University of Dundee Technical Report The Argument Interchange Format: Consolidation & Extensions

Chris Reed[*], Simon Wells[†], Floris Bex[‡], Katarzyna Budzynska[§], Joseph Devereux[¶], and John Lawrence[‖]
School of Computing,
Queen Mother Building,
University of Dundee,
Dundee, DD1 4HN,
United Kingdom.

April 21, 2010

Abstract

The Argument Interchange Format (AIF) is a nascent format for representing and sharing argument resources. Since the initial AIF meeting in Budapest in 2005, the AIF has been adopted by a variety of researchers working in a wide range of argumentation related contexts. Adoption of the AIF by these different groups has lead to a number of extensions to the original core ontology which could contribute to a discussion on the requirements of an enhanced AIF. A meeting to discuss AIF 2.0 is to be held in Scotland in 2010 and this technical report has been published in preparation and discusses the past, present, and possible future of the AIF.

Introduction

This technical report has been published in preparation for the second AIF meeting, to be held in Scotland in May 2010. The report is split into two parts. The first part summarises the existing AIF standard, bringing together the specification laid out at the first AIF meeting in Budapest in 2005 with various subsequent reifications in RDF, OWL-DL, etc. The second part deals with a number of extensions to the AIF that have

[*] creed@computing.dundee.ac.uk
[†] swells@computing.dundee.ac.uk
[‡] florisbex@computing.dundee.ac.uk
[§] k.budzynska@uksw.edu.pl
[¶] josephdevereux@computing.dundee.ac.uk
[‖] johnlawrence@computing.dundee.ac.uk

been explored in Argumentation Research Group at Dundee and which are offered as input to the AIF 2.0 meeting.

Part I
The Argument Interchange Format

1 Introduction

Argumentation is a large and diverse field stretching from analytical philosophy to communication theory and social psychology. The computational investigation of the space has multiplied that spectrum by a diversity of its own in semantics, logics and inferential systems. One of the problems associated with the diversity and productivity of the field, however, is fragmentation: with many researchers from various backgrounds focusing on different aspects of argumentation, it is increasingly difficult to reintegrate results into a coherent whole. This in turn makes it difficult for new research to build upon old. To tackle this problem, the community has initiated an effort aimed at building a common ontology for argument, which will support interchange between different research projects and applications in the area: this is the Argument Interchange Format (AIF).

The Argument Interchange Format started life as a skeleton for contributions from participants at an AgentLink Technical Forum Group (Budapest 2005) and received additional input from third parties. The results were subsequently improved and added to by online discussion to form a more substantial consensus, which was published as [9]. The moniker Argument Interchange Format is slightly misleading. The goal of the AIF project is not to develop a concrete format but rather to develop a general abstract ontology for argumentation. This ontology can then be used as an "interlingua", which allows researchers from varying backgrounds to communicate their ideas about argumentation. That said, the AIF also has clear practical goals, as it intends to aid the development of argumentation tools by facilitating the exchange of data and ideas between such tools. This basic idea is expressed by the following figure, which was taken from Verheij's input to the original meeting in Budapest [1].

Figure 1 shows that the ontology can be the centrepiece of multiple argumentation formats, where each argumentation format uses the core AIF ontology as its starting point and direct translations between argumentation formats are optional. Since the original specification, a number of extensions to the basic AIF ontology have been suggested. Modgil and McGinnis [18] as well as Reed et al. [27] have made steps to incorporate dialogue into the AIF specification, Rahwan et al. [25] have extended the AIF to be able to fully express argumentation schemes. There are also examples of specific domain knowledge being added to the core AIF: Lindgren [17] uses the AIF to structure medical knowledge like clinical guidelines and Letia and Groza [16] use an extended version of the AIF in the context of reasoning about food safety.

[1] `http://www.ai.rug.nl/~verheij/publications/aif2005.htm`

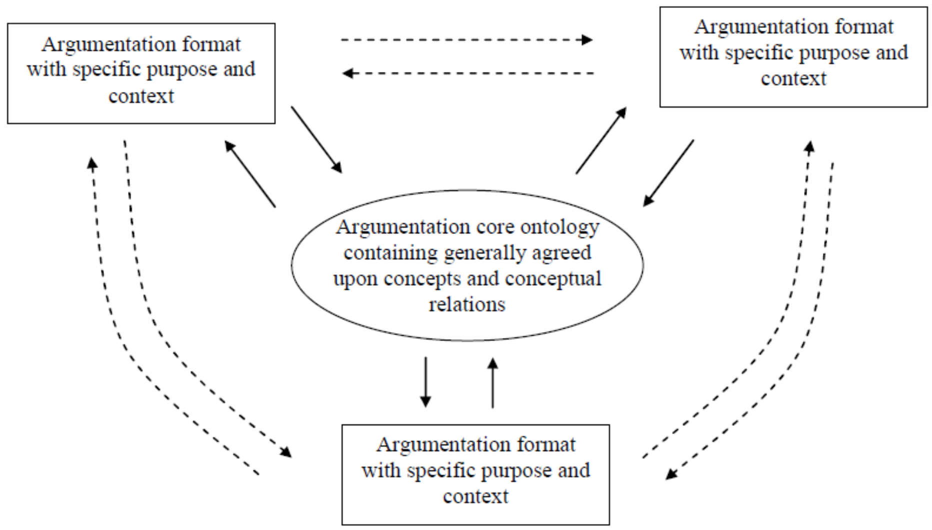

Figure 1: The Role of the AIF.

