On Building Argumentation Schemes Using the Argument Interchange Format

Iyad Rahwan,[1,2,*] **Chris Reed,**[3,†] **Fouad Zablith**[1,‡]
[1]Institute of Informatics, British University in Dubai, P.O.Box 502216, Dubai, UAE
[2](Fellow) School of Informatics, University of Edinburgh, UK
[3]School of Computing, University of Dundee, UK

Abstract

In this paper, we present an extension to the recently proposed Argument Interchange Format (AIF) to capture Walton's argument schemes.

1 Introduction

Argumentation-based techniques and results have found a wide range of applications in both theoretical and practical branches of artificial intelligence and computer science [Prakken and Vreeswijk, 2002; Chesñevar *et al.*, 2000; Carbogim *et al.*, 2000]. One area that witnessed significant growth is *argumentation-support systems*. Many systems are now available for users to create and represent arguments, such as Araucaria [Rowe *et al.*, 2003], truthmapping [truthmapping, 2006], Compendium [Bachler *et al.*, 2003], Reason!Able [Gelder, 2002], and others.

Any system for argumentation and deliberation support must have a way to describe arguments and their interrelationships. This is often done using mark-up languages. For example, Araucaria uses an XML-based Argument Markup Language (AML). These mark-up languages do not have rich formal semantics, and are therefore not designed to enable sophisticated automated processing of argumentative statements. Such semantics may help improve applications of electronic deliberative democracy [Atkinson *et al.*, 2006; Lüehrs *et al.*, 2001] by enabling citizens to annotate, query and navigate arguments and elements of arguments. Rich formal semantics may also improve capabilities for argumentation among autonomous software agents [Parsons *et al.*, 1998; Rahwan *et al.*, 2003; Sadri *et al.*, 2002] by enabling the exchange arguments in open multi-agent systems using a standardised format.

In response to the above, an effort towards a standard Argument Interchange Format (AIF) has recently commenced [Chesñevar *et al.*, 2006]. The aim was to consolidate the work that has already been done in argumentation mark-up languages and multi-agent systems frameworks. It was hoped that this effort will provide a convergence point for theoretical and practical work in this area, and in particular facilitate: (i) argument interchange between agents within a particular multi-agent framework; (ii) argument interchange between agents across separate multi-agent frameworks; (iii) inspection/manipulation of arguments through argument visualisation tools; and (iv) interchange between argumentation visualisation tools.

In this paper, we explore extending the AIF in order to express argument schemes in general, and Walton's schemes in particular [Walton, 1996]. This is part of an ongoing work on building a Semantic Web-based system for argument representation and manipulation.

The paper is organised as follows. In the next section, we give a brief summary of the AIF. In Section 3, we describe our extensions to the core AIF ontology, followed by an example in Section 4. We then summarise the ontology in Section 5 and conclude the paper in Section 6.

2 Background: The Argument Interchange Format (AIF) Core Ontology

In this section, we provide a brief overview of the current state of the Argument Interchange Format [Chesñevar *et al.*, 2006]. The AIF is a core ontology of argument-related concepts. This core ontology is specified in such a way that it can be extended to capture a variety of argumentation formalisms and schemes. To maintain generality, the AIF core ontology assumes that argument entities can be represented as nodes in a directed graph (di-graph). This di-graph is informally called an *argument network*.

2.1 Argument Representation

Arguments are represented in the system using a set $Nodes$ of nodes connected by edges. There are 2 types of nodes: the *information nodes* (I-Nodes) which hold pieces of information or data, and *scheme nodes* (S-Nodes) representing the arguments' scheme. These are represented by two disjoint sets of concepts, $INode$ and $SNode$ respectively, where $INode \cup SNode = Nodes$ and $INode \cap SNode = \emptyset$. We describe the nodes in more detail below, referring to Figure 1, which visualises the classes of the AIF ontology and their interrelationships.

Information nodes are used to represent *passive* information used in an argument, such as a piece of information that acts as a claim, premise, data etc. On the other hand, S-Nodes capture the application of *schemes* (i.e. patterns of

