclear cases, by partially defining open-textured concepts, reducing the 'penumbra

[2] More formally, let Γ be a set of propositions in classical logic, expressing the valid legal rules which, according to Hart, can be found by applying the rule of recognition. Let Δ be a set of propositions describing the facts of some case and let ϕ be a proposition at issue in the case. According to Hart, the case is clear regarding this issue if and only if $\Gamma \cup \Delta \models \phi$ or $\Gamma \cup \Delta \models \neg\phi$.

of doubt' in future cases. Dorkwin, another legal philosopher, constrained the discretion of judges even further than Hart, by arguing that legal decisions must conform not only to the valid legal rules of some jurisdiction, i.e. positive law, but also to general legal 'principals' which are more universal, some of which might be founded in natural law [45]. I view such principals as another source of arguments, along with arguments from other legal sources, such as statues and case law. So while I agree with Dworkin that there are other 'constraints' on judicial discretion than the positive law, since the space of possible arguments is larger than the space of arguments from positive legal rules and arguments constrain the decisions which can be justified, my theory construction view of legal reasoning nonetheless give judges broader discretion than either Hart's or Dworkin's theory. In computer science terms, Hart and Dworkin seem to assume that the space of arguments from rules and principals, respectively, is recursively enumerable. I doubt this to be the case. The space of arguments from rules and principals is not only infinite, but not well-defined. Creative lawyers and judges can interpret legal sources in new ways, constructing new arguments outside any *a priori*, recursively enumerable space of arguments. Thus, in my model the concepts of clear cases and legal issues was defined using abduction from a set of hypothesis constructed from the finite set of arguments which had been actually put forward by the parties in a dialogue, rather than the infinite and not recursively enumerable set of potential arguments. Whether or not a case is clear can only be determined retroactively, given the arguments which were actually made. An apparently clear case might have been less clear if one or more of the parties had been more creative in their argumentation.

It was in "An Abductive Theory of Legal Issues" [64] that I first published an explicit list of argumentation use cases, which eventually led to the use case model of argumentation illustrated in Figure 1.2. I wrote:

> ... legal reasoning, in its full breadth and richness, consists of a variety of reasoning tasks, such as:
>
> 1. Identifying and interpreting relevant legal texts, such as statutes;
> 2. Constructing alternative views of the facts;
> 3. Designing legal arguments;
> 4. Managing dependencies between versions of the facts, alternative interpretations of legal texts, arguments and inferences;
> 5. Selecting the best arguments; and
> 6. Documenting chosen arguments persuasively.

2.8 A Theory Construction Approach to Legal Document Assembly

In "A Theory Construction Approach to Legal Document Assembly" [65] I presented the design of practical application of my theory construction model of legal reasoning. One of the most widespread classes of applications of computers in the legal domain are "document assembly" systems. The first system of this kind was Sprowl's ABF processor [159], so named because it was developed in a project with the American Bar Foundation. There are now quite a number of commercial applications of this kind, including one offered by a major legal publisher. Most of theses systems model the relevant legal rules in an entirely procedural way. Procedural code modeling the logic of the law is embedded within "boilerplate" blocks of text containing variables to be instantiated with facts of the particular case, much as many dynamic web sites are now implemented by embedding procedural code in some scripting language within blocks of HTML code.

Thus, the conventional approach to legal document assembly presumes that the relevant legal rules and concepts can be modeled adequately in procedural code prior to a legal analysis of the facts of particular cases. This approach is compatible with the "mechanical jurisprudence" view of legal reasoning, which claims that the legal consequences $\mathcal{C}$ of a case can be derived deductively using classical logic from a prior axiomatization $\mathcal{L}$ of a theory of the law and the facts $\mathcal{F}$ of the case: $\mathcal{L} \cup \mathcal{F} \vdash \mathcal{C}$. But rather than using a theorem prover to derive these consequences, procedural code implementing the theory of the law is implemented manually for particular reasoning tasks.

A more advanced knowledge-based approach to legal document assembly was developed in the German KOKON project [107]. In KOKON, the legal rules were separated out from the text templates and modeled in a declarative rule-based language.

My aim with this paper was to design a legal document assembly system founded on the modeling [51] conception of legal reasoning, in which the theories of the facts of a case, theories of the applicable law, and arguments recording dependencies between these theories and their consequences are constructed together in an iterative, goal-directed process.

A principal feature of the system was that it applied argumentation methods uniformly to construct and evaluate legal theories and documents, in an integrated fashion. Knowledge of the legal domain and of relationships between boilerplate text templates was modeled declaratively, using first-order logic. Arguments were constructed from this knowledge base using Poole's THEORIST approach to defeasible reasoning, in which theories are constructed using abduction from assumptions and hypotheses [130]. Dependencies between arguments were managed using de Kleer's Assumption-Based Truth Maintenance System [41]. Text templates were modeled as terms in first-order logic. Particular text blocks were constructed by unifying the variables in the templates with constants representing entities of the case.

The system was designed to assist human users explore the consequences of different theories of the law and facts. The ATMS computed the minimal set of assumptions necessary for each theory. The subset of formulas entailed by a selected theory which represented the document were translated to the Standard Generalized Markup Language (SGML) [35] and then processed by an SGML document formatting system to generate the final document.

The system was rather influential in the AI and Law community [109, 30, 40] but, unfortunately, to my knowledge the system was never implemented in its entirety, even though its basic design still seems sound to me. Recently, I've returned to the subject of document assembly systems, in collaboration with Marc Lauritsen. We've begun designing a system which makes use of argumentation technology for a variety of document assembly tasks [110].

2.9 The Pleadings Game

In "The Pleadings Game — An Exercise in Computational Dialectics" [67], the journal article summarizing my Ph.D. Thesis [66], later published as a book [68], I further developed my theory of how the model construction view of practical reasoning rationally constrains the decisions which can be justified, and thus the discretion of judges, without accepting the basic tenets of positivism. The main innovation, inspired by Alexy's theory of legal argumentation [4], was to shift my focus from the modeling activity of an individual to the process of collaboratively constructing models in argumentative dialogues, regulated by procedural rules or discourse norms, now also called, following Loui [117], 'protocols'. The Pleadings Game was quite influential in the artificial intelligence and law field, since it was one of the first efforts, along with Hage [93], to construct a computational model of legal reasoning from a dialogical perspective. It was here that I also first coined the term "computational dialectics", which I described as a "new subfield of artificial intelligence" whose:

> subject matter is the design and implementation of systems which mediate and regulate the flow of messages between agents in distributed systems, so as to facilitate the recognition and achievement of common goals in a rational, fair and effective way. [67, p. 240]

One of the contributions of the Pleadings Game was to identify some of the dimensions of the design space of argumentative dialogues:

> The design space for systems of this kind is large. The dimensions of discourse games in this space include the purpose of the game, the types of data exchanged in speech acts, the types of speech acts of 'moves', the number and roles of the players, the rights and obligations or 'commitments' of the players, the kinds of resources and the means of their distribution, and finally the kinds of issues which may be raised and addressed during the game. [67, p. 240]

I also coined the term 'mediation system' in the Pleadings Game:

> The Pleadings Game demonstrates that a machine can monitor a discussion, helping ensure that discourse norms are not violated. ... Unlike the idea of a computer judge, there would seem to be little basis for fundamental opposition to the idea of a *mediation system*. The human judge is retained, as a player with a particular role, whose discretion is restricted by the rules of the game.
>
> There is an important difference between a mediation system and ... expert systems, as they are usually conceived. In expert systems, the knowledge base is intended to be a single, consistent theory of some domain, with which users have little opportunity to disagree. A mediation system supports a discussion about alternative theories. A theory is constructed during the game. Experts still have a role to play. They can prepare formal theories, comparable to traditional treatises, which the players can mold into arguments. [67, p. 290]

In the full dissertation and book versions of the Pleadings Game [66, 68] I contrasted relational and dialogical models of practical reasoning, such as legal reasoning. I used the term 'relational model' to broadly cover all models in which rational or acceptable decisions are defined as a mathematical relation between some representation of the knowledge of the domain, the facts of particular cases, arguments and decisions, no matter whether the logic used to define the relation is classical or nonmonotonic. Missing from such relational models is the concept of practical reasoning as process, as dialogues regulated by discourse norms.

I conjectured that "no relational model of legal reasoning can adequately model judicial discretion" [68, p. 5]. In retrospect, this conjecture was more specific that it could have been. Today I would generalize this to claim that no relational model is adequate for practical reasoning, in any domain.

In the Pleadings Game the rule language of the Argument Construction Set [61] was developed further, to better distinguish various ways of undercutting and rebutting arguments from rules and provide better support for modeling rules as they are usually formulated in legislation. Rules were reified and made subject to explicit exceptions; conflicts between rules could be resolved using both specificity and rules about priorities; the applicability of rules could be reasoned about and excluded by other rules; and the validity (backing) of rules could be questioned.

The way arguments were constructed from rules in the Pleadings Game is somewhat complicated. Rules were interpreted as a *background context* of a default theory in the nonmontonic logic of conditional entailment [56]. A background context is a pair $\langle L, D\rangle$, where L is a set of closed formulas of predicate logic and D is a set of *defaults*. A default is a pair $\langle p, q\rangle$, denoted $p \Rightarrow q$, where both p and q are first-order formulas which may contain free variables.

Rules had the form `(rule` r `(`$x_1 ... x_n$`) if` a `then` c `unless` e`)`, where r is a symbol naming the rule, $x_1 ... x_n$ are schema variables, and a, the antecedent of the rule, c, the conclusion of the rule, and e, the exception of the rule, are formulas of predicate logic which may contain free variables.

Rules were interpreted by mapping them to a background context $\langle L, D\rangle$ of conditional entailment as follows. L consists of four formulas of first-order logic. The first stated that the antecedent of the rule is satisfied if a is true and r is a valid rule.[3] This purpose of this formula was to reveal the requirement that the rule be valid. In effect, validity is an implicit antecedent of every rule. The next two formulas, taken together, stated that the conclusion of the rule is true, for a given *instance* of the rule, where a rule instance is formed by replacing the schema variables of the rule with constant terms, if the antecedent of the rule is true, including the validity condition, assuming the rule can be consistently applied in this instance. This assumption was denoted by formulas of the form `(ap (inst` r `(parms` $x_1 ... x_n$`))`. The final formula stated the rule cannot be consistently applied in this instance if the exception of the rule is true in this instance. The default D for this rule stated that it can be presumed that the rule instance applies if the antecedent of the rule instance, including the validity condition, is true.

Arguments in the Pleadings Game were interpreted as sets of predicate logic formulas, in accordance with the argumentation-theoretic proof theory of the logic of conditional entailment. Arguments were constructed from a rule by instantiating the rule, replacing its schema variables with constant terms, and including the applicability assumption for the rule instance in the set of formulas making up the argument, along with whatever other formulas were necessary to support the conclusion of the argument.

Conditional entailment uses specificity to order conflicting arguments. This feature of conditional entailment is made use of in the mapping of rules to background contexts of default theories to assure that arguments from more specific rules, i.e rules with more specific antecedents, are preferred. Explicit exceptions to rules are supported by the final formula in the mapping, stating that a rule cannot be consistently applied in some instance if the exception of the rule is true in the instance. Principles for prioritizing rules, such as *lex posterior* (prefer the later rule) and *lex superior* (prefer the rule from the higher authority) could be formulated as rules, making use of a syntactic criterion in conditional entailment for specificity. This technique could also be used to model exclusionary rules that cancel the applicability of some rule under certain conditions.

Conditional entailment was chosen as a foundation for the Pleadings Game because it was a state-of-the-art nonmonotonic logic at the time, with both a model-theoretic semantics, using preference orderings on models [153] and an argumentation-theoretic proof theory. But conditional entailment does have a weakness. Specificity is the overriding ordering principal for resolving conflicts among arguments. For example, an argument from a more specific rule had priority over an argument from a more general rule, even if this more general rule was from a higher authority. This may not be correct, depending on the application domain. What is needed is some way to use domain-specific principals and rules to argue about whether or not a more specific argument has priority over some argument which may be preferable for some other reason.

Another problem with conditional entailment as a model of argumentation relates to the way arguments are modeled as set of propositions, rather than as premise-conclusion pairs. In this approach, applying an argumentation scheme, such as the scheme for arguments from rules, does not directly result in a argument, but rather only takes part in the definition of a default theory from which arguments can be constructed. Arguments are constructed from the assumptions introduced by the set of *all* argumentation schemes which have been instantiated, similar to the way explanations are abduced from hypotheses. Instead of a one-to-many correspondence between an argumentation scheme and the arguments constructed by instantiating the scheme, there is a many-to-many relationship between argumentation schemes and the arguments containing assumptions introduced by instantiating these schemes. This makes it more difficult to critically evaluate arguments using argumentation schemes, for example to identify missing premises, since the relationship between the premises and conclusion of the scheme must be reconstructed somehow from the set of propositions.[4] Not having explicit links between the premises and conclusion of each argument also makes it more difficult to explain or justify decisions. It may be possible to

[3]Following Toulmin [161], we used the term 'backing' as a synonym for rule validity in the Pleadings Game. A rule which has such backing is a valid rule.

[4]Which schemes have applied, on the other hand, is apparent from applicability assumptions in the argument.

reconstruct this information from dialectical graphs, i.e. a graph of arguments (as sets of propositions) linked by support, rebut and defeat relations [57, p. 155–162], but this was not attempted in the Pleadings Game.

The Pleadings Game integrated and extended a number of ideas from prior formal models of argumentation dialogues. Like Rescher's system [143], the Pleadings Game went beyond Dialogue Logic [116] and Mackenzie's system [120] by supporting defeasible argumentation about substantive issues, such as whether or not an agreement is contractually binding. In contrast, Dialogue Logic was a game theoretic proof theory for intuitionistic logic. The only kind of issue which could be discussed using Dialogue Logic was whether or not some formula is a tautology in intuitionistic logic. In Rescher's system conflicts between arguments were resolved by ranking them by their specificity. The Pleadings Game, being based on the logic of conditional entailment, also used specificity to resolve conflicts among arguments, but went beyond this with a method for encoding other kinds of priority relations among rules in terms of specificity. This enabled conflicts among arguments to be resolved by arguing about priority relations among the rules used to construct these arguments. Principals for ordering rules, such as *lex posterior* and *lex superior* could be represented in the rule language of the Pleadings Game and arguments about rule priorities could be constructed from these rules. Like Mackenzie's system [120], the Pleadings Game kept track of the claims of the participants in a "commitment store" and used these commitments in the dialogue rules, i.e. the argumentation protocol, to constrain the speech acts of the players. For example, player's were not allowed to make arguments with premises which contradicted statements in their commitment store. But the Pleadings Game strengthened Mackenzie's concept of commitment by defining a tractable but weak entailment relation, called the 'known' relation, to commit participants to some of the logical consequences of the statements in their commitment store. The Pleadings Game was the first system to formalize a notion of relevance and use this to focus dialogues by disallowing irrelevant arguments.

Finally, the Pleadings Game was novel for the type of dialogue, civil pleading, it was designed to simulate. Most prior systems modeled persuasion dialogues, where the goal of the dialogue is to determine the acceptability of some claim. In persuasion dialogues, the proponent of the claim tries to prove the claim and the opponent tries to prevent the proponent from doing so.[5] In civil pleading, however, the goal is only to identify the issues of case, not resolve them finally. The output of the Pleadings Game consists of a theory of the law and facts agreed upon by the parties, a set of issues which they were unable to resolve and a network of arguments clarifying dependencies between the main issue of the case and alternative resolutions of these open issues.

Thus, the Pleadings Game can be viewed as a computational model of a multi-agent version of the theory construction conception of practical reasoning.

2.10 Computational Dialectics

In "Computational Dialectics" [69] I provided another description of computational dialectics, as a subfield of computer science for studying "computational models of norms for rational discourse" and defended the thesis that "rationality can best be understood as theory construction regulated by discourse norms". I also promoted the use of the term 'mediation systems', first used in the Pleadings Game, for computer systems which use formal models of dialogue protocols to help groups of people to work together on resolving disputes and solving problems and presented the design of the first version of my Zeno mediation system for collaboratively constructing, browsing and visualizing arguments.

This is where I first described some of the conditions under which decisions are typically made:

[5] In some persuasion dialogues, which Walton calls "disputes" [173], the proponent and opponent have symmetrical, reciprocal roles, with the proponent trying to prove a claim and the respondent trying to prove a contradictory claim.

1. There is both not enough and too much information. For some parts of the problem relevant information which would be useful for making a decision will be missing. For other parts, there will be more information than the persons responsible for making the decision will have time to even retrieve, let alone comprehend.

2. The resources which can be applied to finding a solution are limited. Time, in particular, may be "of the essence": a solution must be found before the issue becomes moot.

3. The expected value of the known alternative decisions is not high enough to make it cost effective to invest substantial resources in implementing a program, knowledge base, or other kind of elaborate computer model to use in helping make the decision.

4. However much information is available, opinions differ about its truth, relevance or value for deciding the issue.

5. Arguments can and will be made pro and contra each alternative solution.

6. Reasoning is defeasible. Whatever choice seems best at the moment, further information can cause some other alternative to appear preferable.

7. Factual knowledge about how the world functions and its current state is not sufficient for making a decision. Value judgments about ethical, political, legal and aesthetic factors must not only also be taken into consideration, but are the critical issues requiring the most attention.

8. Several persons have a role to play in making the decision and will be affected by it. Conflicts of interest are inevitable; support for negotiation and other procedures for achieving consensus and compromise are required.

9. Finally, the persons responsible for making the decision are not proficient in mathematics, logic or any other formal methods for solving problems.

I motivated research on computational dialectics by arguing that existing information technology falls short of providing comprehensive decision support given the above conditions, which are prevalent:

> The main purpose and promise of computers and information technology is to improve the procedures for making choices ... in industry, government, and other kinds of organizations and groups. The improvement may be in effectiveness, efficiency or, when the conflicting interests of multiple parties are involved, fairness. The different subfields of computer science contribute to this abstract goal in complementary ways. When there is perfect information about a problem, an efficient algorithm or theorem prover may be used to compute or search for a solution. Large data bases make a wealth of relevant information readily available. Knowledge-based systems are useful for tasks where there is sufficient consensus about the knowledge required and the costs of knowledge acquisition and maintenance can be amortized over the expected life time of the system. High capacity networks and hypermedia technology are making it cheaper and easier to disseminate and access all kinds of information, including text, sound, color graphics and video. So-called "virtual reality" systems and other kinds of computer-simulation make it possible to explore and vividly imagine the likely effects of alternative courses of action. Even applications as banal as word processing, spreadsheets, and electronic mail flourish in the end because of their role in the processing and distributing information to be used in making decisions. As useful as these technologies have been shown to be, none of them squarely confronts the problem of supporting effective, fair and rational decision making procedures under the conditions

which usually prevail. Either they only deal with a part of the problem, such as providing access to relevant information, or they restrict their attention to special problem solving contexts where certain simplifying assumptions, such as perfect information, can be made.