In order to be used as the basis for a software tool, the abstract ontology needs to be expressed in some machine-readable language. To this end, the abstract ontology has already been reified in various formats such as RDF Schema [25], DOT (OVAgen and AVERS, see below) and OWL-DL [22]. Currently, there are a number of tools which are explicitly or implicitly based on AIF or one of its extensions: Argument Blogging[2] ArgDF and Avicenna[3], ArgDMSS[4] AVERS[5], Carneades[6] and the OVA suite of argumentation and visualization tools[7] Other tools which may include AIF support at some

[2]Argument Blogging is the process of harvesting textual resources from the web and structuring them into distributed argumentative dialogues. A prototype system for this was developed by Gourlay and Wells, see `http://www.arg.computing.dundee.ac.uk/downloads/media/colin.avi` and [32].

[3]Systems through which users can create and query arguments, see `http://www.argdf.org/` and [25, 22]

[4]DMSS stands for Dementia Management and Support System, a clinical decision support system. A prototype system based on the (extended) AIF was implemented by Winnberg, see `http://www8.cs.umu.se/education/examina/Rapporter/PeterWinnberg\_kand.pdf`

[5]Argument Visualization for Evidential Reasoning based on Stories, a prototype sense-making and visualization tool for Dutch police analysts, see [4]

[6]An argument visualization tool originally developed for the visualization of legal reasoning. It is based on the Legal Knowledge Interchange Format (LKIF), of which the argument-graph component is based on the conceptual model of the AIF. See `http://carneades.berlios.de/` and [11]

[7]The overview of these tools is provided on `http://www.arg.dundee.ac.uk/?page\_id=143`. OVA stands for Online Visualization of Argument and is a graph-based visualization tool for constructing arguments, see `http://ova.computing.dundee.ac.uk/`. Arguments can be viewed in OVAview, see `http://ova.computing.dundee.ac.uk/OvaView/sample.php` and the arguments from the online database ArgDB (`http://argdb.computing.dundee.ac.uk/`), which are also rendered using OVAview. Finally, OVAgen allows Abstract Argument Frameworks to be constructed and visualizes the acceptability of arguments under various semantics, see http://ova.computing.dundee.ac.uk/ova-gen/

point in the future are Araucaria[8], ArgKit[9] and Cohere[10]. Finally, Rahwan et al. [25] present a vision of the World Wide Argument Web (WWAW) of interconnected arguments and debates, founded upon the AIF. This WWAW would essentially combine the various (techniques of the) software tools and thus allow for the generation, visualization, analysis and querying of arguments found on the web (e.g. in blogs, forums, web pages, reports and so on).

The current base AIF specification and its reifications mark a starting point. As experience with AIF grows, and different systems and research programmes make call upon it, the specification will inevitably shift to accommodate the broadening demands. This shifting poses two distinct problems: one in the short term, and one in the longer term. The first problem is one of bootstrapping. With a number of teams working to implement slightly different reifications of the AIF, tracking versions to ensure at least some compatibility is becoming tricky. To some extent, the core AIF ontology is stabilising, and as long as new or updated reifications remain close to the spirit of this ontology, compatibility will be improved. The second, related problem, concerns the process of solidification by which the AIF settles into stability. It is important that this solidification not happen too early: the AIF must support the theories and systems that are being developed right across the community. But on the other hand, it must also not happen too late, or we risk fragmentation and a loss of coherence, which is the raison d'être of the specification. Currently, the balance is being successfully struck informally through personal networks and regular communication. If AIF starts to scale, a control system that is less lightweight may be necessary to maintain stability of at least a common core.

The Argument Interchange Format (AIF) is a core ontology of argument-related concepts which has been formulated to provide flexibility in capturing a variety of argumentation formalisms and schemes. The core ontology is a directed graph containing two types of nodes, Information nodes (I-nodes) that represent passive information and scheme nodes (S-Nodes) that capture the application of schemes representing patterns of reasoning and relationships between sets of I-Nodes. By providing a flexible language which supports a wide range of formalisms and can be used to share structured analyses of arguments between a wide range of end-user and back-end software tools, the AIF is thus a core tool that underpins the emerging WWAW.