*irahwan@acm.org
†chris.reed@computing.dundee.ac.uk
‡fouad@zablith.org

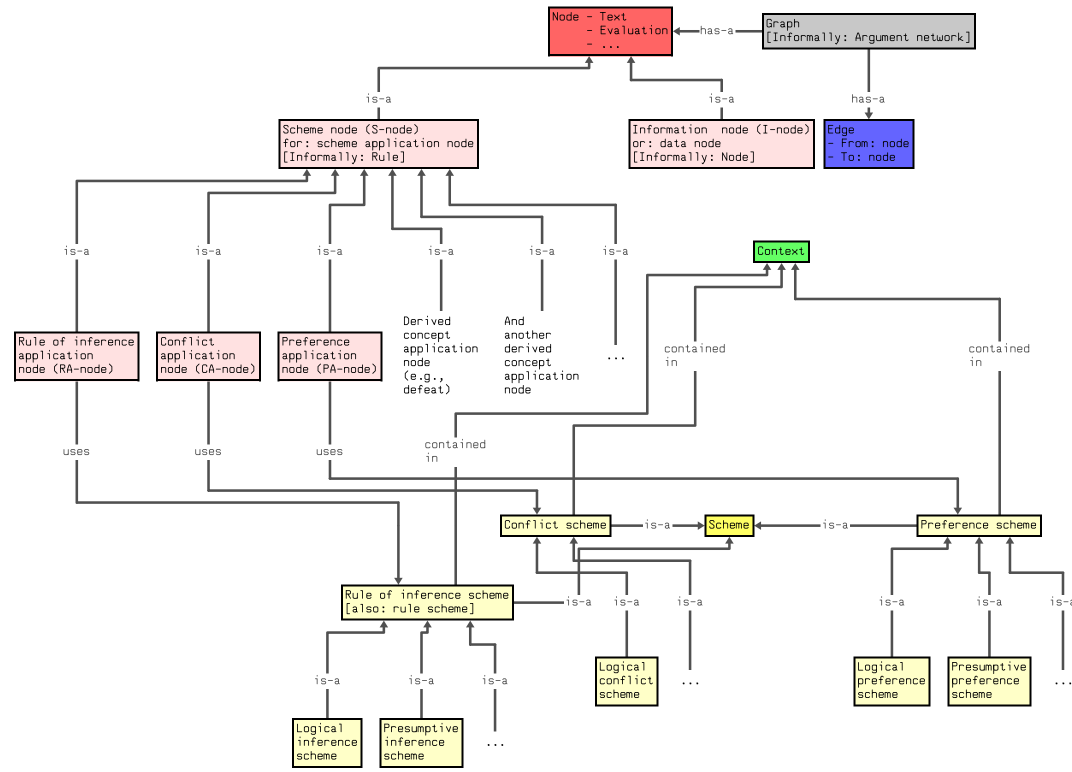

Figure 1: Concepts and Relations in the AIF Ontology

reasoning). Such schemes may be considered as domain-independent patterns of reasoning (which resemble rules of inference in deductive logics but broadened to include non-deductive logics that are not restricted to classical logical inference). The present ontology deals with three different types of schemes, namely *rule of inference application* (RA), *preference application* (PA) and *conflict application* (CA). Intuitively, the set of inference application nodes $RANode$ captures nodes that represent (possibly non-deductive) rules of inference. The set of conflict application nodes $CANodes$ captures applications of criteria (declarative specifications) defining conflict (e.g. among a proposition and its negation, among values etc.). Finally, the set of nodes $PANode$ are applications of (possibly abstract) criteria of preference among evaluated nodes. These sets of nodes are disjoint: $RANode \cap PANode \cap CANode = \emptyset$.

2.2 Edges

The edges connecting the different arguments' parts have been presented as classes' attributes. Formally, we define the class of edges as a predicate $edge : Node \times Node$. In the original AIF specification, edges are not typed. Instead, their semantics is understood implicitly from the types of nodes they connect. The informal semantics of edges are listed in Table 1. Broadly speaking, an edge coming out from an I-Node to an S-Node (that is RA-Node, PA-Node or CA-Node) class represents the fact that information is provided as an input to an inference rule. On the other hand, an edge coming out from an S-Node to an I-Nodes represents that the inference rule defined in the former is used to infer the information in the latter.

One of the restrictions imposed by the AIF is that no outgoing edge from an I-Node can be directed directly to another I-Node. Formally: $\nexists(i, j) \in edge$ where both $i \in INode$ and $j \in INode$. This ensurs that any connection between two information items must be explicitly typed through an intermediate S-Node.

Note that S-to-S edges allow us to represent what might more properly be considered as modes of 'meta-reasoning.' For example, RA-to-RA and RA-to-PA edges might indicate some kind of meta-justification for application of an inference rule or particular criterion for defining preferences. Some instances of Toulmin backings [Toulmin, 1958], for example, could most accurately be captured through the use of RA-to-RA links. An RA-to-CA node could encode some rationale for why two I-nodes are in conflict. For example, that each I-node specifies two alternative actions for realising a goal (in which case arguments supporting each action are considered to be in conflict). Of course, once we consider these forms of meta-reasoning, then this paves the way for 'meta-argumentation' in that two preference applications might be in conflict (PA-to-CA and CA-to-PA), requiring the definition of a preference between preference applications (PA-to-PA) [Modgil, 2006].