Finally, I responded in this paper to Suchman's concerns [160] about using speech-act theory, and thus argumentation protocols, as a basis for computer systems for coordinating human activity in organizations. Suchman argued that the use of explicit 'categories' in the definitions of speech acts lead to rigid systems which carry with them an "agenda of discipline and control", furthering interests of management at the expense of employees. I drew an analogy with legal systems to argue that such systems can be designed to respect and protect the interests of all stakeholders, including employees, claiming that technology for modelling categories, rules and protocols is interest neutral. The principal of equality under the law requires rules to be expressed using general categories, rather than concrete situations, so that like cases can be treated alike. Suchman's objection to general categories is based on a deductive view of reasoning reminiscent of nineteenth century German conceptualism ('Begriffsjurisprudence'). The tension between the principal of equality under the law and the desire to do justice to the "specificity, heterogeneity and practicality of organizational life", quoting Suchman, can be resolved by using argumentation from cases, principals and values to reinterpret rules when constructing theories of the concrete case at hand.

Suchman's criticism about using speech-act theory as a foundation for computer systems in organizations was in response to a system proposed by Winograd and Flores [175]. Her criticism does have some merit with respect to the particular kind of system proposed, since it used computer models of dialogue protocols not as normative guidelines, but rather to define and constrain the space of possible speech acts in organizations. That is, the system constructed a working environment which restricted the things people were actually able to do in the environment. In Lessig's terms [113], this is using 'architecture' to regulate behavior. Lessig considers four 'modalities' of regulation. Architecture is one. The others are law, 'norms', and the market.[6] While I agree with Lessig's view that architecture is an important tool in the regulator's toolbox, I also accept Suchman's point that using architecture to constrain behavior has the disadvantage of rigidity and strictness, allowing little room for adapting behavior to circumstances unforeseen, and possibly unforseeable, to the regulators.

2.11 The Zeno Argumentation Framework

After completing the Pleadings Game, which was a rather complex formal model of argumentation developed principally for theoretical purposes, I felt the need to try to develop a more practical system, one that could serve in the short term as a foundation for software tools helping people to argue and deliberate. At this time, the World Wide Web was new and spreading rapidly. In our research group at GMD, the German National Research Center for Computer Science, we hit upon the idea of developing a web-based platform to support public deliberations about political and governmental issues, founded on argumentation theory. This was a number of years before eDemocracy and eParticipation had become established as important research topics.

It was clear from the beginning the a major obstacle would be usability. The target users were ordinary citizens, not experts in formal logic, argumentation theory or related topics. We were aware of relatively simple, informal models of argument, such as Toulmin diagrams [161] and Conklin's graphical implementation [39] of Rittel's "issue-based information system" (IBIS) model of argument [144]. Our aim was to develop a formal model of argument which is as simple to use and understand as these informal models, but which is able to provide users with useful feedback about whether or not some statement at issue is logically well supported by the given arguments.

[6] Lessig used the term 'norm' only for social conventions and moral norms, in contrast to laws and regulations. In philosophy, 'norm' is a broader term, also covering laws and regulations.

Intuitively, a decision to accept a position which is well supported can be justified and explained, using the given arguments. A decision to accept a position which is not well supported may still be a good decision, but it may appear arbitrary, subjective or irrational.

In the intended application context, where many people are deliberating some issue together, the ability of the system to inform users about which positions are currently well supported has the additional advantage of helping users to allocate their investigative and reasoning resources. One need not invest further effort in defending a position which is acceptable. The burden is first on the opponent's of this position to find counterarguments. In the meantime we know from research by the eParticipation community that one of the main obstacles is to motivate citizens to participate. A system which helps to make transparent which positions are winning or losing should help motive stakeholders to participate in order to defend their interests. This feature could contribute to transforming an eParticipation platform from a content management or hypertext system into an exciting playing field for public debate.

Whereas my previous "Computational Dialectics" article [69], discussed in Section 2.10, presented the motivational background and an initial high-level sketch of the design of the Zeno system, "The Zeno Argumentation Framework" [77] is a technical article presenting the formal model designed to meet the objective of an easy-to-use system, comparable to IBIS, but with the ability to compute the acceptability of alternative positions in a deliberation, given the arguments which have been put forward at this stage of the dialogue.

We called the model an "argumentation framework" following Prakken [133], who used this term to name the component of an argumentation system responsible for computing the status of the arguments which have been put forward at a particular stage of a dialogue. At about the same time, Dung used the term to mean a pair $\langle \mathcal{A}, \mathcal{D} \rangle$, where $\mathcal{A}$ is a set of arguments and $\mathcal{D}$ is a binary relation on arguments such that $(a_1, a_2) \in D$ iff a_1 'attacks' a_2 [44]. Dung's formal model defined the notion of the 'acceptability' of an argument in such an argumentation framework. Thus, Prakken's and Dung's conceptions of argumentation frameworks seem quite close; both can be used to determine the status of arguments in a finite set of conflicting arguments put forward at a stage of dialogue. In retrospect, perhaps a clearer term would be "argument evaluation structure" or, at more technical, software level, "argument evaluation component".

As it turns out, Dung's model of an argumentation framework has become one of the most influential models in the computational models of argument field. Perhaps this is due to its elegance and simplicity. It is hard to imagine how a binary attack relation on fully abstract arguments, with no further structure, could be reduced further. Zeno was intended to be simple. Our starting point, Rittel's informal IBIS model of arguments is also a very simple model of argument, but nonetheless somewhat richer than Dung's model. The elements of IBIS are issues, positions, and pro and con arguments. Issues represent problems. Positions are proposals for resolving an issue. Pro and con arguments represent propositions which speak for or against, respectively, some position. As a special case, an issue can be whether or not some proposition is true, in which case there are usually only two positions, yes and no.[7] Arguments can raise further issues, for example regarding whether or not the premises of the argument are true. IBIS models have the nice property of being easy to visualize, as directed graphs. Perhaps for this reason they have been quite successful in practice. Several of the leading argument mapping software packages have been based on IBIS [106].

IBIS was intended to support deliberations about alternative courses of action and thus seemed a reasonable starting point for our research, since our aim with Zeno was to support public deliberations about city planning and other political issues. Our challenge was to try to find a way to formalize the acceptability of positions in an IBIS graph. We achieved this by extending each issue of an IBIS graph with a set of qualitative constraints on the strengths of the arguments pro and

[7] Other positions are conceivable for issues about the truth of some proposition. Consider a loaded question, such as "Have you stopped beating your wife?". Here an appropriate position would express the proposition that that you have never beat your wife in the first place.

con the positions of the issue. Inspired by work by Farley and Freeman [54], proof standards were formalized for various ways to aggregate these qualitative constraints. Similar to the way it was possible in the Pleadings Game to construct arguments about rule priorities, we provided Zeno with the ability to argue about the qualitative constraints on the relative strengths of arguments with the same model of argument used for other issues.

Despite their differences, Zeno and Dung's model of an argumentation framework have something in common. They both, at about the same time, were the first algebraic models of argument evaluation to my knowledge which did not conceive of this task as an application of formal logic. Prakken has been a leading proponent of the use of defeasible logics for argument evaluation [133]. This was also the approach I took in the Pleadings Game, using the defeasible logic of conditional entailment [56]. Just as Dung's model abstracts from logic and defines argument acceptability directly in terms of attack relations, in Zeno the acceptability of positions was defined using proof standards and qualitative constraints to aggregate arguments.

The Zeno project continued for a number of years after the publication of this article and produced an eParticipation platform for the World Wide Web which was successfully piloted in several German cities, including Sankt Augustin [?], Esslingen [121], Hamburg [118] and Berlin [95]. In 1999, Zeno won the best exhibit prize at the 23rd German Annual Conference for Artificial Intelligence (KI99) in Bonn, Germany. The final research prototype version of the Zeno system was presented at a European eGovernment conference in 2002 [82]. A commercial version of Zeno is now available under the name of 'Dito'[8]. All versions of Zeno continued to support the IBIS model of argument, but unfortunately the model of argument evaluation developed in "The Zeno Argumentation Framework" did not survive in these later versions, due to the usability challenge we knew from the beginning would be problematical, but in the end we failed to meet. We were not able to find a way to enable ordinary citizens, with no training whatsoever, to enter and understand the qualitative constraints on arguments. Even the basic IBIS model is too challenging for most users in eParticipation application scenarios. IBIS has been most successful in small, meeting room contexts, where the IBIS graphs are entered by a trained facilitator.

2.12 The Carneades Model of Argument and Burden of Proof

Carneades is my latest argumentation system. My ambition for Carneades is a system which supports all the argumentation tasks shown in Figure 1.2, including argument construction from knowledge bases, argument evaluation, argument visualization and support for applying argumentation protocols and managing commitments in dialogues.

In "The Carneades Model of Argument and Burden of Proof" [81], work done with Doug Walton and Henry Prakken, we presented a new algebraic structure for argument evaluation, building on our prior experiences with the Pleadings Game and the Zeno Argumentation Framework. Again, the function of argument evaluation is to determine the status of arguments and statements in a particular stage of a dialogue, given the arguments which have been put forward by the parties thus far. Prior work has used logic, usually defeasible logic, for this purpose [133]. The information state of a stage of dialogue is modeled in this approach as a set of formulas together with, in some defeasible logics, a set of defeasible inference rules. Notice that defeasible logics were not designed for this purpose, and thus do not provide any direct way to model other information which may be relevant for evaluating arguments in a stage of a dialogue, such as which propositions are at issue and, for each issue, the allocation of the burden of proof to some party and the applicable proof standard. The Carneades argument evaluation structure was designed explicitly for this purpose.

Whereas Zeno started with Rittel's issue-based information system (IBIS) conception of argument, in Carneades we collaborated with Doug Walton, one of the leading argumentation philoso-

[8] http://www.ontopica.de/

phers [170, 173], with the aim to develop a formal, mathematical model of argument evaluation informed by the state-of-the-art in the philosophy of argumentation. Wherever possible, we used the terminology and conceptual models of argumentation theory in philosophy.

This effort is evident in the formal model of a single argument in Carneades, as a tuple $\langle c, d, P \rangle$, where c, the conclusion of the argument is a proposition, $d \in \{pro, con\}$ is the direction of the argument, and P is a set of premises. Other computational models of argument have reconstructed arguments as abstract entities [44], proof trees [133] or as sets of assumptions [27], which is also the approach of conditional entailment [56], used in the Pleadings Game [67]. But a simple tuple linking a set of premises to a conclusion comes closest to the conception of argument in philosophy.

Another important concept in the philosophy of argumentation is the notion of an argumentation scheme. These are conventional patterns of argument. Particular arguments are instantiations of these patterns. Argumentation schemes are used for several purposes. When reconstructing arguments from natural language texts, which is an interpretation task, argumentation schemes serve a heuristic purpose. On the assumption that a set of argumentation schemes are accepted as conventions by the community to which the author of the text belongs, these schemes can be used to create hypotheses about the intended meaning of the text, guiding the search for a coherent interpretation. When evaluating arguments, after they have been reconstructed, schemes can be used to reveal missing premises. Keep in mind that most arguments in natural language texts are 'enthymemes', that is they leave one or more premises unstated for rhetorical and efficiency reasons. For example, premises which the author believes are common knowledge, already accepted by the intended audience as being true, are often left implicit to avoid being pedantic or verbose. Only after these premises have been revealed, using the argumentation schemes, can one question whether or not they are in fact true. As will be discussed in more detail in Section 2.16, argumentation schemes also serve as *methods* or procedures for constructing arguments fitting the pattern. That is, they are useful not only for *reconstructing* or evaluating arguments made by others, but also for inventing or *constructing* arguments to put forward in dialogues in the first place. Some have viewed argumentation schemes as inference rules, taking the position that a general, universal method for constructing arguments from schemes is to instantiate them, by substituting their schema variables with constants denoting objects in the domain of discourse. While some argumentation schemes can be viewed this way, this conception is not sufficiently general to capture all methods of constructing arguments using schemes.

Finally, argumentation schemes play a role in allocating the burden of proof in dialogues. Each argumentation scheme includes a set of 'critical questions' which can be asked to challenge arguments constructed using the scheme. For example, in the scheme for arguments from expert witness testimony, there are critical questions for challenging the credibility and reliability of the witness, among others. In the philosophy literature on argumentation schemes, there are two competing theories of the effect of critical questions on the burden of proof, called the 'backup evidence' theory (BE) and the 'shifting burden' theory (SB). According the BE theory, the party who asks the critical question has the burden of producing an argument to support the claim implied by the question. For example, if a party questions the credibility of a witness, the BE evidence theory would require the party to produce an argument with evidence that the witness is not credible. The SB theory, on the other hand, would shift the burden back to the party who made the argument from expert witness testimony to produce another argument with evidence that the witness is credible.

Walton's position on this issue, which we have followed in Carneades, is that whether or not the BE or the SB theory applies depends on the critical question. More precisely, since argumentation schemes are reasoning conventions (norms) of a community, the affect on the burden of proof of each critical question is an issue that needs to be addressed by the community when developing or drafting these conventions. Which party should bear the burden of proof is a policy issue which can only be resolved using knowledge of the domain and the values of the community, considering risks, costs and benefits. For example, given uncertainty about the credibility of a witness, who should bear the burden of establishing the witness's credibility, the party who called the witness or the party who questions his credibility? Given the values of the community, would the efficiency benefits of assuming witnesses to be credible unless there is evidence to the contrary outweigh the costs of occasionally believing witnesses who in fact lack credibility, but succeed in hiding this fact?

The two kinds of critical questions have been modeled in Carneades as additional premises and distinguishing between three types of premises: ordinary premises, exceptions and assumptions. When an argumentation scheme is instantiated to produce an argument, its critical questions become exceptions and assumptions of the argument. Critical questions which cause the burden of proof to shift back to the party who made the argument become assumptions, and critical questions which require the party who asks the question to produce an argument supporting the claim implied by the question become exceptions. Enthymetic arguments leave some premises of an argument implicit. The exceptions and assumptions of an argument, representing its critical questions, are likely candidates to leave implicit. One effect of the speech act of asking a critical question would be to reveal the exception or assumption for the critical question asked.

The way arguments are evaluated in Carneades makes use of the distinction between the three types of premises to allocate the burden of proof appropriately, implementing the policy specified by the argumentation schemes. The goal of argument evaluation is to determine the acceptability of statements in a particular stage of a dialogue.[9]

A stage of dialogue is a tuple $\langle \mathcal{G}, \mathsf{status}, \mathsf{proof\text{-}standard}, > \rangle$, where $\mathcal{G}$ is a set of arguments, status is a function mapping statements to one of $\{\mathsf{stated}, \mathsf{questioned}, \mathsf{accepted}, \mathsf{rejected}\}$, $\mathsf{proof\text{-}standard}$ is a function mapping statements to the proof standard applicable to the statement, and $>$ is a partial order on arguments, for ordering arguments by the 'strength' or 'weight' assigned them by the relevant audience.[10]

The acceptability of statements in a stage is defined recursively. A statement is acceptable if it *satisfies* its proof standard in the stage. Proof standards aggregate pro and con arguments. Only *defensible* arguments are considered during the aggregation process. An argument is defensible only if all of its premises *hold*.[11] Finally, we end up where we began: whether or not a premise holds depends on its type (ordinary, assumption, exception), and the status or, recursively, acceptability of its statement.

Argument graphs are restricted, by definition, to finite, acyclic graphs. Thus the acceptability of a statement in a stage is decidable. The restriction against cycles is not severe. Dialogue protocols usually prohibit putting forward arguments which create such cycles. And pro and con arguments with the same conclusion do not introduce cycles.

We first modeled proof standards in Zeno [77], following Farley and Freeman [54]. In [81], we illustrated how to define proof standards for Carneades with three examples, called scintilla of the evidence, best argument, and dialectical validity. Scintilla is satisfied iff there is at least one defensible pro argument. The best argument standard is satisfied iff there is at least one defensible pro argument which is stronger than any defensible con argument. Finally, the dialectical validity standard is satisfied iff there is at least one defensible pro argument and no con argument is defensible. An open research question is how best to model three important legal proof standards: 1) preponderance of the evidence, 2) clear and convincing evidence, and 3) beyond a reasonable doubt. These have typically been interpreted probabilistically, although it is implausible that judges and juries in fact use probability theory, even implicitly, when applying these standards.[12]

[9] Statements are called 'claims' by some authors [23], following Toulmin [161]. We prefer the more neutral term 'statement', since arguments can be made hypothetically, to determine the effect the argument would have if it were to be put forward, without claiming that the conclusion or premises of the argument are true.

[10] In [?], we organized this a bit differently, by defining a 'context' to be a tuple consisting of all the elements of a stage, except the set of arguments. The acceptability of a statement was defined relative to a context and a set of arguments, rather than, as here, relative to a stage. This new approach is simpler and makes the relationship of argument evaluation to dialogues even clearer.

[11] The conclusion of a defensible argument can still be rebutted by other arguments.

[12] In the meantime, I've developed an initial mathematical model of these legal proof standards [86].

Carneades' "relational core", stripped of its support for multiple proof standards and allocating various kinds of burden of proof, was shown by Prakken in [81] to be very similar to the ambiguity-blocking variant of Defeasible Logic (DL) [124]. Governatori [87] investigated the relationship between DL and Dung's abstract argumentation framework [44]. He proved that the ambiguity-propagating variant of DL instantiates Dung's grounded semantics, for a version of DL without strict rules. He also proved this result for the ambiguity-blocking variant of DL, but to obtain this result he had to change Dung's notion of acceptability of an argument with respect to a set of arguments. Prakken conjectures that, because of its ambiguity-blocking character, the relational core of Carneades cannot be proven to instantiate any of the four semantics of Dung's abstract argumentation framework without changing Dung's notion of acceptability.