2 The Upper Ontology and the Forms Ontology

The AIF falls naturally into two halves: the Upper Ontology and the Forms ontology. Figure 2 visualizes the top layer of the AIF ontology; the two concepts highest in the

[8] The analysis and visualization tool Araucaria (`http://araucaria.computing.dundee.ac.uk/`, [26]) has a large user base, but is now suffering from limitations of its underlying representation and increasingly dated interface and interaction metaphors. An early alpha of Araucaria 4 is available with reusable code modules for processing AIF resources.

[9] A software library intended to assist a developer in building applications that use argumentation, see `http://www.argkit.org/`

[10] Cohere is a broad sense-making or idea management tool, which allows one to draw connections between ideas, statements and so on. See `http://technologies.kmi.open.ac.uk/cohere/` and [8]

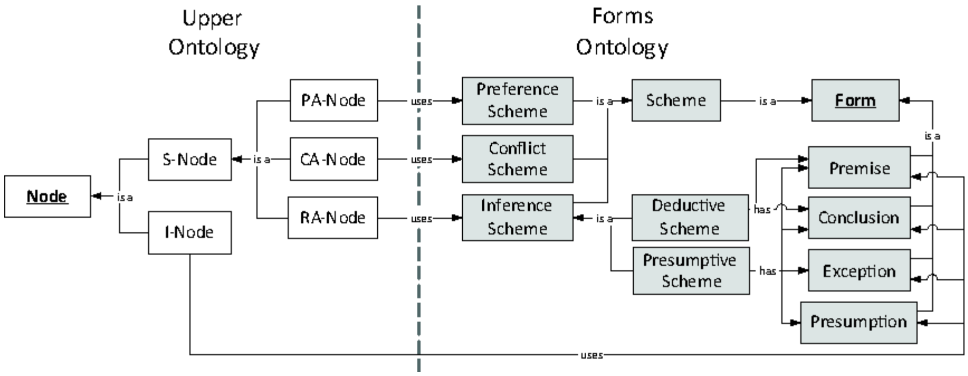

Figure 2: The AIF Ontology

hierarchy, node and form, have been emphasized

The Upper Ontology is a hierarchy of concepts linked by instantiation all the way down to the object layer - i.e all the links are 'is-a' links. The Forms ontology is a hierarchy of concepts linked by instantiation or fulfillment - i.e. some of the links are 'is-a' links, while others are 'fulfils' links. In a particular argument from expert opinion (for example), the inference *is* an instance of the argument from expert opinion scheme, while the premises *fulfil* the premise descriptors of the scheme, while each of those premise descriptors *is* a premise description.

The AIF places at its core a distinction between information, such as propositions and sentences, and schemes, the intangible mechanisms which link information, such as inference or attack[11]. Accordingly, the Upper Ontology defines two types of nodes: information nodes (I-nodes) and scheme nodes (S-nodes). Scheme nodes in turn come in several different guises: rule application nodes (RA-nodes), which correspond to applications of an inference rule, conflict application nodes (CA-nodes), which correspond to instances of an actual conflict or attack, and preference application nodes (PA-nodes), which correspond to instances of actual preferences. There are various constraints on how components interact: I-nodes, for example, can only be connected to other I-nodes via S-nodes of one sort or another. S-nodes, on the other hand, can be connected to other S-nodes directly (for example, when a reason is given for an attack, there is an edge from the RA-node to the CA-node).

Note in Figure 2 how each of the S-nodes uses a particular scheme type from the Forms Ontology: RA-nodes use Inference Schemes, CA-nodes use Conflict Schemes and PA-nodes use Preference Schemes. The different scheme types can be further classified; for example, Inference Schemes can be deductive or presumptive (i.e. defeasible). Notice how deductive schemes only have premises and conclusions. In contrast, defeasible schemes also have exceptions and presumptions[12]. Exceptions point

[11] AIF's conception of schemes is similar to, but not the same as, that of argumentation schemes. The main difference is that in the AIF schemes are not just patterns of inference but can also be other general reasoning patterns.

[12] In previous specifications of the AIF, premises were called premiseDescriptors, conclusions conclusionDescriptors and so on.

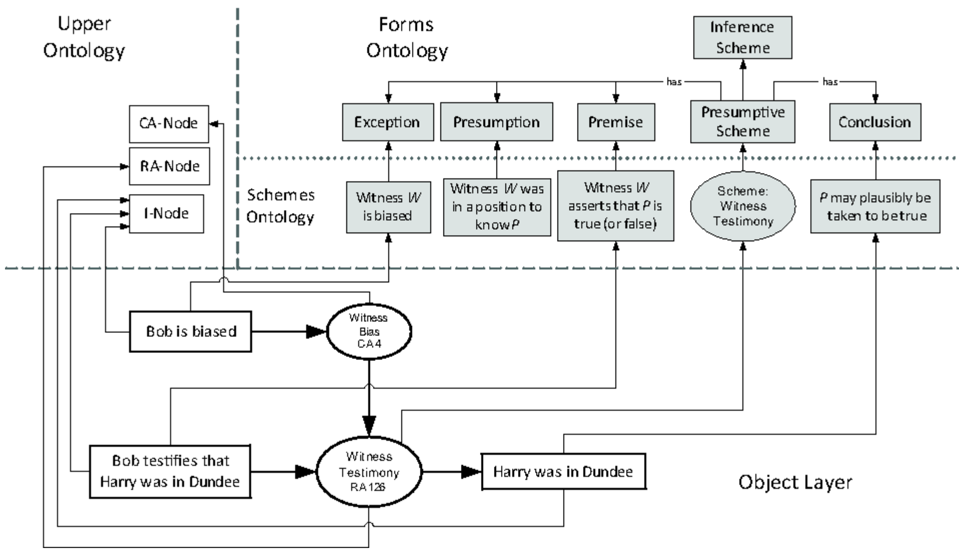

Figure 3:

to specific circumstances in which the general rule does not hold and presumptions are circumstances which are presumed to be true unless actively questioned or denied. The deductive and presumptive scheme sub-types can be further subdivided into specific argumentation schemes. For example, the set of presumptive schemes may include defeasible modus ponens, the perception scheme and the witness testimony scheme. Thus, the forms ontology is a hierarchy of concepts linked by instantiation all the way down to the object layer, which contains the actual arguments. As an example, take 3. In this figure, the actual argument is depicted at the bottom and rendered with thick lines and bigger arrows. The argument is connected to the Upper and Forms Ontologies; in this way, it can be derived what kind of graph is represented (viz. the connection to the Upper Ontology) as well as what kind of argument is represented (viz. the connection to the Forms Ontology). As in Figure 2, the forms have been rendered in grey. Any connections between the Object Layer and the ontologies or within the ontologies are is–a relations unless noted otherwise. Furthermore, the 'has' relations between the Witness Testimony Scheme (in the Schemes Ontology) and its individual parts (e.g. its premise 'Witness W asserts that P is true) have for readability not been rendered in the figure; they are essentially a mirror of the has relations between the (more abstract) Presumptive Scheme form and its parts. Finally, specific instances of schemes (i.e. S-nodes) will from now on be marked as, for example, RA 126 (meaning the 126th rule application) or CA 4 (meaning the 4th conflict application).

Figure 3 reflects the idea that the Upper Ontology and Forms Ontology together comprise a layer sitting on top of the Object Layer. Notice the Schemes Ontology, which is a sub-ontology of the Forms Ontology. This Schemes Ontology contains specific inference schemes and may vary from very simple (containing only the basic deductive and defeasible schemes) to extensive (containing a large number of specific deductive and defeasible argumentation schemes). Note also that there is no direct link between the more abstract concepts in the Forms Ontology and the Object Layer - all links are through the Schemes Ontology.

The Forms ontology is necessary for a thorough description of schemes. For example someone might make an argument from expert opinion, citing the premises (i) that Expert1 said Y; and (ii) that Expert1 is an expert. Someone else might attack this argument by claiming (iii) that the expert was biased. Without the forms ontology, the attack (iii) could not be represented. The proposition 'Expert1 is not biased' would not be represented as an I-node, because no-one has uttered it. It must instead be represented as a fulfillment of a presumption descriptor (which itself is a Presumption Description).

While both the Upper Ontology and the Forms ontology are necessary, the Forms ontology might be a more intuitive starting point for argumentation-oriented people, while the upper ontology might be a more intuitive starting-point for computation-oriented people.

2.1 Bibliography

- Towards an AIF for MAS, Willmott et al., ARGMAS 2006 [34]
- Towards an Argument Interchange Format, Chesnevar et al., KER 2006 [34]
- Rahwan et al AIJ 2007 [25]
- Rahwan COMMA 2008 [21]
- Rahwan and Reed in Rahwan and Simari book 2009 [23]

3 Reifications

Since the original AIF paper by Chesnevar et al. [9], a number of extensions to, reifications of, and implementations of the basic AIF have been given. Basically, if you view the AIF as an abstract ontology of concepts and relations between them, an extension adds new concepts to the ontology, a reification makes the concepts in the ontology more concrete by giving them a specific form in a (logical) language and an implementation uses the concepts from the ontology in a software system.

The various extensions, reifications and implementations of AIF are summarised in this section. With "representation of the core concepts," we mean the way in which the (abstract) concepts in the AIF (e.g. I-nodes, edges) are expressed, that is, the way in which the AIF is characterized. For example, in [9] all concepts are rendered as a visual, boxes-and-arrows graph. In [25] the concepts are rendered visually but also defined using set-theoretic notions (e.g. the set of I-nodes, NI, is a subset of the set of nodes, N, viz. $NI \subseteq N$). "Core concepts" stands for the AIF concepts introduced in the particular extension. "Language of the reification" is just that: the (formal) language used in that particular reification. "Implementation" is the system in which an extension and reification have been implemented.

Note that below, an extension to the core AIF usually goes hand in hand with a new reification and implementation. However, this does not mean that particular versions of the AIF, reifications and implementations are necessarily tied to each other: the AIF (or an extension of it) can be reified in several different ways and a particular reification can be implemented in multiple ways.