An attack from one information or scheme node to another information or scheme node can be captured through a "CA-

	to *I-node*	to *RA-node*	to *PA-node*	to *CA-node*
from *I-node*		I-node data used in applying an inference	I-node data used in applying a preference	I-node data in conflict with information in node supported by CA-node
from *RA-node*	inferring a conclusion in the form of a claim	inferring a conclusion in the form of an inference application	inferring a conclusion in the form of a preference application	inferring a conclusion in the form of a conflict definition application
from *PA-node*	applying a preference over data in I-node	applying a preference over inference application in RA-node	meta-preferences: applying a preference over preference application in supported PA-node	preference application in supporting PA-node in conflict with preference application in PA-node supported by CA-node
from *CA-node*	applying conflict definition to data in I-node	applying conflict definition to inference application in RA-node	applying conflict definition to preference application in PA-node	showing a conflict holds between a conflict definition and some other piece of information

Table 1: Informal semantics of untyped edges in the core AIF

Node," which is a type of S-Node. A CA-Node captures the type of attack. The attacker is linked to the CA-Node through an edge. The CANode is then linked to the attacked node via another edge. Symmetric conflicts are captured using two CANodes, one per attack direction.

3 Extending the Core AIF: Representation of Argumentation Schemes

The core AIF ontology described in the previous section is very general and expressive. However, it is not sufficiently detailed in order to readily express arguments following a particular style of argument analysis. In this Section, we explore how the AIF can be extended to express arguments organised based on Walton's argument schemes.

An argumentation scheme is a representation of a common form of argument, capturing its the premise-conclusion and inference structures [Walton, 1996]. The concept of schemes is an idea towards the categorisation of the way arguments should be built, and offer a common understanding of arguments' structures. Each Walton scheme type has a name, conclusion, set of premises and a set of critical questions bound to this scheme. Critical questions are a way to let the user know about the weakness of the arguments in this scheme, and give a way for others to attack those arguments.

A common example of Walton-style schemes is the Argument from Expert Opinion [Walton, 1996]:

- **Major Premise:** Source E is an expert in the subject domain S containing proposition A.
- **Minor Premise:** E asserts that proposition A in domains S is true.
- **Conclusion:** A may plausibly be taken to be true.

Schemes in our extended ontology are represented as class instances and not as classes. This offers the possibility for the user to add new schemes from the system interface, without having to modify the ontology itself.[1] Moreover, we use typed edges. This makes the interpretation of different relationships more explicit, which can improve diagram visualisation and processing.

The class "SchemeDescription" is the main class handling the main type of the schemes. It has three subclasses: the "ConflictScheme," "PreferenceScheme" and "RuleScheme." The "SchemeDescription" general class has just a single attribute:

[1]This allows for functionality similar to Araucaria's "schemeset" construction

- *hasSchemeName*: of type "string", it's a slot for entering the name of the scheme, having at most one value.

Walton's schemes [Walton, 1996] are considered instances of the "PresumptiveInferenceScheme," and are used as the exemplar throughout this paper. The framework can support arbitrary scheme definitions, offering a direct mapping of the five schemesets currently available in Araucaria (the only other implemented software system that supports schemes), and in principle, others besides.[2]

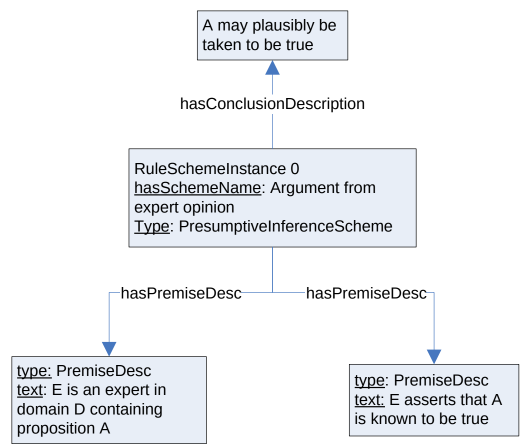

Figure 2: The base for a network representation of the scheme for "Argument from Expert Opinion"

This scheme design will help guide a user in specifying which scheme his/her argument belongs to, as in the ontology itself he/she will have scheme instances that act as examples to clarify how to build his/her argument. A given scheme (an "instance") is specified in part by listing its premises, conclusion and name. By way of example, the Walton's account of the scheme for "Argument from expert opinion" is characterised as the graph in Figure 2.

One of the challenges for any computational account of argumentation schemes is how the presumptions that accompany the scheme are captured. In Walton's approach, these presumptions are (part) of the role of critical questions. Critical questions function to test the presumptions behind the

[2]Araucaria caters for schemes in the style of Walton [Walton, 1996], Perelman and Olbrechts-Tyteca [Perelman and Olbrechts-Tyteca, 1969], Pollock [Pollock, 1995], Katzav and Reed [Reed and Katzav, 2004], and Grennan [Grennan, 1997].