The question is what to make of this inability to view Carneades as an instantiation of an abstract argumentation framework in Dung's model. Dung's model is currently dominant in the computational models of argument field. It is viewed by many as setting the standard. Our work with Carneades raises doubts about the validity of Dung's abstract argumentation frameworks as a model of argument evaluation in dialogues. Dung did not design abstract argumentation frameworks for this purpose, and validated them in his seminal paper with examples from game theory, the stable marriage problem, and reconstructions of various nonmonotonic logics. None of these are examples of argumentation in dialogues. Carneades is based on the state-of-the-art of argumentation theory in philosophy which, although informal, has been developed through a methodical analysis of real-world arguments and dialogues. Many in the computational models of argument community view argumentation more abstractly, as a kind of reasoning with uncertain and incomplete information, which can take place both within and without dialogues. But this more abstract conception threatens to collapse the distinction between logic and argumentation.

2.13 Pierson vs. Post Revisted

The Pierson vs. Post case [1] has become an important benchmark in the field of AI and Law for computational models of argumentation. In [13], Bench-Capon used Pierson vs. Post to motivate the use of values and value preferences in his theory-construction account of legal argument. And in a more a recent paper by Atkinson, Bench-Capon and McBurney [9], it was used to illustrate a formalization of an argumentation scheme for practical reasoning. In "Pierson vs Post Revisted" [84] we offered yet another reconstruction of Pierson vs. Post, using our Carneades argument evaluation structure, both to help validate Carneades and compare it to this prior work.

Pierson vs. Post is a classic property law case, widely used in legal education. Don Berman and Carole Hafner were the first in the Artificial Intelligence and Law community, to our knowledge, to use Pierson vs. Post, and related well-known property cases, as part of their research on the role of teleological reasoning in the law [21, 91]. In a special issue of the Artificial Intelligence and Law Journal, in memory of Donald Berman, several articles presented models of telelogical reasoning using the Pierson vs. Post case as a benchmark, including a paper by Bench-Capon [13]. Since then, Bench-Capon and his colleagues have continually made use of Pierson vs. Post as a testbed for their research on legal argumentation, including the 2005 ICAIL paper with Atkinson and McBurney [9].

This is not the place to present our reconstruction of the arguments of the case. Let us refer you the original paper for that. Rather, for present purposes it should suffice to summarize our results. We claim that, using Carneades, we were able both to capture the structure of the arguments of the court, at a high level of abstraction, and to evaluate these arguments automatically. The result of this evaluation is compatible with the decision of the court; the judgment of the court appears acceptable given the arguments in the opinion. This does not mean that the decision is necessarily correct or beyond criticism. On the contrary, the model, and also its visualization, helps us to understand the arguments in the opinion and to reveal their weaknesses.

Our reconstruction of Pierson vs. Post in Carneades helps to make clearer some differences in scope and purpose of prior work on computational models of argument which also made use of this case, in particular work by Bench-Capon in [13] as well the work by Atkinson, Bench-Capon and McBurney in [9].

In [13], Bench-Capon's primary concern is to analyze the role of telelogical reasoning in legal argument, motivated by the seminal paper by Berman and Hafner [21], which identified limitations of the HYPO approach to case-based reasoning in the law [7]. Bench-Capon's central idea in [13] is that the rules and rule preferences cannot be derived solely from factors in precedent cases, but must also be informed by the purposes of the rules, i.e. by the values promoted by the rules. Shortly thereafter, Bench-Capon, in collaboration with Sartor, developed this basic idea into a theory-construction model of legal argument [20]. In this model, legal theories are constructed from precedent cases in a process which takes values and value preferences into consideration to derive and order rules, which may then be applied to the facts of cases to reach decisions.

The paper by Atkinson, Bench-Capon and McBurney [9] views legal reasoning as a kind of practial reasoning, following [88], and illustrates this view using Pierson vs. Post. Towards this end, an argumentation scheme for practical reasoning is developed and applied to model a simulated dialog among four agents, based on the facts and arguments in Pierson vs. Post.

In our view, each of these papers used Pierson vs. Post to illustrate computational models of particular argument schemes, rather than attempting to provide a general framework which can accomodate all the argumentation schemes actually used in the case. But legal argumentation has in common with argumentation in general the application of a large variety of argumentation schemes. Our reconstruction of Pierson vs. Post in Carneades illustrates how a variety of argumentation schemes can be used together in a single case.

2.14 Constructing Arguments with a Computational Model of an Argumentation Scheme for Legal Rules

One problem with the Pleadings Game, due to its use of conditional entailment [56], relates to the way arguments are modeled as set of propositions, rather than as premise-conclusion pairs. In this approach, applying an argumentation scheme, such as the scheme for arguments from rules, does not directly result in a argument, but rather only takes part in the definition of a default theory from which arguments can be constructed. Arguments are constructed from the assumptions introduced by the set of *all* argumentation schemes which have been instantiated, similar to the way explanations are abduced from hypotheses. In this approach, there is a many-to-many relationship between argumentation schemes and the arguments introduced by instantiating these schemes. This makes it more difficult to critically evaluate arguments using argumentation schemes, for example to identify missing premises, since the relationship between the premises and conclusion of the scheme must be reconstructed somehow the set of assumptions.[13] Not having explicit links between the premises and conclusion of each argument also makes it more difficult to explain or justify decisions. It may be possible to reconstruct this information from dialectical graphs, i.e. the graph of arguments (as sets of propositions) linked by support, rebut and defeat relations [57, p. 155–162], but this was not attempted in the Pleadings Game.

In "Constructing Arguments with a Computational Model of an Argumentation Scheme for Legal Rules" [74], I developed a computational model of a scheme for arguments from rules which overcomes these weaknesses of the Pleadings Game. Arguments are instantiations of argumentation schemes. Although there are many kinds of argumentation schemes, with many different methods for constructing arguments which instantiate the patterns of the schemes, one common method is to use the patterns as inference rules and construct arguments simply by replacing variables in the rules by constants denoting objects in the domain of discourse. The model developed in this

[13]Which schemes have been applied, on the other hand, is apparent from applicability assumptions in the argument.

paper exemplifies this inference rule approach. Argument from rules is interpreted to be a *class* of argumentation schemes, rather than a particular scheme. Each rule is an instance of this class, modeling a single argumentation scheme.

The language for specifying rules is very similar to the language developed for the Pleadings Game: rules are reified (represented by terms in formulas), with support for explicit exceptions and exclusionary rules. Priority rules for resolving rule conflicts, implementing principles such as *lex superior* and *lex posterior* can be represented. The applicability of rules can be reasoned about. For example, the rules which can be used to support some proposition can be inferred. An argument from a rule can be undercut by questioning the validity of the rule. In addition to exceptions, rules can also be conditioned on *assumptions*.

The arguments constructed by instantiating rules in this new system are instances of our Carneades model of argument [81], which takes over the tasks performed by conditional entailment in the Pleadings Game, such as checking whether some proposition is acceptable given the arguments which have been put forward in a stage of a dialogue. Assumptions and exceptions in Carneades provide a way to model both kinds of critical questions of argumentation schemes, with regard to their effect on the burden of proof, applying either the *shifing burden* or the *backup evidence* theory [81].

Unlike in conditional entailment, specificity is not used to *automatically* resolve conflicts among arguments. Specificity is one property among many which can be used to define priority rules, such as date of enactment or level of authority, but like these other properties cannot be derived by formal means in this model. Users need to assert these properties, or write rules for deriving these properties from other facts. This task need be no more onerous than associating rules with concepts in a taxonomy, as in inheritance hierarchies [49, 162] or overriding methods in subclasses in object-oriented programming. This was the approach we already took in 1987, with Oblog-2 [59]. Loosing the ability of conditional entailment and some other nonmonotonic logics [42, 108] to automatically order rules by specificity is unfortunate, but necessary so long as no way has been found to combine this ability with some way to give other ordering principles priority over specificity when required by application domain.

Still another difference between the Pleadings Game and Carneades concerns the directionality of rules. In the Pleadings Game, the way rules are mapped to material implications in conditional entailment default theories supports contrapositive reasoning. If the conclusion of a rule is not true, the rule can be used backwards to derive that the antecedent of the rule is not true, assuming the rule is valid and applicable. Conditional entailment strengthens classical logic. All inferences sanctioned by classical logic are still valid in conditional entailment, including *modus tollens*, $\{p \rightarrow q, \neg q\} \vdash \neg p$.

In the nonmonotonic logic community, there has been a clash of intuitions about whether or not default rules should support contrapositive reasoning. For example, Simari and Loui intentionally designed their nonmonotonic logic to suppress contrapositive inferences [156]. In the legal domain, Allen has argued that legal rules do not contrapose [5] and prefers relevance logic [6], which is weaker than classical logic, for modeling legal rules for this reason. Prakken agrees with Allen that legal rules do not contrapose but has always shared my view that legal rules allow conclusions to be drawn which are only presumptively true [132]. For this purpose relevance logic is too weak. If one accepts that rules should not contrapose, then a way to model and reason with rules is needed which is both stronger and weaker, in the right ways, than the result of interpreting rules to be material implications in classical logic.

In the Pleadings Game, I restated arguments on both sides of the issue about whether or not rules should contrapose, but had yet to make up my mind about which position to accept [68, pp. 100-102]. Conditional entailment was chosen for the Pleadings Game despite its stance on this issue, because its model-theoretic semantics seemed an important advantage at the time.

In the meantime I have come to share Allen and Prakken's view that rules, in the sense of regulations, should not contrapose. Rules serve their normative function, guiding behavior, by helping people to think about the legal consequences of their actions before they act. For example, the criminal law rule against murder, defined as the "unlawful killing of a human being with malice aforethought", expresses not only a policy against such killings, but also a reasoning policy to presume that a murder has taken place given proof that a human being was intentionally killed. There are exceptions, such as killings in self defense, but the rules are formulated so as to deter killings. Consider the alternative: a rule stating that intentional killings of human beings are lawful, with an exception for killings which are not in self-defense! Logically these alternatives may be equivalent. But the normative effect of these alternative rules on behavior couldn't be more different. Rules must be useful not only for classifying actions or states as legal or illegal, after the fact, but also for helping people to behave in accordance with a policy. Simple general rules, subject to exceptions, make it easier to do the right thing most of the time. Allowing rules to be used contrapositively would undermine the normative purpose of rules. It is already difficult enough to understand the rules well enough to apply them in a forwards direction, without also having to understand how to reason backwards from consequents to antecedents. Not only would rules which need to be applied backwards be ineffective for achieving policy goals, it would seem quite unreasonable to hold people accountable for violations of rules which can only be recognized by applying rules backwards, due to the unnecessary complexity of this task.

In philosophy, argumentation schemes are more like inference rules than material implications. Inference rules cannot be used backwards to derive information about the premises from the conclusion, not even in a calculus for classical logic. Like regulations, argumentation schemes express conventions and norms, grounded in policies balancing risks and benefits, for guiding the behavior of a community. Argumentation schemes, together with dialogue protocols, regulate reasoning behavior.

Given that both rules and argumentation schemes guide reasoning behavior, it seems more natural to model rules as argumentation schemes, as we have done here, than as material implications. But I am tempted to go further by claiming that rules *are* argumentation schemes, not only capable of being modeled using argumentation schemes. Certainly they have much in common, since both are used to express context-dependent conventions and norms of a community.

Conditional entailment's model-theoretic semantics was one of the main reasons for choosing it for the Pleadings Game. But it is this semantics which gives specificity priority over all other principals for ordering rules and sanctions contrapositive inferences. Is there some way to modify its semantics to make it more suitable for practical reasoning? Conditional entailment's semantics is based on Shoham's idea of preferred models [153]. Given a preference order on models, a nonmonotonic consequence relation can be defined in terms of the propositions which are true in the preferred models, rather than in all models. There is an inverse relationship between the number of models of a theory and the number of propositions entailed by the theory. This makes sense, because the greater the number of models of a theory, the more general the theory, i.e. the theory describes only the features all these models have in common.

Propositions which are true in the preferred models, but not all models, are presumably or probably true, but not necessarily true. A rational basis for making such inferences is just what we are looking for in the semantics of a nonmonotonic logic. But where does the preference order on models come from? Why should one prefer one model of some theory over another? Geffner and Pearl sidestep this issue to some extent by modifying Shoham's approach so as to define conditional entailment in terms of all *admissible* preference orderings on models, rather than a particular ordering, where admissibility is defined in an abstract, domain-independent way. Nonetheless, conditional entailment assumes there can be some reason for preferring one model over another.

The source of the preference order on models depends on the purpose of the theory. If the purpose of the theory is prediction, the preference order could represent probabilities. One of the standard examples of nonmonotonic reasoning in artificial intelligence, about birds normally being able to fly, seems to be of this type. If some animal is a bird, then it probably can fly. In theories about rules, in the regulatory sense of interest to us here, the source of the preference order on models is some policy balancing the risks and chances of making errors. For example, consider once again Walton's example about seeing something which looks like a coiled snake in a dark cave [171, p. 52]. Even if it is probably a rope, it may be a good policy to presume that it is snake. By preferring the model in which the coiled thing is a snake, given insufficient information to eliminate this possibility, we are not concluding that it is necessarily a snake, or that it is probably a snake, but only that it makes good sense to presume that it is a snake.

While it may be possible to define a model theory using preferred models for a nonmonotonic logic which overcomes the problems of conditional entailment regarding specificity and contrapositive inferences, a solution to these problems would not be enough to give us a model theory for practical reasoning. Practical reasoning, in our view, is a modeling *process*. Let us suppose, for the sake of simplicity, that this process produces a sequence of theories, $T_1 ... T_n$, where each T_i is a default theory for some nonmonotonic logic. The model theory of the logic tells us, for some issue p, whether or not p is entailed by some theory T_i in this sequence, $T_i \models p$. But such a model theory would tell us nothing about the correctness or acceptability of any of these theories. It provides us with no normative advice as to whether or not the process which produced this sequence of theories is rational. A model theory covering the entire theory construction process is missing. For some issue, p, we would like the model theory to tells us whether it is acceptable to presume p is true at the end of the process, taking into consideration not only the theories constructed but also pragmatic considerations such as the goal of the process and resource limitations.

It is unclear whether the model theory of some nonmonotonic logic would be useful as part of a model theory for practical reasoning, viewed as a process. More generally, it is not clear whether the kinds of mathematical models needed for argumentation have much in common with the models developed for evaluating the correctness and completeness of calculuses for deductive logics. This does not mean that deductive logic and model theory are irrelevant for argumentation. Classical logic, for example, provides some outer bound rationality constraints for argumentation protocols. In an ideal world, with no computational or other resource limitations, the set of propositions accepted or claimed by a participant in a dialogue should be consistent. And the commitments of a party in a dialogue should include not only his explicit claims or concessions, but also some of their logical consequences. But if the classical logic used for these purposes is undecidable, such as full first-order predicate logic, these constraints are of theoretical interest but limited practical utility. While process models for argumentation should consider logical constraints, the overriding goal of these models is to facilitate good, fair and transparent decisions, despite practical constraints.[14]

While we do not reject model theory and would consider it worthwhile to try to discover or invent a model theory which overcomes these problems, we do not believe that a model theory of this kind is essential for a well-founded formal model of argument from rules. The semantics of the rule language we have developed is sufficiently grounded in the dialectical and argumentation-theoretic approach articulated by Ron Loui in his landmark article "Process and Policy: Resource-Bounded Non-Demonstrative Reasoning" [117]. Essentially, we interpret rules as tools for representing policies for reasoning in resource-limited, decision-making processes. The semantics is purely procedural, in line with Rawl's concept of procedural justice [140] and the modern procedural, dialogical perspective of argumentation in philosophy [173]. Despite the expressiveness of the rule language, which would result in an undecidable logic, argumentation protocols are responsible for assuring proceedings terminate in finite time with acceptable, presumptively true conclusions.

[14]Emerson wrote famously "A foolish consistency is the hobgoblin of little minds, adored by little statesmen and philosophers and divines." [46]. Emerson, along with Thoreau and others, was a American Transcendentalist. They believed it can be important in life to follow one's intuitions, even when these intuitions are known to be inconsistent with an established doctrine that one accepts in principal. Although the idea of making decisions intuitively may seem to be in opposition to the idea of making decisions by conscious reasoning, as in argumentation, intuition, like perception, may be viewed as an argumentation scheme. The gut feeling that some proposition is true, gives us one reason to presume the statement is true. Argument from an established doctrine is similar to argument from scientific theories. When arguments from intuition conflict with arguments from doctrine or theories, preferring to follow one's intuition can be the right thing to do.

Rules provide a way to represent a broad class of argumentation schemes, which enable methods from theorem proving and logic programming to be adapted to invent arguments, in a goal-directed, backwards chaining way. A set of rules induces a search space over argument graphs. Constructing arguments with rules can be viewed as heuristic search in this space, looking for argument graphs in which some disputed claim is acceptable or not, depending on the interests of the party. In dialogues, the parties take turns, looking for counterarguments. But in general the space of potential arguments cannot be fixed in advance of the dialogue. Each party can make use of its own set of rules to construct arguments, or other argumentation schemes whose methods for constructing arguments do not make use of inference rules. Arguments from perception, for example, could be constructed with the help of programs connected to sensing devices.

An important feature of our present model of argument from rules, is that it makes use of and illustrates a more general model of argumentation schemes. As discussed in Section 2.16, we have modeled a number of other argumentation schemes have using this framework, including argument from precedent cases and testimonial evidence. The architecture of the system enables any number of diverse methods for constructing arguments, not just inference rules, to be used together to solve problems, resulting in a form of hybrid reasoning system.

2.15 Visualizing Carneades Argument Graphs

Diagramming arguments has a fairly long history, going back to at least Wigmore early in the twentieth century [174]. Wigmore was a lawyer who developed a method for diagramming relationships between various kinds of evidence and factual issues in legal cases. Toulmin's diagramming method in his "Uses of Argument" [161] has been extremely influential. The argument diagramming method developed first by Beardsley [12] and refined by Freeman [53] has become the de facto standard in the humanities.

Conklin's gIBIS model of argument visualization [39], based on Rittel and Weber's idea of an issue-based information system [144], was perhaps the first software designed for visualizing arguments. There are now quite a few software packages available for visualizing arguments, such as Araucaria [141, 148], ArguMed [167], Compendium [151], Rationale [10] and ArguNet [24]. Some of these are commercial products. Argument visualization software is being increasingly used in critical thinking and philosophy courses [97] and for educational purposes in other fields as well, such as law [3, 36, 8].