3.1 AIF-RDF

A reification of the AIF using RDF.

3.1.1 Summary

Representation of the abstract concepts: Boxes, arrows, natural language and a formal, set-theoretic syntax similar to DL

New Core concepts: forms, scheme descriptions

Language of the reification: RDF Schema ontology language

Implementations: ArgDF, online at *www.argdf.org*

Papers:

- Towards Representing and Querying Arguments on the Semantic Web, Rahwan and Sakeer, COMMA 06 [24]
- Laying the Foundations for a WWAW, Rahwan et al., AIJ 2007 [25]

3.2 AIF-DL

A reification of the AIF using OWL-DL.

3.2.1 Summary

New Core concepts: none

Language of the reification: OWL-DL

Implementations: Avicenna

Papers:

- Arguments in OWL: A Progress Report, Rahwan and Banihashemi, COMMA 08 [21].
- Representing and Classifying Arguments on the Semantic Web, Rahwan et al., KER 2010 [22].

3.3 AIF-DB

AIF-DB is an AIF reification which allows for the storage and retrieval of argument data in multiple forms. It consists of two main components:

1. A web service interface which allows for the addition and retrieval of elements corresponding to the top level concepts of the AIF ontology. For example nodes, edges and schemes can all be posted individually to AIF-DB. Received elements are checked for validity and that they do not currently exist in the database before adding them. The web service also offers the ability to query an AIF-DB installation for the arguments it contains.

2. An underlying database for the storage of each of these elements.

AIF-DB currently uses a MySQL database, though it is designed to support a wide range of databases with the web service abstracting the database interaction.

Additional components allow for addition and retrieval of AIF-DF directly from AIF-DB through a RESTful interface, though these are not a core part of the database solution and could easily be replaced by components to handle input and output of other AIF reifications. Access to a live AIF-DB repository, named ArgDB, is available from `http://argdb.computing.dundee.ac.uk`

3.3.1 Summary

A reification of the AIF using SQL.

New Core concepts: none

Language of the reification: SQL

Implementations: ArgDB

Papers: None yet

3.4 AIF-DOT

DOT is a simple graph description language used by variety of tools and systems, particularly those based on the Graphviz package. Several argumentation systems such as AVERS [] and OVAgen[13] can export their output in DOT. A reification of the AIF in DOT is rather trivial because DOT is, semantically, an extremely lightweight language. What is interesting about DOT as a reification is that it provides one of the first examples of true integration between tools. AVERS and OVAgen were developed independently, and more or less in isolation, but both in the context of the original AIF specification. When AVERS exports in DOT format, it does not do so in an arbitrary fashion: there are specific conventions (ellipses for S-nodes, etc.) that only make sense against the backdrop of the AIF. With this common backdrop in place, one can interpret arguments expressed in DOT as semantically much richer than the raw contents of the

[13] `http://ova.computing.dundee.ac.uk/ova-gen/`

DOT expressions on their own would support. This has allowed AVERS exports to be imported both into OVAgen and into ArgDB.

New Core concepts: none

Language of the reification: DOT

Implementations: OVAgen, AVERS

Papers:

- Sensemaking software for crime investigation: how to combine stories and arguments? [4]
- Sensemaking software for crime analysis [5]

4 Conclusions

The AIF core ontology has supported several reifications (in RDF, OWL-DL, as a database schema and as an "interpretation" of DOT). These reifications have been used in several different tools, including Araucaria, ArgDB, ArgDF, AVERS, Avicenna, DMSS, OVA and OVAgen. Initial signs are that the AIF is indeed providing the right balance between semantic coherence across application, and flexibility in expressivity. Increasingly, though, the AIF is being pulled in different directions and extended in different ways, and ensuring coherence between these extensions is the task for the AIF 2.0 meeting.

Part II
Extensions to the Core AIF

5 Introduction

The widespread adoption of the AIF has shed light upon the limitations of the core AIF and has suggested directions in which a revised version might be developed. Two such directions have been developed at Dundee in the context of specific research projects. The first extension is referred to as AIF+ and introduces dialogical extensions to the AIF. The second is sAIF which introduces concepts from abstract argumentation into the AIF.