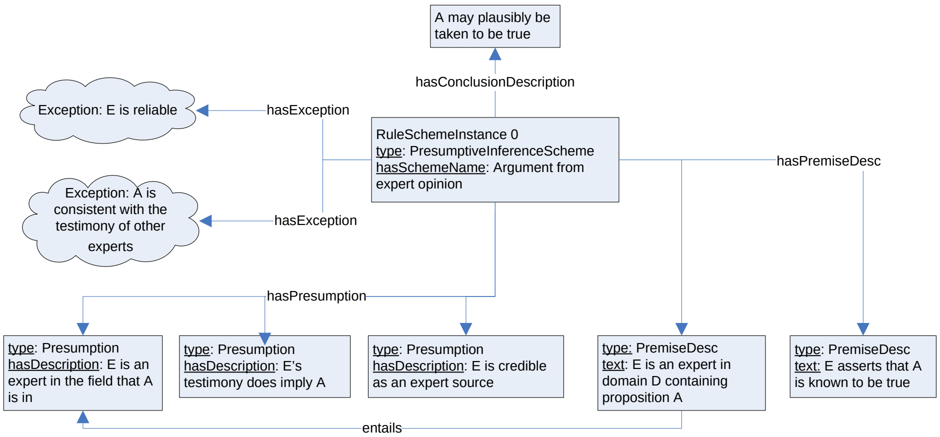

Figure 3: The extended representation of the scheme for "Argument from Expert Opinion"

warrant of a scheme, and to probe the exceptions by which it might default. So, for example, in the canonical scheme for "Argument from expert opinion," there are six critical questions:

1. *Expertise Question:* How credible is expert E as an expert source?
2. *Field Question:* Is E an expert in the field that the assertion, A, is in?
3. *Opinion Question:* Does E's testimony imply A?
4. *Trustworthiness Question:* Is E reliable?
5. *Consistency Question:* Is A consistent with the testimony of other experts?
6. *Backup Evidence Question:* Is A supported by evidence?

As Prakken et al [Prakken *et al.*, 2005] and Gordon and Walton [Gordon and Walton, 2006] have argued that these questions are not all alike. The first, second, third and sixth questions function as assumptions that the speaker makes, or, more accurately, *presumptions* required for the inference to go through. The proposer of the argument retains the burden of proof if these questions are asked. Numbers four and five, however, are somewhat different in that if asked, the burden of proof shifts (*ceteris paribus*) to the questioner. They capture *exceptions* to the general rule, and correspond well to the Rebuttal in Toulmin's [Toulmin, 1958] model of argument and its computational interpretation [Reed and Rowe, 2005].

The Carneades model [Gordon and Walton, 2006] is by far the most developed in terms of accounting representationally for these two distinct forms of implicit information present in schemes. Unfortunately for our purposes, however, Carneades lacks an explicit semantics, or at least, a semantics that is machine communicable. The first step in tackling the problem is to distinguish between presumptions and exceptions, and support their explicit representation.

Though we might, for simplicity's sake, be tempted to represent presumptions, exceptions, and critical questions alike as attributes on a scheme instance, this would prohibit reifying their contents and re-using them in argument networks, which is something we shall later want to do. So for both presumptions and exceptions, we create new structures in much the same way as we have done for premise descriptions. Continuing the example from above, the full structure for Argument from Expert Opinion is shown in 3.

Note that in this way, there is no longer any need to represent critical questions directly. Since, in Walton's account, all the presumptions and exceptions of a scheme can be questioned, the set of critical questions is now inferable from the structure in 3, *viz.*, for every presumption or exception i, in a scheme instance, that scheme instance can be said to have a critical question "Is it the case that i?"

Note that in Walton's account of schemes, some presumptions are somewhat related to certain premises. More specifically, a presumption may be implicitly or explicitly *entailed* by a premise. For example, the premise "Source E is an expert in subject domain S containing proposition A" entails the presumption that "E is an expert in the field that A is in" and the presumption that "E's assertion is based on evidence." While the truth of a premise may be questioned directly, questioning associated with the underlying presumptions can be more specific, capturing the nuances expressed in Walton's characterisation. And we want to capture this relationship between some premises and presumptions explicitly, as it can allow us to guide users in their critical questioning. Thus we make use of a predicate $entails : PremiseDescription \times Presumption$. Note, however, that not every presumption entails a particular premises, since some presumptions capture implicit assumptions underlying the whole scheme.