Almost all of these argument diagramming systems are based on informal models of argument and thus are unable to evaluate arguments to advise users about the acceptability of statements at issue. Verheij's ArguMed system [167], which is founded on Reason-Based Logic [94, 166], is an exception. The commercial Rationale system provides as much support for argument evaluation as may be possible with an informal model of argument. Rationale provides tools to help users to note, track and visualize their manual evaluations in a systematic way. But Rationale is not able to advise users about the acceptability of statements given the arguments in the diagram. In [106], a distinction is made between 'sense-making' systems, which are tools for helping human users to reason and argue, and systems which are able to automatically construct or evaluate arguments, using formal logic and artificial intelligence methods.

In "Visualizing Carneades Argument Graphs" [75] I presented a diagramming method for the Carneades model of argument structure and evaluation, with the goal of integrating the features of these two kinds of systems. The main challenge was to do this in a way which retains the usability and intuitiveness of the leading argument diagramming tools. The paper was first presented at the conference on "Graphic and Visual Representations of Evidence and Inference in Legal Settings" at Cardozo Law School in New York, at the invitation of Peter Tillers. Due to the historical importance of Wigmore diagrams in this community, I also showed how Wigmore diagram's could be emulated using Carneades, without loss of intuitiveness but with the advantage of a formal model enabling the acceptability of statements to be computed.

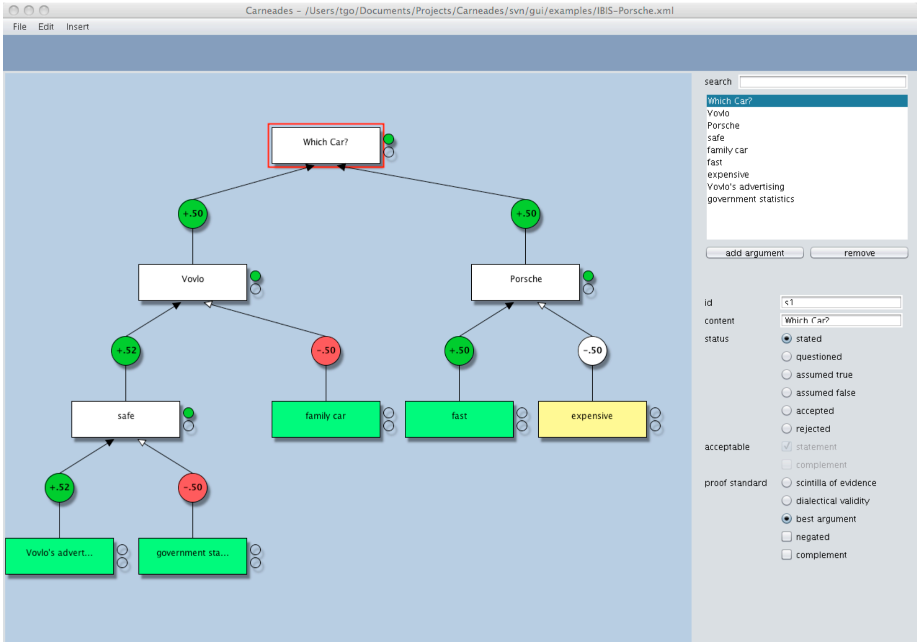

Figure 2.1: Screen Shot of the Carneades Argument Diagramming Application

In the meantime, thanks to a grant from Google as part of its "Summer of Code" program for developing Open Source software, Matthias Grabmair, a PhD student of Kevin Ashley's at the University of Pittsburgh has been working with me to develop an argument diagramming tool for Carneades, based on the work presented in this article. A first version of the program is already available.[15] Figure 2.1 shows a screen shot. The argument graph displayed in the screen shot is a reconstruction in Carneades of the leading Zeno example, from [77], with the arguments put forward by a husband and wife about which car to buy, illustrating one way to construct IBIS style argument graphs using Carneades. Doug Walton has received a grant from a Canadian foundation to work with me on the further development of this diagramming tool for a year. One of our goals will be to integrate this graphical front-end with the components I have developed for generating arguments from ontologies, rules and cases, which were presented in the paper discussed in the Section 2.16.

2.16 Hybrid Reasoning with Argumentation Schemes

In "Constructing Arguments with a Computational Model of an Argumentation Scheme for Legal Rules" [74], discussed in Section 2.14, I explored the relationship between legal rules, argumentation schemes, material implications and inference rules. I argued that legal rules can be viewed as argumentation schemes, since they regulate not only rights and obligations but also the way we reason about them, with a role to play in the way the burden of proof is allocated. Thus, inference rules are more appropriate for modeling legal rules than material implications, since inference rules, as their name suggests, regulate how inferences may be drawn. They can also be formulated so as to support presumptive, defeasible reasoning and avoid unintended contrapositive inferences.

Legal rules are argumentation schemes and can be modeled with defeasible inference rules. Thus some argumentation schemes can be modeled as inference rules. But can all argumentation schemes

[15]http://carneades.berlios.de

be modeled this way? This is the question I addressed in "Hybrid Reasoning with Argumentation Schemes" [76].

Consider Pollock's scheme for arguments from perception [129], which says if something looks like it has some property then one can presume that it does have this property. Can this scheme be modeled as an inference rule? Let's use Pollock's example about things which look red presumably being red. If we already accept or are willing to assume that the thing looks red, then an inference rule formulation of the scheme could be used to derive, in a forwards manner, the presumptive conclusion that it is red. But if someone has claimed that an object is red and our task is to critically assess this claim, then the inference rule formulation of the scheme does not get us very far. While the inference rule could be used backwards to reduce the task of determining whether the object is red to the task of finding out whether or not the object looks red, interpreting the scheme in this way would give us no reason to believe the object is red when the scheme is applied. The resulting argument would be more conditional then we want, saying only that *if* the object looks red we *would* have a reason to presume it to be red.

A more promising interpretation of argumentation schemes, which overcomes this problem, is to view them as prescriptions, methods or procedures for constructing arguments which instantiate the template or pattern of the scheme. That is, argumentation schemes have two parts: 1) an argument pattern or template, which is instantiated to create an argument by substituting variables in the pattern with constants denoting objects in the domain, and 2) a method for binding the variables in the template.

For example, using this new conception of argumentation schemes, the method part of the scheme for argument from perception could be some procedure for actually going out into the world and using sensors to gather data about the object in question. An argument constructed using such a scheme would give the user a reason to believe the object has the perceived property and some confidence the premises are true before claiming them when putting forward the argument in a dialogue.

Another example of an argumentation scheme which cannot be fully understood as an inference rule is the scheme for argument from witness testimony [173]. To be useful for constructing arguments, the scheme must be understood as a prescription for interviewing witnesses. The scheme hasn't been applied if no witness has testified. The scheme is not usually applied hypothetically. Arguments constructed using the scheme do not have the form "P would presumably be the case if there were a witness who testified that P is case." Rather arguments from witness testimony have the form "P is presumably the case since witness w did testify that P is the case.".

Argumentation schemes are often used to support the interpretation task of reconstructing arguments in natural language texts. For this purpose an inference rule or argument pattern conception of argumentation schemes may be adequate. But argumentation schemes are also useful for constructing or inventing arguments to put forward in dialogues. This is not an interpretation task. Rather, at each stage of an argumentation process, usually a dialogue, one task will be to try to find or construct additional arguments which can be put forward to make an acceptable statement unacceptable, or vice versa. Whereas the problem of checking the acceptability of a statement given a finite set of arguments is decidable, the problem of finding or constructing arguments is in general ill-defined, open and undecidable. Since models are abstractions developed for particular purposes and provide only specialized views onto reality, several models may be relevant and useful for an argumentation process, depending on the issues.

For example, a legal case may raise issues requiring argument from precedent cases, rules, policy goals, moral principles, jurisprudential doctrine, social values and evidence. Argumentation tools are needed for helping people to argue their cases effectively in court and administrative proceedings. Most prior work on computational models of argumentation schemes has focused on their role as patterns for classifying arguments and revealing implicit premises, to support the process of argument reconstruction. For example, Aracuaria provides a way to define templates for argumentation schemes and to use these templates to classify arguments and their premises [142].

Recently, an OWL ontology of many of Walton's argumentation schemes has been developed, with the aim of being able to use description logic theorem provers to classify arguments [138]. Tools for reconstructing, visualizing and evaluating arguments, while important, are not sufficient in this context. A party to a legal or administrative proceeding is not in the role of an analyst trying to understand a previous dialogue, but rather in the role of an advocate needing to construct and put forward effective arguments as the dialogue unfolds.

How can arguments constructed from multiple, hybrid models be integrated, aggregated and evaluated? Our thesis is that argumentation schemes enable the methods used to construct arguments to be separated and abstracted from the form and content of arguments. Structures for aggregating and evaluating arguments [44, 81] depend only on relationships between arguments, not the methods used to construct them. This separation makes it possible to use a variety of hybrid methods to construct arguments but to aggregate and evaluate these hybrid arguments using a common structure. Each argumentation scheme, in its role as a method, implements a protocol for mapping an issue and a model of some information or knowledge to a set of arguments about the issue. A set of such argumentation schemes induces a search space over sets of arguments. Heuristic methods can be used to search this space for sets of arguments in which some goal statement would be acceptable, using the common argument evaluation structure.

The main result of this paper is a formal protocol for argumentation schemes, in their role as methods for constructing arguments, which enables methods implementing diverse schemes to be used together in an argumentation process. We use the term *argument generator* to mean the argument construction method of an argumentation scheme. A set of argument generators induces a search space. Each argument produced by an argument generator can be used to construct a successor state in the space. The space can be searched heuristically for states in which the statement at issue is either acceptable or not, depending on whether the goal is to prove or disprove the statement.

The protocol which argument generators must implement can be formalized as a function type. Let $\mathcal{L}$ be a language, i.e. a set of formulas denoting propositions in some domain. Let $\mathcal{A}$ denote the set of arguments which can be formed using formulas from $\mathcal{L}$. A *stage* in an argumentation process is modeled as a tuple $(\mathsf{arguments}, \mathsf{assumptions}, \mathsf{issue})$, where $\mathsf{arguments} \in \mathcal{P}(\mathcal{A})$, $\mathsf{assumptions} \in \mathcal{P}(\mathcal{L})$, and $\mathsf{issue} \in \mathcal{L}$. An argument generator is a function mapping a stage to a (possibly infinite) sequence of arguments.

Intuitively, a stage represents the current state of the argumentation process and the task of an argument generator is to find or construct arguments which would alter the acceptability of the issue, if the argument would be put forward by adding the argument to the arguments of the stage, assuming the premises of the argument are true. If the statement at issue is acceptable in the stage, the task of the argument generator is to find counterarguments which would make the statement unacceptable, and vice versa.

The paper illustrated this conception of argumentation schemes as methods for generating arguments with computational models of four schemes, for arguments from rules, ontologies, cases and witness testimony. The scheme for arguments from rules was published previously, in [74]. The scheme for argument from ontologies was modeled by mapping the Description Logic Programming [90] dialect of description logic [11] to rules. The scheme for argument from cases is based on Wyner and Bench-Capon's reconstruction [176] of Aleven's CATO model of case-based reasoning by analogy [3]. The model of the scheme from witness testimony is like the dialog component of a classical expert system: it asks questions and stores the answers in a database, using forms to structure the dialogue so that related questions are asked in a coherent way.

In artificial intelligence, the *blackboard architecture* for hybrid knowledge-based systems, as first implemented in the Hearsay-II speech understanding system [48], provides a way for multiple inference engines to work together on solving a problem. Each inference engine uses its own *knowledge source*, modeled in whatever way is appropriate for its particular reasoning methods. In blackboard systems, the inference engines collaborate by writing statements to a shared data structure, called the 'blackboard'. In Hearsay-II, the statements represent hypotheses about the utterances being interpreted. Inference engines use the statements on the blackboard as input to their reasoning methods, in a forward-chaining, data-driven way. Whenever sufficient data is on the blackboard for some reasoning method of an inference engine to be applicable, the inference engine announces this to a scheduler. If several inference engines have applicable methods, the scheduler decides in which order to apply the methods. The inference engines can derive conflicting conclusions. Hearsay-II provides some way to weigh or order inference engines to resolve these conflicts.

Later blackboard systems, such as Walker's Expander legal expert system [168], recorded not only statements on the blackboard, but also *justifications* for these statements, what we would now call arguments, using a reason-maintenance system [43, 41] to manage dependencies between statements. As the inference engines continue to work on problems and post further information to the blackboard, the reason maintenance system would update the status of the statements on the blackboard, labeling them 'in' or 'out'.

Clearly there are similarities between our approach to hybrid reasoning using argumentation schemes and blackboard systems. The role of inference engines is played by argument generators. Argument evaluation structures can be viewed as providing reason-maintenance services. There are however significant differences. Firstly, argument generators are not problem solvers. They do not implement problem-solving methods with their own control strategies. Rather, a set of argument generators induces a space of sets of arguments which can be searched using a centralized search strategy. Secondly, whereas blackboard systems forward chain from the statements on the blackboard, our approach allows the space of arguments to be searched in a goal-directed way.

As Walker notes, any architecture for integrating hybrid reasoners requires a formal language for expressing statements which is "powerful enough to express the input to and output from any of the knowledge sources" [168, p. 76]. Our approach places few restrictions on the language used for expressing statements in argumentation schemes, requiring only equality and complement operators. The problem of translating between the languages used by different problem-solving methods is the focus of a recent publication by Henry Prakken [135], in which he develops a model of 'I/O transformers' between problem-solvers, and illustrates this method with transformers for first-order logic, Bayesian probability theory and two versions of default logic. We speculate that such I/O transformers can be reconstructed as argument generators in our framework, but this needs to be validated in future work.

Finally, I would like to mention the Babylon system [37], developed by the German National Research Center for Computer Science (GMD). Babylon is an expert system 'shell' integrating inference engines for production rules, Horn clause logic (as in Prolog), frames (i.e. a form of object-oriented programming), and a constraint satisfaction system. Bablyon's approach to hybrid reasoning consists of an extensible meta-interpreter, which manages a set of problem-solving tasks and dispatches tasks to inference engines based on the structure of the task and meta-knowledge about the kinds of task each inference engine can handle. Babylon's main achievement was in finding a way to integrate a number of high-level programming paradigms, taking into considering their various operational semantics. In contrast, our system is designed to integrate methods for constructing arguments from diverse argument sources, such as legislation, precedent cases and testimonial evidence.

Some research in the artificial intelligence and law field has addressed ways of integrating reasoning with rules and cases [55, 29, 157, 137] and ways to resolves conflicts among arguments, such as prefering arguments from cases to arguments from rules [55, 29, 157]. Our work aims to generalize these results by providing an open, extensible architecture for integrating models of any argumentation scheme.

All the models of argumentation schemes presented in the paper have been implemented in Carneades, as part of the European ESTRELLA project, and used to build a number of demonstrators in the legal domain. Carneades is freely available on the Web, as Open Source software.[16]

The demonstrators of the ESTRELLA project are prototypes of expert systems for helping citizens to apply legislation in order to assess their legal rights and obligations. Most deployed legal expert systems are currently built using rule-based systems technology from the 1980s. While such systems have proven their usefulness for supporting the processing of complex claims in public administration as well as the private sector, for example in the insurance industry, they are based on the simplifying assumption that the relevant laws and regulations can be adequately modeled as a logical theory. Claims assessment is viewed as deduction, in which a theory is applied to the facts of the case to deduce legal consequences. Lawyers have long understood that in general legal reasoning cannot be reduced to deduction in this way. Rather, legal reasoning generally involves the iterative construction and comparison of alternative theories of the facts and the law, interpreting both the evidence and the relevant legal sources, in an argumentative process.

Our aim in modeling argumentation schemes is not fully automatic reasoners, but rather tools which can help people to construct a wide variety of arguments, improving their ability to protect their interests in dialogues, especially in the legal domain.

[16] http://carneades.berlios.de

Chapter 3

Conclusions and Future Work

To conclude, let me return to the use cases shown in Figure 1.2 and discuss how the work reported in the submitted publications has contributed to a better understanding of each of these tasks and the kinds of computational models and software tools which are needed to support them. Whereas Chapter 2 presented the submitted publications in chronological order, here the contributions are grouped together on a task-by-task basis.

Starting at the logical level, my early work focused on overcoming the problems of representing knowledge for practical reasoning, especially in the legal domain, in first-order logic. An analysis of legislation revealed that legislation is typically organized as general rules subject to exceptions of various kinds together with legal principles such as *lex superior* exist for resolving conflicts among arguments constructed from rules. [58]. This legal analysis led me to believe that Horn clause logic, used by the logic programming language Prolog, was not adequate for modeling legal rules [60]. Oblog-2 was my first attempt to develop a knowledge representation language suitable for representing legislation in a more isomorphic way, preserving the structure of general rules subject to exceptions [59]. This work predated and influenced research by others on isomorphism in the field of AI and Law [146, 105, 15].[1] In Oblog-2, an inheritance hierarchy of concepts was used to order rules by specificity, in accordance with the principal of *lex specialis*. But other ordering principles were not supported. The rule language of the Argument Construction Set [61] introduced an 'unless' operator, for modeling rules with explicit exceptions. In the Pleadings Game this rule language was extended and given a model-theoretic semantics based on the nonmonotonic logic of conditional entailment [67]. But since this logic sanctioned contrapositive inferences and gave more specific arguments priority over all others, such as *lex superior*, even though the relative priority of these principles is not universal but rather domain-dependent, this model-theoretic approach was abandoned in my most recent work on Carneades, which interprets rules as schemes for constructing arguments [74]. Finally, this work on argumentation schemes was generalized to cover arguments from terminology (ontologies), precedent cases and testimonial evidence, in addition to arguments from rules [76, 85].

In Artificial Intelligence one goal of research on knowledge representation has been to finds ways to represent knowledge in a declarative, task-independent way. Knowledge representation formalisms based on first-order predicate calculus, such as Horn clause logic and description logic, have been proposed for this purpose. One of the main results of my research is the realization that it is often useful to represent knowledge in a way which is more task dependent, using for example general rules and exceptions. Such rules are not declarative statements about what is probably or usually true in the world, but rather are argumentation schemes which provide guidance about how to reason in certain practical contexts, taking into consideration an assessment of the risks and

[1] Recently, in collaboration with Trevor Bench-Capon, I have returned to the subject of isomorphic modeling of legislation [19].

costs of errors, as well as allocating the burden of producing evidence to the party, the proponent or opponent of the argument, which can produce the evidence at least cost.