6 AIF+: Dialogical Extensions to the AIF

AIF+ extends the base AIF language to support descriptions of dialogue transcripts and dialogue protocols. The AIF does not explicitly handle dialogue, and, crucially, cannot connect the execution of dialogue protocols to the creation and manipulation of argument structures. This limitation has been explored in [18] and [27], but these accounts leave important questions unanswered. As has been argued elsewhere [23], the job of the AIF is solely to represent argument, not to perform computation on it. Of course, the representation should have various characteristics that make the sorts of computation we are expecting to want to perform as straightforward as possible. So, for example, both analysis and mark-up of naturally occuring argument and automatic computation of acceptability status using argumentation semantics defined in the style of [10], are computations we are likely to want to perform, and therefore, to a certain extent, drive representational considerations. From a dialogical point of view, we can describe, at an abstract level, an entire class of transformations and computations that we want to perform on structures defined using the AIF^{+}. Following in a long tradition in both the multi-agent community, and before that, the discourse and pragmatics community, dialogical action can be seen as update to a shared information state. In such a model, one needs a representation of the shared state, a representation of how specific dialogical action modifies the state, and, typically, a representation of dialogue *per se*. Then one needs some sort of execution engine that allows dialogues to be conducted (or analysed) to produce both a specific dialogue history and a shared dialogue state. It is usual for the dialogue history to change only monotonically as a dialogue proceeds (that is, once something has been said it cannot be unsaid). In representation of the shared state, monotonicity is less common: perhaps for some specific dialogue protocols, modifications to the information state will happen to be monotonic, but in general, the shared state may lose information as well as gain it – after all, one would not wish to prohibit retraction or withdrawal. The AIF^{+} takes on all three representational roles described above (*viz.* the shared state, the dialogue history and the ways in which dialogue updates the shared state), and, to be precise, treats both the general and instantiated forms of the last role. That is, both the general rules about how dialogue moves could update the shared state, and the specific application of those rules actually updating the shared state in given dialogic situations, are handled by the AIF$^+$. The shared state is represented simply by AIF structures. The dialogue history is represented by Locution (L-) nodes, which are a special type of I-node, and the connections between L-nodes, by Transition Application (TA-) nodes, which are a special type of inference [27]. The general part of describing how dialogue moves update the state is what constitutes (a part of) the dialogue protocol. This is represented in the Forms ontology. In the AIF simpliciter, the Forms ontology describes templates that define how schemes connect premises, conclusions and other schemes of particular types. In the AIF$^+$, the role is exactly the same: the Forms ontology again describes templates that define how schemes connect premises, conclusions and other schemes. In the AIF, the schemes concern inference, conflict and preference. In the AIF$^+$, the schemes concern dialogic transitions (which were the focus of [27]) and illocutionary relations; the former connect components of dialogue, the latter connect components of a dialogue history with specific parts of an AIF structure. The AIF$^+$ thus represents all (but no more) of the information required for a general purpose dialogue execution algorithm to run a particular dialogue protocol, and create both dialogue history and underlying AIF structures. One interesting issue concerns monotonicity of the AIF structures. On the one hand, most dialogues permit some form of retraction or other nonmonotonic change to the shared information space. On the other hand, the AIF and AIF$^+$ must record all that has happened in a dialogue, including the assertions that led up to a retraction (for example). The current solution to this dilemma is to assign commitment (loosely speaking) to particular propositions nonmonotonically, but to leave the accretion of AIF structures as monotonic.

6.1 The meaning of an argument in the AIF+

6.1.1 Two kinds of arguments

According to the WordNet 2.1[14], the noun "argument" has in English five senses (first four from tagged texts):

1. (23) argument, statement – (a fact or assertion offered as evidence that something is true; "it was a strong argument that his hypothesis was true"),
2. (21) controversy, contention, contestation, disputation, disceptation, tilt, argument, arguing – (a contentious speech act; a dispute where there is strong disagreement; "they were involved in a violent argument"),
3. (3) argument, argumentation, debate – (a discussion in which reasons are advanced for and against some proposition or proposal; "the argument over foreign aid goes on and on"),
4. (1) argument, literary argument – (a summary of the subject or plot of a literary work or play or movie; "the editor added the argument to the poem"),

[14] WordNet is a large lexical database of English, developed under the direction of George A. Miller. It was created in 1985 and is being maintained at the Cognitive Science Laboratory of Princeton University.

5. argument – (a variable in a logical or mathematical expression whose value determines the dependent variable; if $f(x) = y$, x is the independent variable).

The first meaning of argument that the AIF$^+$ refers to, is described in the item (1) in the WordNet specification. It relates the word argument to the notion of argumentation understood as reasoning in a logical sense. That is, it is interpreted as one or more reasons for or against a given claim. The second meaning of argument, which is interesting for the AIF$^+$, is described in items (1) and (2) in the WordNet characterization, where it relates argument to the notion of dialogue. That is, it is interpreted as a discussion in which parties state the reason(s) for or against a claim. The meanings in the WordNet's items 4 and 5 are not considered within the AIF$^+$ framework.

The first and second meaning reflect the discussion on the nature of argument between Brockriede [6, 7] and O'Keefe [19]. The result of this discussion was the distinction between argument$_1$ and argument$_2$:

> On the one hand it [the word "argument"] refers to a kind of utterance or a sort of communicative act. This sense of the term I will call "argument$_1$". It is the sense contained in sentences such as "he made an argument". On the other hand, "argument" sometimes refers to a particular kind of interaction. This sense, "argument$_2$", appears in sentences such as "they had an argument" [19, 121].

These two kinds of arguments have different properties, e.g.: the arguments$_1$ can be refuted, invalid or fallacious, while the arguments$_2$ can be pointless or unproductive [19, 121].