The graph in Figure 3 is not quite complete, however, as it does not account for the detailed representation of exceptions. One alternative would be to view exceptions in exactly the same way, and simply introduce a new type, as we have done for presumptions. The AIF, however, offers a much more powerful possibility. The clue comes from noting that exceptions function in a similar way to Toulmin Rebuttals: exceptions are providing a way to challenge the use of an argument scheme. The function of challenging corresponds to the notion of a ConflictScheme in the AIF. In just the same way that stereotypical patterns of the passage of deductive, inductive

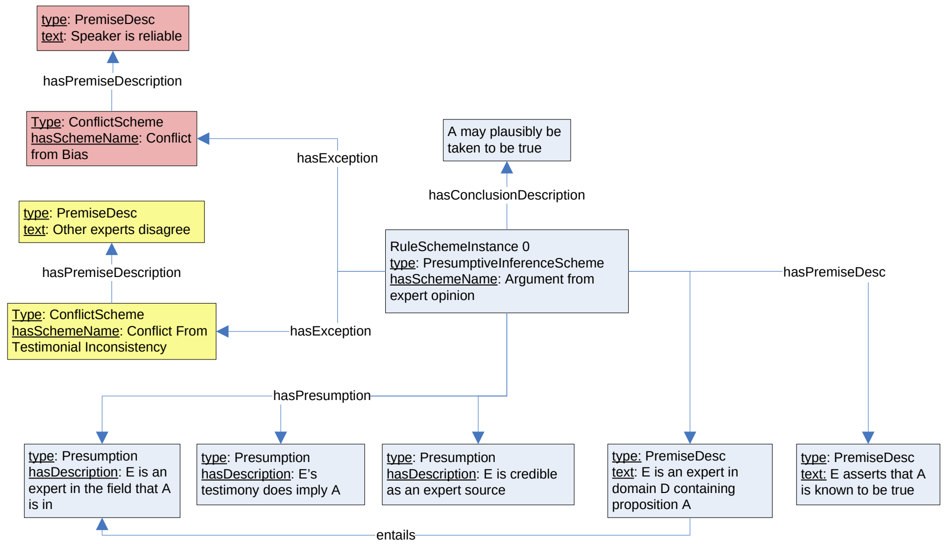

Figure 4: A graphical representation of the scheme instance for "Argument from Expert Opinion" including the conflicts corresponding to its exceptions

and presumptive inference can be captured as a RuleScheme, so too, the stereotypical ways of characterising conflict can be captured as a ConflictScheme. Conflict, like inference, has some patterns that are reminiscent of deduction in their absolutism (such as the conflict between a proposition and its converse), as well as others that are reminiscent of nondeduction inference in their heuristic nature (such as the conflict between two courses of action with incompatible resource allocations). By providing a way to attack an argumentation scheme, exceptions can most accurately – and most expressively – be presented as conflict scheme descriptions. The picture in Figure 3 can thus be completed as in Figure 4.

Of course, there is nothing to stop individual ConflictScheme instances (such as that named ConflictFromBias in Figure 4) being used elsewhere - ConflictFromBias, for example, might have a role in argument schemes from witness testimony in general. They may also have a role divorced from RuleSchemes entirely - ConflictFromTestimonialInconsistency might be used independently to throw doubt on some conclusion not supported by testimonial arguments.

To show how these ontological structure govern and account for instantiated arguments, the next section links the picture in Figure 4 to a simple example.

4 An Example

Consider the following example that follows the expert opinion argumentation scheme:

- Alice's Argument:
  Conclusion:
  - Brazil has the best football team
  Premises:
  - Allen says that Brazil has the best football team
  - Allen is an expert in sports
  - Football is within sports
- Bob's Challenge
  Undermining Moves
  - But Allen's biased *and*
  - Allen is not an expert in sports

The graph of this argument, which uses the scheme for "Argument from Expert Opinion," is visualized in Figure 5. It represents a surprisingly complex analysis for what appears to be a simple text. The reason for this is that the ontological superstructure needs to capture not only the content of the argument but also all the "growth points" at which new arguments might be added.

The first thing to notice about Figure 5 is that there are three distinct levels of analysis. At the bottom of the Figure, are the components that instantiate real arguments - these are the actual premises, conclusions, inferences, conflicts and other components used in the expression of the argument. On the upper side of the Figure (in blue colour) lies an intermediate level describing the types of inference (i.e. the RuleScheme instances), the types of conflict (i.e. the ConflictScheme instances) and the types of I-nodes (i.e. the presumptions, premises and conclusions).[3] Finally the true ontological layer is part of the core ontology (and is not shown in the Figure). This layer simply views PresumptiveInferenceScheme as a general class with many instances, Presumption as a general class with many instances, and so on. The ontology level is thus providing the types for the nodes at the description level, which is in turn providing the specific analytical and generative material for the argument level. This tripartite approach is important in order to provide an AIF implementation that is both able to provide operationalisable tools for argument construction and analysis, and also able to interact with other AIF implementations that make use of different description level data (different schemesets, for example).

[3]This level would also include PreferenceScheme instances if there were any.