Moving up to the dialectical level of the use-case diagram, having been inspired by legal philosophers such as Hart and founders of the field of computer science and law, such as Bing and Fiedler, I realized early [58, 60] that practical reasoning, as exemplified by legal reasoning, cannot be reduced to deduction. As Hart noted [98], legal concepts are 'open-textured' and, as noted by Bing [25] and Fiedler [50], the application of legal concepts to the concrete facts of particular cases is a modeling process in which theories and arguments are constructed together. My contribution was to develop more precise computational models of this modeling process, along with prototypes of software tools for helping people to build and use these models to make and justify practical decisions.

My design of the Argument Construction Set [61], implemented by Schweichhart [150], was one of the first interactive software tools for constructing theories and arguments. It was developed at about the same time as Conklin's influential graphical IBIS system [39], which was based on Rittel's theory of wicked problems.[2] The Argument Construction Set made use of a truth-maintenance system [43] to manage dependencies between arguments and claims.

My next system, for spotting issues by searching interpretation spaces [63], replaced this truth maintenance system with an assumption-based truth maintenance system (ATMS) [41]. Defeasible reasoning was modeling using Poole's assumption-based approach [130], which matched the ATMS well. In "An Abductive Theory of Legal Issues" [64] I refined this model of theory construction by abstracting from the technical details of reason maintenance systems to a mathematical conception of abduction and applied the resulting model in a computational reconstruction of Hart's concept of clear cases. Subsequently I applied this theory construction model to the practical problem of assembling legal documents [65]. I retained this approach to managing dependencies in the Pleadings Game, but developed a more efficient but less powerful version of the ATMS, called the Minimal Reason Maintenance System [68, p. 182] sufficient for our purposes.

The Zeno argumentation framework [69, 78] included a new kind of reason maintenance system, using a mathematical model of *dialectical graphs* based on Rittel's conception of an Issue-Based Information System (IBIS).[3] These dialectical graphs consisted of issues, positions, arguments pro and con the positions, and qualitative constraints over the relative strengths or weights of the arguments. These constraints could be argued about like any other issue. Whereas most of my prior research was focused on supporting legal reasoning in persuasion dialogues, Zeno was designed primarily to support deliberation dialogues in which groups of people collaborate by proposing solutions to practical problems and then evaluating the pros and cons of these proposals. Zeno was later applied to support dialogues about city planning issues on the Internet, between citizens and government in e-participation and e-democracy projects [79, 83, 82]. In 1999, Zeno won the best exhibit prize at the 23rd German Annual Conference for Artificial Intelligence (KI99) in Bonn, Germany. In the meantime, Zeno has been become a commercial product, marketed by the name 'Dito'.[4] Recently, in my work with Doug Walton on Carneades, I have developed yet another kind of reason maintenance system, called argument graphs, based on Walton's philosophy of argumentation [81]. Because of the way the model uses pro and con arguments to support nonmonotonic reasoning, my model of argument graphs has more in common with Doyle's original truth-maintenance system than with de Kleer's assumption-based truth-mainteance system. But the model goes beyond Doyle's model due to the way it allows the burden of proof to be distributed using different kinds of premises (ordinary, exceptions and assumptions) and because of its support for variable proof standards.

[2] Interestingly, Rittel, a city planner, had antipostivistic insights which were quite similar to those of legal theorists such as Hart, Bing and Fiedler, presumably because city planning and legal decision-making are both subspecies of practical problems.

[3] The "Zeno Argumentation Framework" [78], written with Nikos Karacapalidis, is probably my most cited work, with about 200 citations currently listed on Google Scholar, `http://scholar.google.com`.

[4] http://www.ontopica.de

The Pleadings Game was my main effort thus far to model the procedural aspects of argumentation dialogues. Whereas my previous models of argument construction were intended for use in tools which help individuals to construct and explore alternative theories of a case, the Pleadings Game was the first of my models to cover the collaborative construction of competing theories by the parties in a proceeding. As its name suggests, the model focused on the pleadings phase of a legal court proceeding, where the goal is to identify, not resolve, the factual and legal issues to be put before the court in the trial. The model was original and unique in many ways, due in part to the type of dialogue modeled. Whereas prior efforts to model argumentation dialogues aimed to provide a proof theory for intuitionistic logic [116], model information-seeking dialogues [96, 120, 101], model persuasion dialogues with defeasible rules [143], or to support interactions with expert systems [16], the Pleadings Game was the first system to model argumentation as a collaborative theory-construction process. The parties in a legal dialogue construct alternative theories of the case. Essentially, the factual and legal issues to be resolved by the court are the relevant differences between these theories.

Finally, my work on document assembly and argument visualization can be viewed as modest contributions to the rhetorical level. In a "Theory Construction Approach to Legal Document Assembly" [65] I applied my then current theory construction model of legal reasoning in a design of a new kind of system for helping lawyers to write standardized legal documents, such as wills, licenses and simple contracts. Lawyers have traditionally used form books for this purpose. In recent years these form books have begun to be replaced by software, called document assembly systems. These systems are typically procedural programs which embed program logic in the text of the document, similar to the way many web sites are programmed today, by embedding PHP programs within HTML documents. While research has been conducted on using knowledge-based systems for this purpose [107], in which rules are applied to facts deductively to generate descriptions of documents, my work was the first to approach this task from a theory construction perspective. However I must admit that the focus of this research was more on constructing alternative legal theories of the case than exploring alternative theories of how best to formulate the document in a persuasive way, given some model of the intended audience. However, recently I've begun collaborating with Marc Lauritsen to work on precisely this problem [110].

Recently, I have been working on argument visualization, in the context of the Carneades project [75]. I have been experimenting with methods for visualizing Carneades argument graphs and designing graphical user interfaces for working with argument graphs. An important difference between my work and most prior work on argument visualization, with the exception of [167], is that my diagrams are views onto a mathematical model of argument graphs and the user interfaces provide ways to modify, control and view the underlying model. Argument diagramming software for Wigmore [174], Beardsley [12] or Toulmin [161] diagrams, such as Araucaria [142] or Rationale[5], lack such a mathematical foundation. Essentially, the diagrams are the models in these other systems, rather than views onto a model. My approach gives us freedom to experiment with different diagramming methods and user interfaces for manipulating Carneades argument graphs, without changing the underlying model of argument. In [75], I described a couple of different approaches, including one which is very close to Wigmore's style of argument diagramming. One diagramming issue Walton and I have been discussing again recently, is how best to represent undercutters, i.e. arguments which directly attack the inferential link between the premises of an argument and its conclusion.

As for future work, my research program remains focused on the argumentation tasks illustrated in Figure 1.2. The logical layer is, I claim, fairly complete and well understood, but it might be interesting to try to deepen and broaden this work, along the lines Prakken has been investigating [135], to find a way to use argumentation schemes to integrate reasoning with a variety of well-established formal methods, such as probability theory and fuzzy logic, where each method can be used, as appropriate, to address parts of a problem. At the dialectical level, I plan to return to

[5]http://rationale.austhink.com/

my work on dialogue protocols, begun with the Pleadings Game, but generalized to other kinds of dialogues. The goal would be to try to develop a formal language for expressing protocols for a wide-range of dialogue types which can be used as a basis for a new generation of mediation systems [80] for helping people to follow the procedural rules of these dialogues. The engineering challenge would be to find ways to design systems which make these protocols accessible in a user-friendly, unobtrusive way. I'd also like to the reconsider the problem of how to support deliberation dialogues, begun with my work on Zeno, but on the basis of work done in Carneades. Carneades was designed primarily to support persuasion dialogs, so it is not clear how suitable it might be for deliberation dialogues, or how to generalize the model if necessary to support such dialogues. In the Carneades project, I claim to have made good progress in modeling the use of proof burdens and standards for use in the aggregation and evaluation of arguments. However work on modeling specific legal proof standards, such as *preponderance of the evidence* and *beyond reasonable doubt*, has only begun recently [86] and is still quite preliminary. Finally, at the rhetorical level much work remains to be done. As mentioned previously, I've begun collaborating with Marc Lauritsen on a theory of document modeling and designing a new generation of document assembly tools, based on computational models of argument. And my colleagues and I are continuing to explore ways to visualize arguments and develop interactive, intuitive user-interfaces for browsing and understanding complex networks of arguments. However the focus of my research in the next few years will be on integrating prior results in the field of computational models of argument, not just my own, in software tools which make these results more accessible and easy-to-use for solving real practical problems and on validating these methods and tools in pilot applications.

References

[1] Pierson vs. Post. 3 Cai R 175 2 Am Dec 264 (Supreme Court of New York), 1805.

[2] Alchourron, C. E., and Bulygin, E. Limits of logic and legal reasoning. In *Expert Systems in Law*, A. A. Martino, Ed. Amsterdam, 1992, pp. 9–27.

[3] Aleven, V. *Teaching Case-Based Argumentation Through a Model and Examples*. Ph.d., University of Pittsburgh, 1997.

[4] Alexy, R. *A Theory of Legal Argumentation*. Oxford University Press, New York, N.Y., 1989.

[5] Allen, L. E., and Saxon, C. S. More ia needed in ai: Interpretation assistance for coping with the problem of multiple structural interpretations. In *Proceedings of the Third International Conference on Artificial Intelligence and Law*. 1991, pp. 53–61.

[6] Anderson, R., and Jr., N. D. B. Entailment: The logic of relevance and necessity.

[7] Ashley, K. D. *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. Artificial Intelligence and Legal Reasoning Series. MIT Press, Bradford Books, 1990.

[8] Ashley, K. D., Pinkwart, N., Lynch, C., and Aleven, V. Learning by diagramming supreme court oral arguments. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Law* (2007), ACM, pp. 271–275.

[9] Atkinson, K., Bench-Capon, T., and McBurney, P. Arguing about cases as practical reasoning. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law* (Bologna, Italy, 2005), pp. 35–44.

[10] Austhink. Rationale — argument mapping software. http://www.austhink.com, 2008.

[11] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., and Patel-Schneider, P., Eds. *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press, 2003.

[12] Beardsley, M. C. *Practical Logic*. Prentice Hall, New York, 1950.

[13] Bench-Capon, T. The missing link revisted: The role of teleology in representing legal argument. *Artificial Intelligence and Law 10*, 1-3 (September 2002), 79–94.

[14] Bench-Capon, T., and Coenen, F. Expoiting isomorphism: Development of a kbs to support british coal insurance claims. In *Proceedings of the Third International Conference on Artificial Intelligence and Law*. Oxford, 1991, pp. 62–68.

[15] Bench-Capon, T., and Coenen, F. P. Isomorphism and legal knowledge based systems. *Artificial Intelligence and Law 1*, 1 (1992), 65–86.

[16] Bench-Capon, T., Dunne, P., and Leng, P. Interacting with knowledge systems through dialogue games. In *Proceedings of the 11th Annual Conference on Expert Systems and their Applications (vol. 1)* (Avignon, 1991), pp. 123–130.

[17] Bench-Capon, T., Freeman, J., Hohmann, H., and Prakken, H. Computational models, argumentation theories and legal practice. In *Argumentation Machines; New Frontiers in Argument and Computation*, C. Reed and T. J. Norman, Eds. Kluwer Academic Publishers, 2003, pp. pp. 85–120.

[18] Bench-Capon, T., Geldard, T., and Leng, R. A method for the computational modelling of dialectical argument with dialog games. *Artificial Intelligence and Law 8*, 2-3 (2000), 233–254.

[19] Bench-Capon, T., and Gordon, T. F. Isomorphism and argumentation. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law (ICAIL 2009)* (New York, NY, USA, 2009), C. D. Hafner, Ed., ACM Press.

[20] Bench-Capon, T., and Sartor, G. A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence 150*, 1–2 (November 2003), 97–143.

[21] Berman, D. H., and Hafner, C. D. Representing teleological structure in case-based legal reasoning: The missing link. In *Proceedings of the 4th International Conference on Artificial Intelligence and Law* (New York, 1993), ACM Press, pp. 50–60.

[22] Berners-Lee, T., Hendler, J., and Lassila, O. The semantic web. *Scientific American 284*, 5 (May 2001), 34–43.

[23] Besnard, P., and Hunter, A. *Elements of Argumentation.* MIT Press, 2008.

[24] Betz, G. Argunet — argument mapping software. http://www.argunet.org, 2008.

[25] Bing, J. Uncertainty, decisions and information systems. In *Artificial Intelligence and Legal Information Systems*, C. Ciampi, Ed. North-Holland, 1982.

[26] Black, H. C. *Black's Law Dictionary.* West Publishing Co., 1979.

[27] Bondarenko, A., Dung, P. M., Kowalski, R. A., and Toni, F. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence 93*, 1-2 (1997), 63–101.

[28] Brachman, R. J., and Schmolze, J. G. An overview of the kl-one knowledge representation system. *Cognitive Science 9* (1985), 171–216.

[29] Branting, L. K. *Reasoning with Rules and Precedents: A Computational Model of Legal Analysis.* Kluwer Academic Publishers, Dordrecht, 2000. Book version of 1991 PhD Thesis.

[30] Branting, L. K., and Lester, J. C. Justification structures for document reuse. *Advances in Case-Based Reasoning 1168* (1996), 76–90.

[31] Breuker, J. Components of problem solving and types of problems. In *Proceedings of the 8th European Knowledge Acquisition Workshop* (London, UK, 1994), Springer-Verlag, pp. 118–136.

[32] Brewka, G. *Nonmonotonic Reasoning: From Theoretical Foundation Towards Efficient Computation.* PhD thesis, University of Hamburg, 1989.

[33] Brewka, G. *Nonmonotonic Reasoning: Logic Foundations of Commonsense.* Cambridge Tracts in Computer Science. Cambridge University Press, 1991.

[34] Brewka, G., and Gordon, T. F. How to buy a Porsche: An approach to defeasible decision making. In *Working Notes of the AAAI-94 Workshop on Computational Dialectics* (Seattle, Washington, 1994), pp. 28–38.

[35] Bryan, M. *SGML, An Author's Guide to the Standard Generalized Markup Language*. Addison-Wesley, 1988.

[36] Carr, C. S. Using computer supported argument visualization to teach legal argumentation. In *Visualizing Argumentation: Software Tools for Collaborative Education and Sense-Making*. Springer, 2003, ch. 4, pp. 75–96.

[37] Christaller, T., Primio, F. d., and Voss, A. *Die KI-Werkbank Babylon, eine offene und portable Entwicklungsumgebung für Expertensysteme*. Addison-Wesley, Reading, Mass., 1989.

[38] Clocksin, W. F., and Mellish, C. S. *Programming in Prolog*. Springer-Verlag, 1981.

[39] Conklin, J., and Begeman, M. gIBIS: a hypertext tool for exploratory policy discussion. *ACM Transactions on Office Information Systems 6*, 4 (1988), 303–331.

[40] Daskalopulu, A., and Sergot, M. The representation of legal contracts. *Artificial Intelligence and Society 11* (1997), 6–17.

[41] de Kleer, J. An assumption-based TMS. *Artificial Intelligence 28*, 2 (1986), 127–162.

[42] Delgrande, J. An approach to default reasoning based on a first-order conditional logic. In *Proceedings of the National Conference on Artificial Intelligence*. Seattle, 1987, pp. 340–345.

[43] Doyle, J. A truth maintenance system. *Artificial Intelligence 12* (1979), 231–272.

[44] Dung, P. M. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence 77*, 2 (1995), 321–357.

[45] Dworkin, R. M. Is law a system of rules? In *The Philosophy of Law*, R. M. Dworkin, Ed. Oxford University Press, 1977, pp. 38–65.

[46] Emerson, R. W. *Self-Reliance*. Philosopher Press, 1901.

[47] Engisch, K. *Logische Studien zur Gesetzesanwendung*. C. Winter, 1960.

[48] Erman, L. D., Hayes-Roth, F., Lesser, V. R., and Reddy, D. R. The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. *ACM Comput. Surv. 12*, 2 (1980), 213–253.

[49] Etherington, D. W., and Reiter, R. On inheritance hierarchies with exceptions. In *Proceedings of the AAAI*. 1983, pp. 104–108.

[50] Fiedler, H. Expert systems as a tool for drafting legal decisions. In *Logica, Informatica, Diritto*, A. A. Martino and F. S. Natali, Eds. Consiglio Nazionale delle Richere, Florence, 1985, pp. 265–274.

[51] Fiedler, H. Logical structure and information technology support of judicial decisions.

[52] Fowler, M., and Scott, K. *UML Distilled — A Brief Guide to the Standard Object Modeling Language*, 2nd ed. Addison Wesley Longman, Inc., 2000.

[53] Freeman, J. B. *Dialectics and the Macrostructure of Arguments: A Theory of Argument Structure*. Walter de Gruyter, Berlin / New York, 1991.

[54] Freeman, K., and Farley, A. M. A model of argumentation and its application to legal reasoning. *Artificial Intelligence and Law 4*, 3-4 (1996), 163–197.

[55] Gardner, A. *An Artificial Intelligence Approach to Legal Reasoning.* MIT Press, 1987.

[56] Geffner, H., and Pearl, J. Conditional entailment: Bridging two approaches to default reasoning. *Artificial Intelligence 53*, 2-3 (1992), 209–244.

[57] Gordon, T. F. On the suitability of modula-2 for artificial intelligence programming. *Angewandte Informatik*, July (1985), 296–303.

[58] Gordon, T. F. The role of exceptions in models of the law. In *Formalisierung im Recht und Ansätze juristischer Expertensysteme*, H. Fiedler and R. Traunmüller, Eds. J. Schweitzer Verlag, Munich, 1986, pp. 52–59.

[59] Gordon, T. F. Oblog-2: A hybrid knowledge representation system for defeasible reasoning. In *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, 1987), pp. 231–239.

[60] Gordon, T. F. Some problems with Prolog as a knowledge representation language for legal expert systems. In *Yearbook of Law, Computers and Technology*, C. Arnold, Ed. Leicester Polytechnic Press, Leicester, England, 1987, pp. 52–67.

[61] Gordon, T. F. The Argument Construction Set — a constructive approach to legal expert systems. Tech. rep., German Research Institute for Mathematics and Data Processing (GMD), 1988.

[62] Gordon, T. F. The importance of nonmonotonicity for legal reasoning. In *Expert Systems in Law; Impacts on Legal Theory and Computer Law*, H. Fiedler, F. Haft, and R. Traunmüller, Eds. Attempto Verlag, Tübingen, 1988, pp. 111–126.

[63] Gordon, T. F. Issue spotting in a system for searching interpretation spaces. In *Proceedings of the Second International Conference on Artificial Intelligence and Law.* Association for Computing Machinery (ACM), New York, 1989, pp. 157–164.