Hitchcock [12] shows that the ambiguity of the word "argument" is not present in other languages:

> In classical Greek, for example, the reason-giving sense is expressed by the word *logos* (e.g. in Plato's *Phaedo*, at 90b-91c) in one of its many senses, whereas the disputational sense is expressed by the word *amphisbêtêsis* or *antilogia*, "dispute" or "controversy". In Latin, the reason-giving sense is expressed by the word *argumentum*, "proof" or "evidence", the disputational sense by the word *disputatio*, "debate" or "dispute". In French, as Plantin (2003: 174) points out in detail, the reason-giving sense is expressed by the verb *argumenter* ("to argue [that]") and its cognates, the disputational sense by the verb *discuter* ("to discuss", in an aggressive way). In Spanish (Claudio Duran, personal communication), the reason-giving sense is expressed by the word *argument*, and the disputational sense by the words *discusión* (discussion) or *controversia* (controversy) or *disputa* (dispute). In Russian, the reason-giving sense is expressed by the word *dovod* (supporting reason), the disputational sense by the word *spor* or *ssora*. In German, the reason-giving sense is expressed by the word *Argument*, the disputational sense by the word *Disput* [12, 102].

The last important clarification, which we want to make here, is related to the popular distinction between argument-as-product and argument-as-process. For example, Brockriede identifies the argument$_1$ with product and the argument$_2$ with process [6].

However, it can be argued that both kinds of arguments may take the form of process or product. We may consider an argument (either argumentation or dialogue) as a process that actually "happens" and is created in interaction among agents. On the other hand, we may focus on the result (product) of this interpersonal event. The AIF$^+$ aims to describe the process of argument with the use of its product, since only static objects can be specified and stored in an ontology.

6.1.2 Motivation

Analyzing argumentation in the context of dialogue provides insight into its important properties which are not expressible in a model of monologic argumentation. Firstly, in real-life practice argumentation is commonly related to and therefore dependent on a dialogue: "Most often, argument occurs in dialogue. (...) to understand an argument, it is very often highly important to know something about the context of dialogue in which the argument has occurred" [30, 411-2]. Moreover, the context of a dialogue is especially important, when we want to evaluate an argumentation: "In order to evaluate an argument as correct or incorrect, it is vital to know the context of conversation [i.e. dialogue] in which this argument was used" [31, 98]. Walton describes the case of a television program "Infomercial" as an example of how a dialectical context influences the evaluation of an argument [31, 120-3]. Infomercial was transmitted in TV as a half-hour talk-show, while in fact it was a commercial. The authors of this program expected to make use of the dialectical shift, i.e. that arguments presented during Infomercial will be evaluated by its viewers in the context of one kind of dialogue, while in fact the other context should be considered. In that case, the program was suggested to be a talk show which assumes the (comparative) objectivity of presented information, while the real context was a sales pitch which presumes one-sided promotion to make a viewer buy some product.

Generally, the lack of distinction between argument$_1$ and argument$_2$ may lead to a confusion in argument's specification and investigation. An example of such a confusion was the framework proposed by Brockriede [6]. O'Keefe showed that six characteristics of an argument identified by Brockriede define not one phenomenon, but two different kinds of phenomena: "a confusion of the two senses of argument will obviously turn on a recognition of the differences between argument$_1$ and the argument$_2$. But Brockriede's elision of the two senses of "argument" is important, because it is indicative of shifting concerns in the study of argument" [19, 126]. The ontology which refers to both meanings of "argument" is therefore capable to provide rich language adequate for complex models of argument assumed in different visualization tools and methods.

6.1.3 Pragmatic account

The AIF$^+$ incorporates a view of argument that is rooted in pragmatics. The central notion of pragmatics is the notion of utterance:

> Pragmatics deals with *utterances*, by which we will mean specific events, the intentional acts of speakers at times and places, typically involving

The level of abstraction	An example
Argument scheme	α, $\alpha \rightarrow \beta$, *therefore* β.
Argument type	Harry was in Dundee, If someone is in Dundee, then he is in Scotland, *therefore* Harry was in Scotland.
Argument token	Harry was in Dundee, If someone is in Dundee, then he is in Scotland, *therefore* Harry was in Scotland, *in the context of* a dialogue between Bob and Wilma.

Figure 4: Different levels of abstraction in argument analysis.

> language. Logic and semantics traditionally deal with properties of *types* of expressions, and not with properties that differ from token to token, or use to use, or, as we shall say, from utterance to utterance, and vary with the particular properties that differentiate them. Pragmatics is sometimes characterized as dealing with the effects of *context*. This is equivalent to saying it deals with utterances, if one collectively refers to all the facts that can vary from utterance to utterance as 'context' [14].