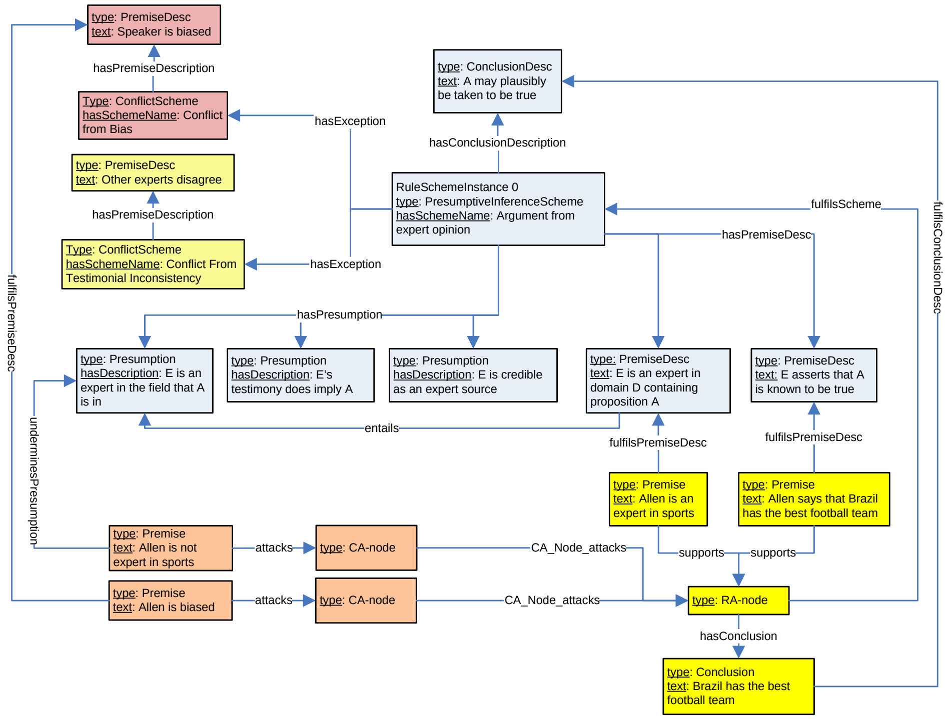

Figure 5: A full example with an argument, two attackers, and the schemes used by both the argument and the attackers. *Alice:* Brazil has the best football team: Allen's a sports expert and he says they do; *Bob:* Yes, but Allen's biased, and he's not an expert sports!

To link the argument level to the description level, it is important to represent how the nodes in the argument fulfil the roles in the various schemes and node types that are available. This is the role of the "fulfils" attribute that is available on "Node" and therefore inherited by all "PA-Nodes," "CA-Nodes," "RA-Nodes," and "I-Nodes." The "fulfils" attribute does not have unary cardinality, as might be expected, because some I-nodes fulfil both premises and presumptions. In Walton's account of critical questions, unpacked in Prakken et al.'s [Prakken *et al.*, 2005] exploration of burdern of proof, presumptions are marked by critical questions, over and above the content of the premises of a scheme. Verheij [Verheij, 2005] has argued that answers to critical questions that are entailed by premises are redundant, and there is even evidence that Walton concurs [Gordon and Walton, 2006]. The presence of sub-critical questions, the subtle differences in phrasing of critical questions and premises, and the very need to identify critical questions uniquely all suggest to us, however, that presumptions should be marked in a scheme explicitly, even when they coincide partly or fully with a premise definition. This means, however, that a claim in an argument that makes a presumption explicit (claiming, for example, that football is indeed a sport) is fulfilling two roles - that of a premise in the scheme and that of a presumption in the same scheme. Of course, it may also be that a single node is playing roles in multiple schemes simultaneously, so that too requires that a node might have multiple "fulfils" links.

5 Summary of the Extended Ontology

Figure 5 showed an *instance* of an argument and two attackers using the new concepts presented in this paper. Three distinct levels of analysis can be identified. Firstly, at the bottom right of the Figure, are the components that instantiate real arguments –these are the actual premises, conclusions, inferences, conflicts and other components used in the expression of the argument. Secondly, further up, across most of the rest of the Figure lies an intermediate level describing the types of inference (i.e. the RuleScheme instances), the types of conflict (i.e. the ConflictScheme instances) and the types of I-Nodes (i.e. the presumptions, premises and conclusions).[4] Note that this level is also expressed in terms of instances (

Thirdly, the true ontological level is part of the extended core ontology, described in Figure 6. This layer simply views a "presumptive inference scheme" as a general class with many instances, "presumption" as a general class with many instances, and so on. The ontology level is thus providing the types for the nodes at the description level, which is in turn providing the specific analytical and generative material for the argument level. This tripartite approach is important in order to provide an AIF implementation that is both able to provide operationalisable tools for argument construction and analysis, and also able to interact with other AIF implementations that make use of different description level data (different schemesets, for example).