[64] Gordon, T. F. An abductive theory of legal issues. *International Journal of Man-Machine Studies 35* (1991), 95–118.

[65] Gordon, T. F. A theory construction approach to legal document assembly. In *Expert Systems in Law*, A. A. Martino, Ed. North-Holland, Amsterdam, 1992, pp. 211–225.

[66] Gordon, T. F. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice.* Ph.d., Technical University of Darmstadt, 1993.

[67] Gordon, T. F. The Pleadings Game — an exercise in computational dialectics. *Artificial Intelligence and Law 2*, 4 (1994), 239–292.

[68] Gordon, T. F. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice.* Springer, New York, 1995. Book version of 1993 Ph.D. Thesis; University of Darmstadt.

[69] Gordon, T. F. Computational dialectics. In *Computers as Assistants - A New Generation of Support Systems*, P. Hoschka, Ed. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1996, pp. 186–203.

[70] Gordon, T. F. Die Bedeutung von E-Governance für de öffentliche Verwaltung. *Verwaltung und Management 5* (2004), 258–263.

[71] Gordon, T. F. Information technology for good governance. In *French-German Symposium on Governance, Law and Technology* (September 2005), D. Bourcier, Ed., University of Paris, pp. 87–95.

[72] Gordon, T. F. *Demokratie Digital.* Wilhelm Fink Verlag, Paderborn, 2006, ch. Informationstechnologien für Good Governance.

[73] Gordon, T. F. Die Bedeutung von E-Governance für de öffentliche Verwaltung. In *Handbuch IT in der Verwaltung*, M. Wind and D. Kröger, Eds. Springer, Berlin Heidelberg, 2006, pp. 733–746.

[74] Gordon, T. F. Constructing arguments with a computational model of an argumentation scheme for legal rules. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Law* (2007), pp. 117–121.

[75] Gordon, T. F. Visualizing Carneades argument graphs. *Law, Probability and Risk 6*, 1-4 (2007), 109–117.

[76] Gordon, T. F. Hybrid reasoning with argumentation schemes. In *Proceedings of the 8th Workshop on Computational Models of Natural Argument (CMNA 08)* (Patras, Greece, July 2008), The 18th European Conference on Artificial Intelligence (ECAI 2008), pp. 16–25.

[77] Gordon, T. F., and Karacapilidis, N. The Zeno argumentation framework. In *Proceedings of the Sixth International Conference on Artificial Intelligence and Law* (Melbourne, Australia, 1997), ACM Press, pp. 10–18.

[78] Gordon, T. F., and Karacapilidis, N. The Zeno argumentation framework. *Künstliche Intelligenz 99*, 3 (1999), 20–29.

[79] Gordon, T. F., Karacapilidis, N., and Voss, H. Zeno — a mediation system for spatial planning. In *CSCW and the Web - Proceedings of the 5th ERCIM/W4G Workshop*, U. Busbach, D. Kerr, and K. Sikkel, Eds., no. 984 in Arbeitspapiere der GMD. GMD, Sankt Augustin, Germany, 1996, pp. 55–61.

[80] Gordon, T. F., and Märker, O. Mediation systems. In *Online-Mediation. Neue Medien in der Konfliktvermittlung - mit Beispielen aus Politik und Wirtschaft*, O. Märker and M. Trénel, Eds. Edition Sigma, Berlin, 2002, pp. 61–84.

[81] Gordon, T. F., Prakken, H., and Walton, D. The Carneades model of argument and burden of proof. *Artificial Intelligence 171*, 10-11 (2007), 875–896.

[82] Gordon, T. F., and Richter, G. Discourse support systems for deliberative democracy. In *eGovernment: State of the Art and Perspectives (EGOV02)* (Berlin / Heidelberg / New York, 2002), R. Traunmüller and K. Lenk, Eds., Springer Verlag, pp. 248–255.

[83] Gordon, T. F., Voss, A., Richter, G., and Märker, O. Zeno: Groupware for discourses on the internet. *Künstliche Intelligenz 2*, 1 (2001), 43–45.

[84] Gordon, T. F., and Walton, D. Pierson vs. Post revisted — a reconstruction using the Carneades Argumentation Framework. In *Proceedings of the First International Conference on Computational Models of Argument (COMMA 06)* (Liverpool, 2006), P. E. Dunne and T. Bench-Capon, Eds., IOS Press.

[85] Gordon, T. F., and Walton, D. Legal reasoning with argumentation schemes. In *12th International Conference on Artificial Intelligence and Law (ICAIL 2009)* (New York, NY, USA, 2009), C. D. Hafner, Ed., ACM Press.

[86] Gordon, T. F., and Walton, D. Proof burdens and standards. In *Argumentation in Artificial Intelligence*, I. Rahwan and G. Simari, Eds. Springer-Verlag, Berlin, Germany, 2009.

[87] Governatori, G., Maher, M., Antoniou, G., and Billington, D. Argumentation semantics for defeasible logic. *Journal of Logic and Computation 14* (2004), 675–702.

[88] Greenwood, K., Bench-Capon, T., and McBurney, P. Towards a computational account of persuasion in law. In *International Conference on Artificial Intelligence and Law* (Edinburgh, 2003), G. Sartor, Ed., ACM, pp. 22–31.

[89] Grennan, W. *Informal Logic.* McGill-Queen's University Press, Kingston, 1997.

[90] Grosof, B. N., Horrocks, I., Volz, R., and Decker, S. Description logic programs: Combining logic programs with description logics. In *Proceedings of the Twelth International World Wide Web Conference (WWW 2003)* (Budapest, Hungary, May 2003), ACM, pp. 48–57.

[91] Hafner, C., and Berman, D. H. The role of context in case-based legal reasoning: Teleological, temporal, and procedural. *Artificial Intelligence and Law 10*, 1-3 (2002), 19–64.

[92] Hage, J. Dialectical models in artificial intelligence and law. *Artificial Intelligence and Law 8*, 2-3 (2000), 137–172.

[93] Hage, J., Span, G., and Lodder, A. A dialogical model of legal reasoning. In *Legal Knowledge Based Systems: Information Technology and Law, JURIX '92*, e. a. C.A. Grütters, Ed. Lelystad, The Netherlands, 1992.

[94] Hage, J. C. *Reasoning with Rules – An Essay on Legal Reasoning and its Underlying Logic.* Kluwer Academic Publishers, Dordrecht, 1997.

[95] Hagedorn, H. Interaktive Bürgerbeteiligung Alexanderplatz Berlin. *Politik Digital* (May 2003).

[96] Hamblin, C. L. Mathematical models of dialogue. *Theoria 37* (1971), 130–155.

[97] Harrell, M. Using argument diagrams to improve critical thinking skills in introductory philosophy. Tech. rep., Department of Philosophy, Carnegie Mellon University, 2005.

[98] Hart, H. L. A. *The Concept of Law.* Clarendon Press, Oxford, 1961.

[99] Hart, H. L. A. *Essays in Jurisprudence and Philosophy.* Oxford University Press, 1983.

[100] Hewitt, C. Offices are open systems. *ACM Transactions on Office Information Systems July* (1986), 271–287.

[101] Hintikka, J. Information-seeking dialogues: A model. *Erkenntnis 14* (1979).

[102] Holsapple, C. W., and Whinston, A. B. *Decision Support Systems: A Knowledge-Based Approach.* West Publishing Co., 1996.

[103] Horn, R. E. Infrastructure for navigating interdisciplinary debates: Critical decisions for representing argumentation. In *Visualizing Argumentation: Software Tools for Collaborative Education and Sense-Making.* Springer, 2003, ch. 8, pp. 165–184.

[104] Jarczyk, A. P. J., Löffler, P., and III, F. M. S. Design rational for software engineering: A survey. In *Proceedings of the 25th International Conference on System Sciences* (1992), vol. 2, pp. 557–586.

[105] Karpf, J. Isomorphic and top-down modelling of statutory law.

[106] Kirschner, P. A., Shum, S. J. B., and Carr, C., Eds. *Visualizing Argumentation: Software Tools for Collaborative Education and Sense-Making.* Springer-Verlag, London, 2002.

[107] Kowalewski, D. L., Scheeberger, J., and Wiefel, S. KOKON-3: ein prototypisches System zur wissensbasierten vertragskonfigurierung. In *GI - 19. Jahrestagung, II Computergestützter Arbeitsplatz* (Berlin, 1989), Springer-Verlag, pp. 79–92.

[108] Kraus, S., Lehmann, D., and Magidor, M. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence 44* (1990), 167–207.

[109] Lauritsen, M. Building legal practice systems with today's commercial authoring tools. *Artificial Intelligence and Law 1*, 1 (1992), 87–102.

[110] Lauritsen, M., and Gordon, T. F. Toward a general theory of document modeling. In *12th International Conference on Artificial Intelligence and Law (ICAIL 2009)* (New York, NY, USA, 2009), C. D. Hafner, Ed., ACM Press.

[111] Leith, P. Clear rules and legal expert systems. In *Logica, Informatica, Diritto*, A. A. M. Natali and F. Socci, Eds. Florence, 1985.

[112] Leith, P. Fundamental errors in legal logic programming. *Computer Journal 29*, 6 (1986), 545–552.

[113] Lessig, L. *Code: Version 2.0.* Basic Books, 2006.

[114] Li, G., Uren, V., Motta, E., Shum, S. B., and Domingue, J. Claimaker: Weaving a semantic web of research papers. In *Proceedings of the First International Semantic Web Conference (ISWC 2002)* (Sardinia, Italy, June 9-12 2002).

[115] Lodder, A. R. *DiaLaw — On Legal Justification and Dialogical Model of Argumentation.* Law and Philosophy Library. Springer, Berlin, Germany, 2002.

[116] Lorenzen, P., and Lorenz, K. *Dialogische Logik.* Wissenschaftliche Buchgesellschaft,, Darmstadt, 1978.

[117] Loui, R. P. Process and policy: resource-bounded non-demonstrative reasoning. *Computational Intelligence 14* (1998), 1–38.

[118] Lührs, R., Albrecht, S., Lübcke, M., and Hohberg, B. How to grow? online consultation about growth in the city of hamburg: Methods, techniques, success factors. In *EGOV 2003* (Prague, 2003), R. Traunmüller, Ed., Springer Verlag, pp. 79–84.

[119] Macintosh, A. Using information and communication technologies to enhance citizen engagement in the policy process. In *Promises and Problems of E-Democracy: Challenges of Online Citizen Engagement.* OECD, Paris, 2004, pp. 19–142.

[120] Mackenzie, J. D. Question-begging in non-cumulative systems. *Journal of Philosophical Logic 8* (1979), 117–133.

[121] Märker, O., Hagedorn, H., Trénel, M., and Gordon, T. F. Internet-based citizen participation in the City of Esslingen. relevance — moderation — software. In *CORP 2002 - "Who plans Europe's future?"* (2002), M. Schrenk, Ed., Technical University of Vienna, pp. 39–45.

[122] McCarty, L. T. Some arguments about legal arguments. In *International Conference on Artificial Intelligence and Law* (Melbourne, 1997), pp. 215–224.

[123] Melle, W. v., Scott, A., Bennett, J., and Peairs, M. The emycin manual.

[124] Nute, D. Defeasible logic. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, D. Gabbay, C. Hogger, and J. Robinson, Eds. Clarendon Press, Oxford, 1994, pp. 253–395.

[125] Perelman, C. *Juristische Logik als Argumentationslehre.* Alber, München, 1979.

[126] Perelman, C. *Justice, Law and Argument: Essays on Moral and Legal Reasoning.* D. Reidel, 1980.

[127] Perelman, C., and Olbrechts-Tyteca, L. *The New Rhetoric.* University of Notre Dame Press, Notre Dame, 1969.

[128] Petri, C. A. *Kommunikation mit Automaten.* Schriften des iim nr. 2, Institut für Instrumentelle Mathematik, Universität Bonn, Bonn, 1962.

[129] Pollock, J. Defeasible reasoning. *Cognitive Science 11*, 4 (1987), 481–518.

[130] Poole, D. A logical framework for default reasoning. *Artificial Intelligence 36*, 1 (1988).

[131] Posner, R. A. Economic analysis of law.

[132] Prakken, H. A tool in modelling disagreement in law: Preferring the most specific argument. In *Proceedings of the Third International Conference on Artificial Intelligence and Law* (Oxford, 1991), pp. 165–174.

[133] Prakken, H. From logic to dialectic in legal argument. In *Proceedings of the Fifth International Conference on Artificial Intelligence and Law* (Maryland, 1995), pp. 165–174.

[134] Prakken, H. *Logical Tools for Modelling Legal Argument : a Study of Defeasible Reasoning in Law.* Law and philosophy library; v. 32. Kluwer Academic Publishers, Dordrecht ; Boston, 1997. Henry Prakken. 25 cm. Rev. and extended version of the author's thesis (Ph. D.–Free University, Amsterdam, 1993).

[135] Prakken, H. Combining modes of reasoning: An application of abstract argumentation. In *Proceedings of The 11th European Conference on Logics in Artificial Intelligence (JELIA 2008)* (Berlin, April 2008), vol. 5293 of *Springer Lecture Notes in AI*, Springer Verlag, pp. 349–361.

[136] Prakken, H., and Sartor, G. A dialectical model of assessing conflicting argument in legal reasoning. *Artificial Intelligence and Law 4*, 3-4 (1996), 331–368.

[137] Prakken, H., and Sartor, G. Modelling reasoning with precedents in a formal dialogue game. *Artificial Intelligence and Law 6*, 2-4 (1998), 231–287.

[138] Rahwan, I., and Banihashemi, B. Arguments in OWL: A progress report. In *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA)* (Amsterdam, The Netherlands, 2008), A. Hunter, Ed., IOS Press.

[139] Rawls, J. Outline of a decision procedure for ethics. *Philosophical Review* (1951), 177–197.

[140] Rawls, J. *A Theory of Justice.* Belknap Press of Harvard University Press, 1971.

[141] Reed, C., and Rowe, G. Araucaria: Software for puzzles in argument diagramming and XML. Technical report, Department of Applied Computing, University of Dundee, 2001.

[142] Reed, C. A., and Rowe, G. W. Araucaria: Software for argument analysis, diagramming and representation. *International Journal of AI Tools 13*, 4 (2004), 961–980.

[143] Rescher, N. *Dialectics: A Controversy-Oriented Approach to the Theory of Knowledge*. State University of New York Press, Albany, New York, USA, 1977.

[144] Rittel, H. W., and Webber, M. M. Dilemmas in a general theory of planning. *Policy Science 4* (1973), 155–169.

[145] Rosenschein, J. S., and Zlotkin, G. *Rules of Encounter: Designing Conventions for Automated Negotation Amoung Computers*. MIT Press, 1994.

[146] Routen, T. Hierarchically organized formulations. In *Proceedings of the 2nd International Conference on Artificial Intelligence and Law* (New York, 1989), ACM Press, pp. 242–250.

[147] Routen, T., and Bench-Capon, T. Hierarchical formalizations. *International Journal of Man-Machine Studies 35* (1991), 69–93.

[148] Rowe, G. W. A., Reed, C. A., and Katzav, J. Araucaria: Marking up argument. In *Working Notes of the European Conference on Computing and Philosophy* (Glasgow, Scotland, 2003).

[149] Russell, S., and Norvig, P. *Artificial Intelligence: A Modern Approach*, second ed. Prentice Hall Series in Artificial Intelligence. Prentice Hall, 2003.

[150] Schweichhart, K. Das argument construction set. Master's thesis, Universität Bonn, 1988.

[151] Selvin, A., Shum, S. B., and Maarten Sierhuis, e. a. Compendium: Making meetings into knowledge events. In *Knowledge Technologies 2001* (Austin, Texas, March 2001).

[152] Sergot, M., Sadri, F., Kowalski, R., Kriwaczek, F., Hammond, P., and Cory, H. The British Nationality Act as a logic program. *Communications of the ACM 29*, 5 (1986), 370–386.

[153] Shoham, Y. A semantical approach to nonmonotonic logics. In *Readings in Nonmonotonic Reasoning*, M. L. Ginsberg, Ed. 1987, pp. 227–250.

[154] Shum, S. B., and Hammond, N. Argumentation-based design rationale: What use and what cost? *International Journal of Human-Computer Studies 40*, 4 (1994), 603–652.

[155] Shum, S. B., Uren, V., Li, G., Domingue, J., and Motta, E. Visualizing internetworked argumentation. In *Visualizing Argumentation: Software Tools for Collaborative Education and Sense-Making*. Springer, 2003, ch. 9, pp. 185–204.

[156] Simari, G. R., and Loui, R. P. A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence 53*, 2-3 (1992), 125–157.

[157] Skalak, D. B., and Rissland, E. L. Arguments and cases: An inevitable intertwining. *Aritificial Intelligence and Law 1*, 1 (1992), 3–45.

[158] Slomkowski, P. *Aristotle's Topics*, vol. 74. Brill Academic Publishers, Leiden, The Netherlands, 1997.

[159] Sprowl, J. Automating the legal reasoning process: A computer that uses regulations and statutes to draft legal documents. *American Bar Foundation Research Journal 4*, 1 (1979), 1–81.

[160] Suchman, L. Do categories have politics? the language/action perspective reconsidered. *Computer-Supported Cooperative Work 2*, 177-190 (1994).

[161] Toulmin, S. E. *The Uses of Argument.* Cambridge University Press, Cambridge, UK, 1958.

[162] Touretzky, D. S. The mathematics of inheritance systems.

[163] Turban, E., and Aronson, J. *Decision Support Systems and Intelligent Systems*, 5th ed. Prentice Hall, 1997.

[164] van Eemeren, F. H., and Grootendorst, R. *Argumentation, Communication and Fallacies.* Lawrence Erlbaum Associates, 1992.

[165] van Eemeren, F. H., Grootendorst, R., Henkemans, F. S., Blair, J. A., Johnson, R. H., Krabbe, E. C. W., Walton, D. N., Willard, C., Woods, J., and Zarefsky, D. *Fundamentals of Argumentation Theory: A Handbook of Historical Backgrounds and Contemporary Developments.* Lawrence Erlbaum Associates, 1996.

[166] Verheij, B. *Rules, Reasons, Arguments. Formal Studies of Argumentation and Defeat.* Ph.d., Universiteit Maastricht, 1996.

[167] Verheij, B. *Virtual Arguments.* TMC Asser Press, The Hague, 2005.