[4]This level would also include PreferenceScheme instances if there were any.

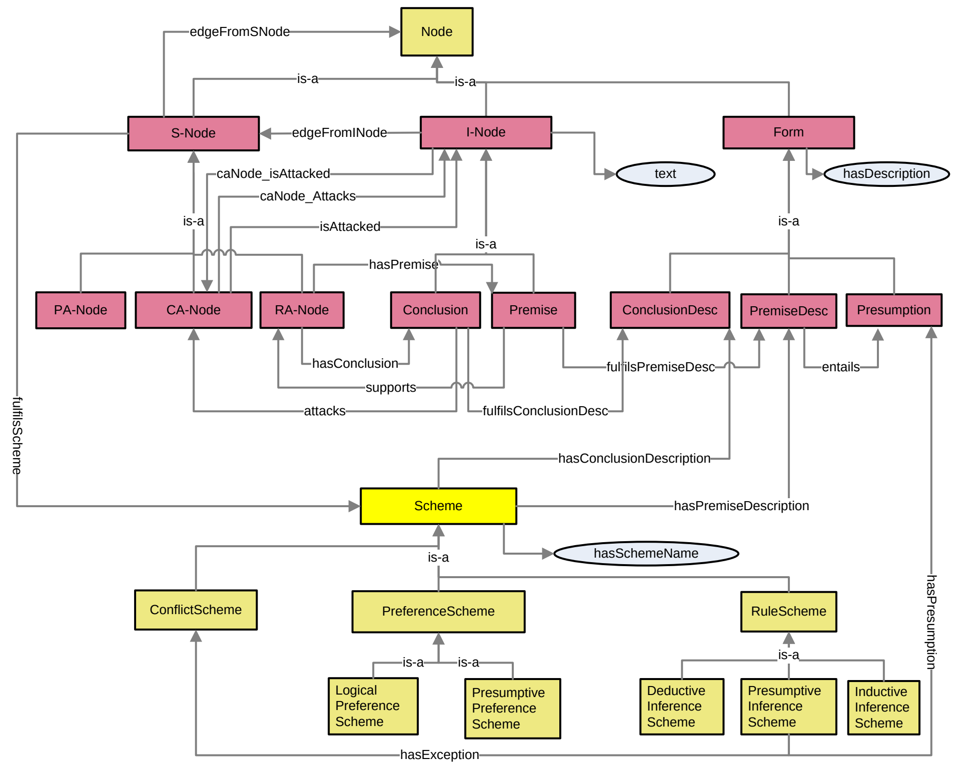

Figure 6: The Extended AIF Ontology

6 Conclusions and Open Issues

With our account of presumptions, premises and exceptions, it becomes possible programmatically to construct a mechanism for presuming. Consider a case in which a user constructs an argument using a scheme which has presumptions, but fails to explicitly add premises corresponding to those presumptions. It could be that this scenario is quite common - after all, presumptions are usually presumed rather than stated. In this case, it is a simple matter to identify the fact that there are no I-nodes that fulfil the presumptions of the scheme. It is also straightforward to then add I-nodes corresponding to those implicit presumptions. In the same way that Araucaria [Rowe *et al.*, 2003] allows statements to be marked as "implicit" in analysis, so the AIF can support a similar attribute on I-nodes. System-added I-nodes might therefore be marked as "implicit." The construction of the content of these added nodes need not pose any natural language challenge. If an interface supports user input of fine-grained atoms into a scheme (such as the name of the expert E in the scheme for Argument from Expert Opinion), then the added node can be constructed according to a detailed template. So for example, if the user has entered "Allan" for the expert E and "sports" for the domain S, the system could construct the presumption "Allan is an expert in sports" from the scheme-dependent presumption template "E is an expert in S." Alternatively, a more vague, canned presumption could be added (such as "the cited authority is an expert in the right field") and the context be sufficient for the user to mentally fill in the blanks.

With the system explicitly performing the act of "presuming" in this way, the argument can be presented to the user with the presumptions made accessible, allowing for challenge or exploration of those presumptions by which the argument inference is warranted. This, of course, is exactly the role that Walton envisaged for his critical questions.

References

[Atkinson *et al.*, 2006] Katie Atkinson, Trevor Bench-Capon, and Peter McBurney. PARMENIDES: facilitating deliberation in democracies. *Artificial Intelligence and Law – T. van Engers and A. Macintosh (editors), Special Issue on eDemocracy*, page (to appear), 2006.