[168] Walker, R. *An Expert System Architecture for Hetergeneous Domains – A Case Study in the Legal Field.* PhD thesis, Vrije Universiteit Amsterdam, 1992.

[169] Walton, D. *The New Dialectic: Conversational Contexts of Argument.* University of Toronto Press, Toronto; Buffalo, 1998. 24 cm.

[170] Walton, D. The new dialectic: A method of evaluating an argument used for some purpose in a given case. *ProtoSociology 13* (1999), 70–91.

[171] Walton, D. Abductive, presumptive and plausible arguments. *Informal Logic 21* (2001), 141–169.

[172] Walton, D. *Argumentation Methods for Artificial Intelligence in Law.* Springer, 2005.

[173] Walton, D. *Fundamentals of Critical Argumentation.* Cambridge University Press, Cambridge, UK, 2006.

[174] Wigmore, J. H. *A Treatise on the System of Evidence in Trials at Common Law: Including the Statutes and Judicial Decisions of all Jurisdictions of the United States.* Little, Brown and Company, Boston, Massachusetts, USA, 1908.

[175] Winograd, T., and Flores, F. *Understanding Computers and Cognition, A New Foundation for Design.* Ablex, Norwood, New Jersey, 1986.

[176] Wyner, A., and Bench-Capon, T. Argument schemes for legal case-based reasoning. In *JURIX 2007: The Twentieth Annual Conference on Legal Knowledge and Information Systems* (2007).

Appendix A

Submitted Publications

1. Thomas F. Gordon. The role of exceptions in models of the law. In H. Fiedler and R. Traunmüller, editors, *Formalisierung im Recht und Ansätze juristischer Expertensysteme*, pages 52–59. J. Schweitzer Verlag, Munich, 1986.

   Existing knowledge representation languages designed for building models of legal domains have required that exceptions to a general rule be collapsed into a single, detailed rule. Although the logic of such rules is clear, they are useless when the complexity of a model grows to realistic proportions. To support this claim, the nature and purpose of models of the law is discussed, and the role exceptions play in furthering this purpose is outlined.

2. Thomas F. Gordon. Oblog-2: A hybrid knowledge representation system for defeasible reasoning. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, pages 231–239, Boston, 1987.

   Oblog-2 is a hybrid knowledge representation system comparable to Krypton and KL-TWO. It combines a terminological reasoner with a Prolog-like inference mechanism. The terminological component supports the description of type and attribute taxonomies. Entities are instances of a set of types. Procedures for determining the values of attributes are Horn clause rules indexed by type. The known types of an entity determine its set of applicable rules, which changes as our knowledge about the types of the entity is refined, supporting a form of defeasible reasoning. Oblog-2 has been designed for modeling legal domains. Laws can be represented as general rules with exceptions, a technique traditionally used in the law, together with burden of proof rules, for reaching decisions when less than perfect information is available.

3. Thomas F. Gordon. Some problems with Prolog as a knowledge representation language for legal expert systems. In C. Arnold, editor, *Yearbook of Law, Computers and Technology*, pages 52–67. Leicester Polytechnic Press, Leicester, England, 1987.

   In vol 2 of this Yearbook, Marek Sergot and his collegues at Imperial College discussed their formalisation in Prolog of the British Nationality Act. The idea of using Prolog as a "knowledge representation language" for legal expert systems has been a popular one in the last few years, but Prolog suffers from a variety of problems which make it unsuitable for this purpose. Some of these problems were mentioned in the British Nationality Act article. I would like to expand upon these difficulties here. I begin by outlining a view of the process of deciding legal cases.

Next, the architecture of knowledge-based systems is described. I then describe some properties these systems must possess to be useful tools for supporting the legal decision-making process, and I measure Prolog against this list of properties. Finally, I briefly discuss Oblog, the language we have been developing to overcome the shortcomings of Prolog for legal expert systems.

4. Thomas F. Gordon. The Argument Construction Set — a constructive approach to legal expert systems. Technical report, German Research Institute for Mathematics and Data Processing (GMD), 1988.

    Usually, legal expert systems are conceived as rule-based systems for subsuming the facts of a case under the general rules of some substantive area of law. In such systems, the role of the lawyer-user is reduced to that of answering questions about the facts of the case. The Imperial College expert system for the British Nationality Act is perhaps the most familiar system of this kind. The guiding view of jurisprudence in such systems is that the substantive law of some field can be adequately represented as a set of rules and that deduction is the central task of legal decision making. There is an opposing view which asserts that deduction, although important, plays only a secondary role and that the principle task of a lawyer when analyzing a case is the construction of the theory of the law and facts from which the legal conclusions deductively follow. In this paper, we describe the architecture of a software system, called the Argument Construction Set, which is a first attempt to support this second, theory construction view of legal decision making. The system does not depend on the existence of clear legal rules and applies Artificial Intelligence ideas concerning reason maintenance and nonmonontonic reasoning.

5. Thomas F. Gordon. The importance of nonmonotonicity for legal reasoning. In H. Fiedler, F. Haft, and R. Traunmüller, editors, *Expert Systems in Law; Impacts on Legal Theory and Computer Law*, pages 111–126. Attempto Verlag, Tübingen, 1988.

    Laws are usually structured as general rules with exceptions. This papers presents arguments showing how this traditional structure furthers the normative and dispute resolution goals of legal systems. It makes the law easier for lay persons to learn and to apply. It minimizes the effort required to update ones knowledge of the law after revisions and modifications. Finally, rules with exceptions lower the cost of deciding cases.

6. Thomas F. Gordon. Issue spotting in a system for searching interpretation spaces. In *Proceedings of the Second International Conference on Artificial Intelligence and Law*, pages 157–164. Association for Computing Machinery (ACM), New York, 1989.

    A method for spotting issues is described which uses a system we are developing for searching interpretations spaces and constructing legal arguments. The system is compatible with the legal philosophy known as legal positivism, but does not depend on its notion of clear cases. AI methods applied in the system include an ATMS reason maintenance system, Poole's framework for default reasoning, and an interactive natural deduction theorem prover with a programmable control component for including domain-dependent heuristic knowledge. Our issue spotting method is compared with Gardner's program for identifying the hard and easy issues raised by offer and acceptance law school examination questions.

7. Thomas F. Gordon. An abductive theory of legal issues. *International Journal of Man-Machine Studies*, 35:95–118, 1991.

A normative theory of legal issues, argument moves and clear cases is presented in which abduction, rather than deduction, is of central importance. The theory is a refinement of Fiedler's constructive view of legal reasoning. Like legal positivism, the theory elaborates the concept of a clear case, but here clearness is defined relative to a set of competing interpretations of the law, rather than a single consistent set of 'valid' rules. A computational model for the theory is described, which uses an ATMS reason maintenance system to implement abduction. Finally, the theory is compared with Anne Gardner's program for spotting issues in offer and acceptance law school examination questions.

8. Thomas F. Gordon. A theory construction approach to legal document assembly. In Antonio A. Martino, editor, *Expert Systems in Law*, pages 211–225. North-Holland, Amsterdam, 1992.

   An overview of a software system under development for "assembling" legal documents is presented. The system applies Artificial Intelligence (AI) methods and is founded on a theory construction or abduction view of legal reasoning. The AI methods employed include an Assumption-based Truth Maintenance System (ATMS), a Natural Deduction theorem prover, and an implementation of Poole's approach to default reasoning.

9. Thomas F. Gordon. The Pleadings Game — an exercise in computational dialectics. *Artificial Intelligence and Law*, 2(4):239–292, 1994.

   The Pleadings Game is a normative formalization and computational model of civil pleading, founded in Robert Alexy's discourse theory of legal argumentation. The consequences of arguments and counterarguments are modeled using Geffner and Pearl's nonmonotonic logic, conditional entailment. Discourse is focussed using the concepts of issue and relevance. Conflicts between argument can be resolved by arguing about the validity and priority of rules, at any level. The computational model is fully implemented and has been tested using examples from Article Nine of the Uniform Commercial Code.

10. Thomas F. Gordon. Computational dialectics. In Peter Hoschka, editor, *Computers as Assistants - A New Generation of Support Systems*, pages 186–203. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1996.

   The central task in practical problem solving is to identify and choose among alternative courses of action. Computer science has failed to provide adequate tools for supporting rational, effective and fair decision-making under the conditions which usually prevail. Especially, computer science has yet to develop models of rational decision-making in groups which adequately take into consideration resource limitations or conflicts of interest and opinion. This paper provides an informal overview of Zeno, a mediating system for supporting discussion, argumentation and decision-making in groups, which explicitly takes these considerations into account. Also, a new subfield of computer science is proposed, computational dialectics, whose subject matter is computational models of norms of rational discourse. Zeno is a contribution to this field, based on the thesis that rationality can best be understood as theory construction regulated by discourse norms.

11. Thomas F. Gordon and Nikos Karacapilidis. The Zeno argumentation framework. In *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*, pages 10–18, Melbourne, Australia, 1997. ACM Press.

The Zeno Argumentation Framework is a formal model of argumentation based on the informal models of Toulmin and Rittel. Its main feature is a labelling function using arguments to compute heuristic information about the relative quality of the alternative positions proposed as solutions for some practical issue. The Zeno Argumentation Framework was designed to be used in mediation systems, an advanced kind of electronic discussion forum with special support for argumentation, negotiation and other structured forms of group decision-making.

12. Thomas F. Gordon and Douglas Walton. Pierson vs. Post revisted — a reconstruction using the Carneades Argumentation Framework. In Paul E. Dunne and Trevor Bench-Capon, editors, *Proceedings of the First International Conference on Computational Models of Argument (COMMA 06)*, Liverpool, 2006. IOS Press.

    Pierson vs. Post has become an important benchmark in the field of AI and Law for computational models of argumentation. Bench-Capon used Pierson vs. Post to motivate the use of values and value preferences in his theory-construction account of legal argument. And in a more a recent paper by Atkinson, Bench-Capon and McBurney it was used to illustrate a formalization of an argumentation scheme for practical reasoning. Here we offer yet another reconstruction of Pierson vs. Post, using our Carneades Argumentation Framework, a formal mathematical model of argument structure and evaluation based on Walton's theory of argumentation, and compare it to this prior work. Carneades, named in honor of the Greek skeptic philosopher who emphasized the importance of plausible reasoning, applies proof standards to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis.

13. Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10-11):875–896, 2007.

    We present a formal, mathematical model of argument structure and evaluation, taking seriously the procedural and dialogical aspects of argumentation. The model applies proof standards to determine the acceptability of statements on an issue-by-issue basis. The model uses different types of premises (ordinary premises, assumptions and exceptions) and information about the dialectical status of statements (stated, questioned, accepted or rejected) to allow the burden of proof to be allocated to the proponent or the respondent, as appropriate, for each premise separately. Our approach allows the burden of proof for a premise to be assigned to a different party than the one who has the burden of proving the conclusion of the argument, and also to change the burden of proof or applicable proof standard as the dialogue progresses from stage to stage. Useful for modeling legal dialogues, the burden of production and burden of persuasion can be handled separately, with a different responsible party and applicable proof standard for each. Carneades enables critical questions of argumentation schemes to be modeled as additional premises, using premise types to capture the varying effect on the burden of proof of different kinds of questions.

14. Thomas F. Gordon. Constructing arguments with a computational model of an argumentation scheme for legal rules. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Law*, pages 117–121, 2007.

    A knowledge representation language for defeasible legal rules is defined, whose semantics is purely procedural, based on Walton's theory of argumentation and Loui's break with the relational tradition in 'Process and Policy'. Legal rules are

interpreted as reasoning policies, by mapping them in the semantics to argumentation schemes. The reasoning process is regulated by argumentation protocols. Reasoning with legal rules is viewed as applying schemes for arguments from rules to construct arguments to be put forward in dialogues.

15. Thomas F. Gordon. Visualizing Carneades argument graphs. *Law, Probability and Risk*, 6(1-4):109–117, 2007.

    Carneades is a computational model of argument, based on the state-of-the-art of argumentation theory in philosophy. This article presents a diagramming method for Carneades, similar to Wigmore charts, and illustrates how to map legal evidence using this method. With suitable computer support, in the form of a special purpose argument diagram editor, users need not understand the mathematics of the computational model to make use of its features. Compared to a generic diagram editor, or even special purpose argument diagramming tools based only on informal models of argument, an argument diagramming tool based on the Carneades computational model of argument has the advantage of being able to inform users, in an intuitively comprehensible way, whether or not a claim satisfies a proof standard, given the evidence and other arguments which have been put forward by the parties. The presentation is entirely informal. No prior expertise in argumentation theory, mathematics or computer science is presumed.

16. Thomas F. Gordon. Hybrid reasoning with argumentation schemes. In *Proceedings of the 8th Workshop on Computational Models of Natural Argument (CMNA 08)*, pages 16–25, 2008.

    Practical reasoning typically requires a variety of argumentation schemes to be used together to solve problems and make decisions. For example, a legal case may raise issues requiring argument from precedent cases, rules, policy goals, moral principles, jurisprudential doctrine, social values and evidence. We present an extensible software architecture which allows diverse computational models of argumentation schemes to be used together in an integrated way to construct and search for arguments. The architecture has been implemented in Carneades, a software library for building argumentation tools. The architecture is illustrated with models of schemes for argument from ontologies, rules, cases and testimonial evidence and compared to blackboard systems for hybrid reasoning.

Appendix B

Related Publications

1. Thomas F. Gordon. Object-oriented predicate logic and its role in representing legal knowledge. In Charles Walter, editor, *Computing Power and Legal Reasoning*, pages 163–203. West Publishing Co., St. Paul, 1985.

2. Thomas F. Gordon and Gerald Quirchmayr. Oblog-2: Ein hybrides wissensrepräsentationssystem zur modellierung rechtswissenschaftlicher probleme. In G. Hommel and S. Schindler, editors, *GI-16. Jahrestagung, Proceedings*, volume 2, pages 406–420, 1986.

3. Thomas F. Gordon and Gerald Quirchmayr. Oblog: eine programmiersprache für juristische expertensysteme. In Ulrich Erdmann, Herbert Fiedler, Fritjof Haft, and Roland Traunmüller, editors, *Computergestützte Juristische Expertensysteme*, pages 123–134. Attempto Verlag, Tübingen, 1986.

4. Thomas F. Gordon and Gerald Quirchmayr. Der einsatz der modellierungssprache oblog zum entwurf von juristischen expertensysteme. In R.R. Wagner, R. Traunmüller, and H.C. Mayr, editors, *Informationsbedarfsermittlung und -analyse für den Entwurk von Informationssystemenanalyse*, pages 137–154. Springer-Verlag, London, 1987.

5. Herbert Fiedler and Thomas F. Gordon. Recht und rechtsanwendung als paradigma wissensbasierter systeme. In W. Brauer Wahlster and W., editors, *Tagungsband Wissensbasierte Systeme*, pages 63–77. Springer, 1987.

6. Thomas F. Gordon. Issue spotting in a system for searching interpretation spaces. In *Proceedings of the Second International Conference on Artificial Intelligence and Law*, pages 157–164. Association for Computing Machinery (ACM), New York, 1989.

7. Thomas F. Gordon. Künstliche Intelligenz und Recht. *Jur PC*, 5:605–608, 1990.

8. Thomas F. Gordon. Künstliche Intelligenz und Recht (Teil 2). *Jur PC*, 6:638–639, 1990.

9. Thomas F. Gordon. Artificial intelligence: A hermeneutic defense. In Christiane Floyd Keil-Slawik, Heinz Züllighoven, Reinhard Budde, and Reinhard, editors, *Software Development and Reality Construction*, pages 280–290. Springer-Verlag, Berlin, 1992.

10. Thomas F. Gordon. The Pleadings Game; formalizing procedural justice. In *Proceedings of the Fourth International Conference on Artificial Intelligence and Law*, pages 10–19. ACM Press, New York, 1993.

11. Thomas F. Gordon. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice*. Ph.d., Darmstadt University of Technology, 1993.

12. Thomas F. Gordon. From Jhering to Alexy — using Artificial Intelligence models in jurisprudence. In H. Prakken Soeteman, A. J. Muntjewerff, and A., editors, *Legal Knowledge Based Systems, The Relation with Legal Theory*, pages 19–31. Koninklijke Vermande BV, Lelystad, 1994.

13. Gerhard Brewka and Thomas F. Gordon. How to buy a Porsche: An approach to defeasible decision making. In *Working Notes of the AAAI-94 Workshop on Computational Dialectics*, pages 28–38, Seattle, Washington, 1994.

14. Thomas F. Gordon. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice*. Springer, New York, 1995. Book version of 1993 Ph.D. Thesis; University of Darmstadt.

15. Gerhard Brewka, Thomas F. Gordon, and Nikos Karacapilidis. Mediating systems for group decision making: the Zeno system. In *KI-95 Workshop on Computational Dialectics: Models of Argumentation, Negotiation and Decision Making*, 1995.

16. Thomas F. Gordon. Zeno: A WWW system for geographical mediation. In P. J. Densham, Marc P. Armstrong, and Karen K. Kemp, editors, *Collaborative Spatial Decision-Making, Scientific Report of the Initiative 17 Specialist Meeting*, volume 95-14, pages 77–89. National Center for Geographic Information Systems, Santa Barbara, 1995.

17. Thomas F. Gordon, Nikos Karacapilidis, and Hans Voss. Zeno — a mediation system for spatial planning. In U. Busbach, D. Kerr, and K. Sikkel, editors, *CSCW and the Web - Proceedings of the 5th ERCIM/W4G Workshop*, number 984 in Arbeitspapiere der GMD, pages 55–61. GMD, Sankt Augustin, Germany, 1996.

18. Thomas F. Gordon, Nikos Karacapilidis, Hans Voss, and Andreas Zauke. Computer-mediated cooperative spatial planning. In H. Timmermans, editor, *Decision Support Systems in Urban Planning*, pages 299–309. E. & F.N. Spon, London, 1997.

19. Thomas F. Gordon and Nikos Karacapilidis. The Zeno argumentation framework. *Künstliche Intelligenz*, 99(3):20–29, 1999.

20. Thomas F. Gordon and Oliver Märker. Mediation systems. In Lorenz M. Hilty and Paul W. Gilgen, editors, *Sustainability in the Information Society; 15th International Symposium Informatics for Environment Protection*, pages 737–742, Zürich, 2001. Metropolis Verlag.