[Bachler *et al.*, 2003] M. Bachler, S. Buckingham Shum, D. De Roure, D. Michaelides, and K. Page. Ontological mediation of meeting structure: Argumentation, annotation, and navigation. In *Proceedings of the 1st International Workshop on Hypermedia and the Semantic Web (HTSW2003), Nottingham, UK*, 2003.

[Carbogim *et al.*, 2000] D. Carbogim, D. Robertson, and J. Lee. Argument-based applications to knowledge engineering. *Knowledge Engineering Review*, 15(2):119–149, 2000.

[Chesñevar *et al.*, 2000] C. I. Chesñevar, A. Maguitman, and R. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.

[Chesñevar *et al.*, 2006] C. I. Chesñevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South,

G. Vreeswijk, and S. Willmott. Towards an argument interchange format. *The Knowledge Engineering Review*, 2006.

[Gelder, 2002] T. Van Gelder. A reason!able approach to critical thinking. *Principal Matters: The Journal for Australasian Secondary School Leaders*, 34(6), 2002.

[Gordon and Walton, 2006] Thomas F. Gordon and Douglas Walton. The Carneades argumentation framework. In P. Dunne and T. Bench-Capon, editors, *Proceedings of the 1st International Conference on Computational Models of Argument (COMMA)*, pages 195–207, Amsterdam, The Netherlands, 2006. IOS Press.

[Grennan, 1997] W. S. Grennan. *Informal Logic: Issues and Techniques*. McGill Queens University Press, Montreal, Quebec, Canada, 1997.

[Lüehrs *et al.*, 2001] Rolf Lüehrs, Thomas Malsch, and Klaus Voss. Internet, discourses and democracy. In Takao Terano, Toyoaki Nishida, Akira Namatame, Shusaku Tsumoto, Yukio Ohsawa, and Takashi Washio, editors, *New Frontiers in Artificial Intelligence*, volume 2253 of *Lecture Notes in Computer Science*, pages 67–74. Springer-Verlag, Heidelberg, Germany, 2001.

[Modgil, 2006] Sanjay Modgil. Hierarchical argumentation. In *Proceedings of the 10th European Conference on Logics in Artificial Intelligence. Liverpool, UK*, 2006.

[Parsons *et al.*, 1998] Simon Parsons, Carles Sierra, and Nick Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.

[Perelman and Olbrechts-Tyteca, 1969] Chaim Perelman and Lucie Olbrechts-Tyteca. *The New Rhetoric: a treatise on argumentation*. University of Notre Dame Press, 1969.

[Pollock, 1995] John L. Pollock. *Cognitive Carpentry: a blueprint for how to build a person*. Bradford Books. MIT Press, Cambridge MA, USA, 1995.

[Prakken and Vreeswijk, 2002] Henry Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 4, pages 219–318. Kluwer Academic Publishers, Dordrecht, Netherlands, second edition, 2002.

[Prakken *et al.*, 2005] Henry Prakken, Chris Reed, and Doug N. Walton. Dialogues about the burden of proof. In *Proceedings of the 10th international conference on artificial intelligence and law (ICAIL)*, pages 115–124, New York NY, USA, 2005. ACM Press.

[Rahwan *et al.*, 2003] Iyad Rahwan, Sarvapali D Ramchurn, Nicholas R. Jennings, Peter McBurney, Simon Parsons, and Liz Sonenberg. Argumentation based negotiation. *Knowledge Engineering Review*, 18(4):343–375, 2003.

[Reed and Katzav, 2004] Chris Reed and J. Katzav. On argumentation schemes and the natural classification of arguments. *Argumentation*, 18(2):239–259, 2004.

[Reed and Rowe, 2005] Chris Reed and Glenn Rowe. Translating Toulmin diagrams: Theory neutrality in argument representation. *Argumentation*, 19:267–286, 2005.

[Rowe *et al.*, 2003] G. W. A. Rowe, C. A. Reed, and J. Katzav. Araucaria: Marking up argument. In *European Conference on Computing and Philosophy*, 2003.

[Sadri *et al.*, 2002] Fariba Sadri, Francesca Toni, and Paolo Torroni. Abductive logic programming architecture for negotiating agents. In *Proceedings of the 8th European Conference on Logics in Artificial Intelligence (JELIA-2002)*, volume 2424 of *Lecture Notes in Computer Science*, pages 419–431. Springer Verlag, Berlin, Germany, 2002.

[Toulmin, 1958] Stephen Toulmin. *The Uses of Argument*. Cambridge University Press, Cambridge, UK, 1958.

[truthmapping, 2006] truthmapping. http://www.truthmapping.com, 2006.

[Verheij, 2005] Bart Verheij. *Virtual Arguments: On the Design of Argument Assistants for Lawyers and Other Arguers*. T.M.C. Asser Press, The Hague, The Netherlands, 2005.

[Walton, 1996] D. N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Erlbaum, Mahwah NJ, USA, 1996.