21. Thomas F. Gordon and Gernot Richter. Discourse support systems for deliberative democracy. In Roland Traunmüller and Klaus Lenk, editors, *eGovernment: State of the Art and Perspectives (EGOV02)*, pages 248–255, Aix-en-Provence, 2002. Springer Verlag.

22. Thomas F. Gordon and Oliver Märker. Mediation systems. In Oliver Märker and Matthias Trénel, editors, *Online-Mediation. Neue Medien in der Konfliktvermittlung - mit Beispielen aus Politik und Wirtschaft*, pages 61–84. Edition Sigma, Berlin, 2002.

23. Thomas F. Gordon. A use case analysis of legal knowledge-based systems. In Danièle Bourcier, editor, *Legal Knowledge and Information Systems (JURIX 2003)*, Frontiers in Artificial Intelligence and Applications, pages 81–90, Utrecht, 2003. IOS Press, Amsterdam.

24. Thomas F. Gordon. An open, scalable and distributed platform for public discourse. In Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, and Wolfgang Wahlster, editors, *Informatik 2003*, volume 2, pages 232–234, Frankfurt am Main, 2003. Springer Verlag.

25. Thomas F. Gordon. Basis wissen. *Move Moderne Verwaltung*, Mai, 2004.

26. Thomas F. Gordon. Die Bedeutung von E-Governance für de öffentliche Verwaltung. *Verwaltung und Management*, 5:258–263, 2004.

27. Thomas F. Gordon. A computational model of argument for legal reasoning support systems. In Paul E. Dunne and Trevor Bench-Capon, editors, *Argumentation in Artificial Intelligence and Law*, IAAIL Workshop Series, pages 53–64, Nijmegen, The Netherlands, 2005. Wolf Legal Publishers.

28. Thomas F. Gordon. Information technology for good governance. In Daniele Bourcier, editor, *French-German Symposium on Governance, Law and Technology*, pages 87–95. University of Paris, September 2005.

29. Olivier Glassey, Thomas Gordon, and Jonas Pattberg. Machbarkeitstudie eines wissenbasierten Rechtsberatungssystems im Kreis Herford. In Erich Schweighofer, Doris Liebwald, Silvia Augeneder, and Thomas Menzel, editors, *Effizienz von e-Lösungen in Staat und Gesellschaft; Aktuelle Fragen der Rechtsinformatik*, pages 379–386. Boorberg-Verlag, 2005.

30. Olivier Glassey and Thomas F. Gordon. Feasibility study for a legal knowledge system in the County of Herford. In Maria Wimmer, Roland Traunmüller, Åke Grönlund, and Kim V. Andersen, editors, *Electronic Government: 4th International Conference (EGOV 2005)*, volume 3591 of *Lecture Notes in Computer Science*, pages 186 – 197. Springer-Verlag, August 2005.

31. Thomas F. Gordon. *Demokratie Digital*, chapter Informationstechnologien für Good Governance. Wilhelm Fink Verlag, Paderborn, 2006.

32. Thomas F. Gordon. Legal knowledge systems. In Ari-Veikko Anttiroiko and Matti Mälkiä, editors, *Encyclopedia of Digital Government*, pages 1161–1166. Idea Group, Hershey, Pennsylvania, 2006.

33. Thomas F. Gordon and Douglas Walton. The Carneades argumentation framework — using presumptions and exceptions to model critical questions. In *Proceedings of the 6th ECAI Workshop on Computational Models of Natural Argument (CMNA 6)*, Riva del Garde, Italy, September 2006.

34. Thomas F. Gordon and Douglas Walton. The Carneades argumentation framework — using presumptions and exceptions to model critical questions. In Paul E. Dunne and Trevor J.M. Bench-Capon, editors, *Computational Models of Argument. Proceedings of COMMA 2006*, pages 195–207, Amsterdam, September 2006. IOS Press.

35. Thomas F. Gordon. Die Bedeutung von E-Governance für de öffentliche Verwaltung. In Martin Wind and Detlef Kröger, editors, *Handbuch IT in der Verwaltung*, pages 733–746. Springer, Berlin Heidelberg, 2006.

36. Thomas F. Gordon, Ann Macintosh, and Alastair Renton. Argumentation support systems. Project Deliverable D5.2, DEMO-Net — The Democracy Network (FP6-2004-IST-4-027219), 2006.

37. Thomas F. Gordon and Doug Walton. Pierson vs. Post: A computational analysis of the majority and minority opinions. In Katie Atkinson, editor, *Jurix 2007 Workshop on Modelling Legal Cases*, 2007.

38. Thomas F. Gordon. Constructing legal arguments with rules in the legal knowledge interchange format (LKIF). In Pompeu Casanovas, Núria Casellas, Rossella Rubino, and Govanni Sartor, editors, *Computable Models of the Law*, number 4884 in Lecture Notes in Computer Science, pages 162–184. Springer Verlag, 2008.

39. Ann Macintosh, Thomas F. Gordon, and Alastair Renton. Providing argument support for e-participation. *Journal of Information Technology & Politics*, 6(1):43–59, 2009.

40. Thomas F. Gordon and Douglas Walton. Proof burdens and standards. In Iyad Rahwan and Guillermo Simari, editors, *Argumentation in Artificial Intelligence*. Springer-Verlag, Berlin, Germany, 2009.

41. Trevor Bench-Capon and Thomas F. Gordon. Isomorphism and argumentation. In Carole D. Hafner, editor, *Proceedings of the 12th International Conference on Artificial Intelligence and Law (ICAIL 2009)*, New York, NY, USA, 2009. ACM Press.

42. Thomas F. Gordon and Douglas Walton. Legal reasoning with argumentation schemes. In Carole D. Hafner, editor, *12th International Conference on Artificial Intelligence and Law (ICAIL 2009)*, New York, NY, USA, 2009. ACM Press.

Appendix C

Complete List of Publications

Journal Articles

1. Thomas F. Gordon. Künstliche Intelligenz und Recht (Teil 2). *Jur PC*, 6:638–639, 1990.

2. Thomas F. Gordon. Künstliche Intelligenz und Recht. *Jur PC*, 5:605–608, 1990.

3. Thomas F. Gordon. An abductive theory of legal issues. *International Journal of Man-Machine Studies*, 35:95–118, 1991.

4. Thomas F. Gordon. The Pleadings Game — an exercise in computational dialectics. *Artificial Intelligence and Law*, 2(4):239–292, 1994.

5. Nikos Karacapilidis, Dimitris Papadias, Thomas F. Gordon, and Hans Voss. Collaborative environmental planning with geomed. *European Journal of Operational Research, Special Issue on Environmental Planning*, 102(2):335–346, 1997.

6. Thomas F. Gordon and Nikos Karacapilidis. The Zeno argumentation framework. *Künstliche Intelligenz*, 99(3):20–29, 1999.

7. Thomas F. Gordon, Angi Voss, Gernot Richter, and Oliver Märker. Zeno: Groupware for discourses on the internet. *Künstliche Intelligenz*, 2(1):43–45, 2001.

8. Thomas F. Gordon. Die Bedeutung von E-Governance für de öffentliche Verwaltung. *Verwaltung und Management*, 5:258–263, 2004.

9. Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10-11):875–896, 2007.

10. Thomas F. Gordon. Visualizing Carneades argument graphs. *Law, Probability and Risk*, page doi: 10.1093/lpr/mgm026, 2007.

11. Ann Macintosh, Thomas F. Gordon, and Alastair Renton. Providing argument support for e-participation. *Journal of Information Technology & Politics*, 6(1):43–59, 2009.

12. Douglas Walton and Thomas F. Gordon. Jumping to a conclusion: Fallacies and standards of proof. *Informal Logic*, 29(2):215–243, 2009.

Books and Book Chapters

1. Thomas F. Gordon. Some problems with prolog as a knowledge representation language for legal expert systems. In C. Arnold, editor, *Yearbook of Law, Computers and Technology*, pages 52–67. Leicester Polytechnic Press, Leicester, England, 1987.

2. Thomas F. Gordon. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice*. Springer, New York, 1995. Book version of 1993 Ph.D. Thesis; University of Darmstadt.

3. Thomas F. Gordon. Computational dialectics. In Peter Hoschka, editor, *Computers as Assistants - A New Generation of Support Systems*, pages 186–203. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1996.

4. Thomas F. Gordon and Oliver Märker. Mediation systems. In Oliver Märker and Matthias Trénel, editors, *Online-Mediation. Neue Medien in der Konfliktvermittlung - mit Beispielen aus Politik und Wirtschaft*, pages 61–84. Edition Sigma, Berlin, 2002.

5. Thomas F. Gordon, editor. *Legal Knowledge and Information Systems: JURIX 2004, the seventeenth annual conference*, volume 120 of *Frontiers in artificial intelligence and applications*. IOS Press, Amsterdam, 2004.

6. Thomas F. Gordon. Legal knowledge systems. In Ari-Veikko Anttiroiko and Matti Mälkiä, editors, *Encyclopedia of Digital Government*, pages 1161–1166. Idea Group, Hershey, Pennsylvania, 2006.

7. Thomas F. Gordon. Constructing legal arguments with rules in the legal knowledge interchange format (LKIF). In Pompeu Casanovas, Núria Casellas, Rossella Rubino, and Govanni Sartor, editors, *Computable Models of the Law*, number 4884 in Lecture Notes in Computer Science, pages 162–184. Springer Verlag, 2008.

8. Thomas F. Gordon and Douglas Walton. Proof burdens and standards. In Iyad Rahwan and Guillermo Simari, editors, *Argumentation in Artificial Intelligence*. Springer-Verlag, Berlin, Germany, 2009.

Referred Conference Papers

1. Thomas F. Gordon. Object-oriented predicate logic and its role in representing legal knowledge. In Charles Walter, editor, *Computing Power and Legal Reasoning*, pages 163–203. West Publishing Co., St. Paul, 1985.

2. Thomas F. Gordon. The role of exceptions in models of the law. In H. Fiedler and R. Traunmüller, editors, *Formalisierung im Recht und Ansätze juristischer Expertensysteme*, pages 52–59. J. Schweitzer Verlag, Munich, 1986.

3. Thomas F. Gordon and Gerald Quirchmayr. Oblog: eine programmiersprache für juristische expertensysteme. In Ulrich Erdmann, Herbert Fiedler, Fritjof Haft, and Roland Traunmüller, editors, *Computergestützte Juristische Expertensysteme*, pages 123–134. Attempto Verlag, Tübingen, 1986.

4. Herbert Fiedler and Thomas F. Gordon. Recht und rechtsanwendung als paradigma wissensbasierter systeme. In W. Brauer Wahlster and W., editors, *Tagungsband Wissensbasierte Systeme*, pages 63–77. Springer, 1987.

5. Thomas F. Gordon. Oblog-2: A hybrid knowledge representation system for defeasible reasoning. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, pages 231–239, Boston, 1987.

6. Thomas F. Gordon. The importance of nonmonotonicity for legal reasoning. In H. Fiedler, F. Haft, and R. Traunmüller, editors, *Expert Systems in Law; Impacts on Legal Theory and Computer Law*, pages 111–126. Attempto Verlag, Tübingen, 1988.

7. Thomas F. Gordon. Issue spotting in a system for searching interpretation spaces. In *Proceedings of the Second International Conference on Artificial Intelligence and Law*, pages 157–164. Association for Computing Machinery (ACM), New York, 1989.

8. Thomas F. Gordon. Artificial intelligence: A hermeneutic defense. In Christiane Floyd Keil-Slawik, Heinz Züllighoven, Reinhard Budde, and Reinhard, editors, *Software Development and Reality Construction*, pages 280–290. Springer-Verlag, Berlin, 1992.

9. Thomas F. Gordon. A theory construction approach to legal document assembly. In Antonio A. Martino, editor, *Expert Systems in Law*, pages 211–225. North-Holland, Amsterdam, 1992.

10. Thomas F. Gordon. The Pleadings Game; formalizing procedural justice. In *Proceedings of the Fourth International Conference on Artificial Intelligence and Law*, pages 10–19. ACM Press, New York, 1993.

11. Gerhard Brewka and Thomas F. Gordon. How to buy a Porsche: An approach to defeasible decision making. In *Working Notes of the AAAI-94 Workshop on Computational Dialectics*, pages 28–38, Seattle, Washington, 1994.

12. Thomas F. Gordon. From Jhering to Alexy — using Artificial Intelligence models in jurisprudence. In H. Prakken Soeteman, A. J. Muntjewerff, and A., editors, *Legal Knowledge Based Systems, The Relation with Legal Theory*, pages 19–31. Koninklijke Vermande BV, Lelystad, 1994.

13. Thomas F. Gordon and Nikos Karacapilidis. The Zeno argumentation framework. In *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*, pages 10–18, Melbourne, Australia, 1997. ACM Press.

14. Thomas F. Gordon, Nikos Karacapilidis, Hans Voss, and Andreas Zauke. Computer-mediated cooperative spatial planning. In H. Timmermans, editor, *Decision Support Systems in Urban Planning*, pages 299–309. E. & F.N. Spon, London, 1997.

15. Henry Prakken and Thomas F. Gordon. Rules of order for electronic group decision making — a formal methodology. In Julian A. Padget, editor, *Collaboration between Human and Artificial Societies*, Springer Lecture Notes in AI, pages 246–263. Berlin, 1999.

16. Thomas F. Gordon and Oliver Märker. Mediation systems. In Lorenz M. Hilty and Paul W. Gilgen, editors, *Sustainability in the Information Society; 15th International Symposium Informatics for Environment Protection*, pages 737–742, Zürich, 2001. Metropolis Verlag.

17. Thomas F. Gordon and Gernot Richter. Discourse support systems for deliberative democracy. In Roland Traunmüller and Klaus Lenk, editors, *eGovernment: State of the Art and Perspectives (EGOV02)*, pages 248–255, Aix-en-Provence, 2002. Springer Verlag.

18. Oliver Märker, Hans Hagedorn, Matthias Trénel, and Thomas F. Gordon. Internet-based citizen participation in the City of Esslingen. relevance — moderation — software. In Manfred Schrenk, editor, *CORP 2002 - "Who plans Europe's future?"*. Technical University

of Vienna, 2002. 7th international symposion on information technology in urban and regional planning and impacts of ICT on physical space.

19. Thomas F. Gordon. On the need for standard document formats for egovernance applications of legal knowledge-based systems. In *Workshop on Artificial Intelligence to Support eGovernment*, Cambridge, UK, 2003. Workshop at the The Twenty-third SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence (AI-2003). Talk only. Workshop proceedings were not published.

20. Thomas F. Gordon. An open, scalable and distributed platform for public discourse. In Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, and Wolfgang Wahlster, editors, *Informatik 2003*, volume 2, pages 232–234, Frankfurt am Main, 2003. Springer Verlag.

21. Thomas F. Gordon. A use case analysis of legal knowledge-based systems. In Danièle Bourcier, editor, *Legal Knowledge and Information Systems (JURIX 2003)*, Frontiers in Artificial Intelligence and Applications, pages 81–90, Utrecht, 2003. IOS Press, Amsterdam.

22. Olivier Glassey, Thomas Gordon, and Jonas Pattberg. Machbarkeitstudie eines wissenbasierten Rechtsberatungssystems im Kreis Herford. In Erich Schweighofer, Doris Liebwald, Silvia Augeneder, and Thomas Menzel, editors, *Effizienz von e-Lösungen in Staat und Gesellschaft; Aktuelle Fragen der Rechtsinformatik*, pages 379–386. Boorberg-Verlag, 2005.

23. Thomas F. Gordon. Artificial intelligence and legal theory at law schools. In Hajime Yoshino, Kevin D. Ashley, and Katsumi Nitta, editors, *Artificial Intelligence and Legal Education*, pages 53–58, Bologna, Italy, 2005. Gedit Edizioni.

24. Thomas F. Gordon. A computational model of argument for legal reasoning support systems. In Paul E. Dunne and Trevor Bench-Capon, editors, *Argumentation in Artificial Intelligence and Law*, IAAIL Workshop Series, pages 53–64, Nijmegen, The Netherlands, 2005. Wolf Legal Publishers.

25. Thomas F. Gordon. Information technology for good governance. In Daniele Bourcier, editor, *French-German Symposium on Governance, Law and Technology*, pages 87–95. University of Paris, September 2005.

26. Douglas Walton and Thomas F. Gordon. Critical questions in computational models of legal argument. In Paul E. Dunne and Trevor Bench-Capon, editors, *Argumentation in Artificial Intelligence and Law*, IAAIL Workshop Series, pages 103–111, Nijmegen, The Netherlands, 2005. Wolf Legal Publishers.

27. Thomas F. Gordon and Douglas Walton. The Carneades argumentation framework — using presumptions and exceptions to model critical questions. In Paul E. Dunne and Trevor J.M. Bench-Capon, editors, *Computational Models of Argument. Proceedings of COMMA 2006*, pages 195–207, Amsterdam, September 2006. IOS Press.

28. Thomas F. Gordon and Douglas Walton. Pierson vs. Post revisted — a reconstruction using the Carneades Argumentation Framework. In Paul E. Dunne and Trevor Bench-Capon, editors, *Proceedings of the First International Conference on Computational Models of Argument (COMMA 06)*, Liverpool, 2006. IOS Press.

29. Thomas F. Gordon. Constructing arguments with a computational model of an argumentation scheme for legal rules. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Law*, pages 117–121, 2007.

30. Thomas F. Gordon. Hybrid reasoning with argumentation schemes. In *Proceedings of the 8th Workshop on Computational Models of Natural Argument (CMNA 08)*, pages 16–25, Patras, Greece, July 2008. The 18th European Conference on Artificial Intelligence (ECAI 2008).

31. Trevor Bench-Capon and Thomas F. Gordon. Isomorphism and argumentation. In Carole D. Hafner, editor, *Proceedings of the 12th International Conference on Artificial Intelligence and Law (ICAIL 2009)*, New York, NY, USA, 2009. ACM Press.

32. Thomas F. Gordon and Douglas Walton. Legal reasoning with argumentation schemes. In Carole D. Hafner, editor, *12th International Conference on Artificial Intelligence and Law (ICAIL 2009)*, New York, NY, USA, 2009. ACM Press.

33. Marc Lauritsen and Thomas F. Gordon. Toward a general theory of document modeling. In Carole D. Hafner, editor, *12th International Conference on Artificial Intelligence and Law (ICAIL 2009)*, New York, NY, USA, 2009. ACM Press.

Ph.D. Thesis

1. Thomas F. Gordon. *The Pleadings Game; An Artificial Intelligence Model of Procedural Justice*. Ph.d., Darmstadt University of Technology, 1993.