In the pragmatic account, an argument would not be a type, but a token. This assumes different levels of abstraction, at which we can look for an argument (see Table 4). At the most abstract level, there are logical schemes such as e.g. Modus Ponens: α, $\alpha \rightarrow \beta$, therefore β, or the scheme from witness testimony (see e.g. [3]): x asserts α, x is in a position to know whether α is true or not, therefore α. However, the schemes are not arguments$_1$. An argument is an instantiation of a given scheme. Yet an instantiation may be understood in two manners: as a type or a token. The logical accounts treat argumentation as an argument type. That is, an instantiation of Modus Ponens such as: *Harry was in Dundee, If someone is in Dundee, then he is in Scotland, therefore Harry was in Scotland*, is a reasoning or an argument, which properties (such as soundness, validity, etc.) may be analyzed by logical tools. However, it is still not an argument$_1$ from the pragmatic point of view. The argument is the instantiation of a scheme *used in a given context*, e.g. in a context of a dialogue between Bob and Wilma. That is, an argument$_1$ is a token of the expression "Harry was in Dundee and if someone is in Dundee, then he is in Scotland, therefore Harry was in Scotland" used (uttered) by Bob, e.g., to convince Wilma that Harry was in Scotland. Similarly, a dialogue is not a type, but a token. Both kinds of arguments are real objects that are possessed by some agents (arguers): they are performed by someone (author, sender, proponent, etc.) and addressed to someone (hearer, receiver, opponent, audience, etc.). As Brockriede states: "Arguments are not in statements but in people" [7, 179].

In the AIF$^+$, both kinds of arguments (i.e. tokens of argument$_1$ and tokens of argument$_2$) are located in the AIF$^+$ object layer. Above that layer (i.e. Forms ontology), there are schemes of arguments. Argument-types may be "extracted" from the

AIF$^+$ by means of "abstracting" from a token of argument. That is, if we want to explore a particular argument-type, then any argument-token corresponding to this type can be chosen and considered in isolation from its context (see e.g. [12, 107-8] for the discussion on the relation between an argument token and type).

According to the pragmatic theory of speech act (see e.g. Austin [2], Searle [28], Searle & Vanderveken [29]), argumentation and dialogue are speech acts. A speech act $F(p)$, such as claim$p$, why$p$, consists of an illocutionary force F and a propositional content p (Searle & Vanderveken [29]). An illocutionary force is an intention of uttering a propositional content. That is, the performer of a speech act may utter p with an intention of asserting, asking, promising and so on. In general, speech acts are characterized by the locutionary, illocutionary and perlocutionary aspects. Locution is an act of performing an *utterance*. In a dialogue between Bob and Wilma, Bob may, e.g., utter that Harry was in Dundee. Illocution is an act of performing the utterance with some *illocutionary force*. For instance, Bob may utter "Harry was in Dundee" with an intention of informing an audience about Harry's visit in Dundee or with an intention of explaining e.g. why Harry wasn't in London at that time. Finally, perlocution is an act of causing an *effect* by performing the utterance. For example, Bob's utterance "Harry was in Dundee" may make Wilma regret that they didn't meet during Harry's visit in Dundee or make her question Bob's belief about Harry's visit in Dundee.

A speech act can be felicitous or infelicitous depending on whether or not it successfully performs a given action. For example, my act of promise that I met you yesterday is infelicitous. The rules that determines what constitutes a successful speech act are called the *constitutive rules*. In [28], Searle distinguishes four classes of those rules:

- propositional content rules: some illocutions can only be achieved with an appropriate propositional content, e.g. a promise may refer only to what is in the future and under the control of a speaker,
- preparatory rules: they determine what a speaker presupposes in performing a speech act, e.g. a speaker cannot marry a couple unless he is legally authorized to do so,
- sincerity rules: they tell what psychological state is expressed (e.g. an assertion expresses belief, a promise expresses an intention to do something) and a speech act is sincere only if a speaker is actually in this state,
- essential rules: they determine what a speech act consists in essentially, e.g. a promise commits a speaker to perform act expressed in a propositional content.

Thus, my promise that I met you yesterday was infelicitous, since I did not fulfil the propositional content condition (the propositional content does not refer to a future action).

The essential conditions are then used to build a taxonomy of speech acts. Searle distinguishes five classes of speech acts:

- assertives describe the speaker's belief and his intention that the hearer forms a similar one, they also commit the speaker to the truth of the propositional content,

- directives express the speaker's attitude about a future act performed by the hearer and the intention that the utterance be taken as reason for the action,
- commissives describe the speaker's intention to do something and the belief that his utterance obliges him to do it, they commit the speaker to do something,
- expressives express feelings toward the hearer, and
- declaratives express that the speaker performs a given action.

Depending on the further characteristics of an illocutionary force, each class divides into various subclasses. For example, assertives split into claim$p$, deny$p$, guess$p$, argue$p$, rebut$p$, etc.

The illocutionary acts can be divided into two categories with respect to the number of propositional contents that the illocutionary force refers to. First, the illocution may be an instance of a property of a content, such as in the following speech acts: claimα, whyα or concedeα. It may also be an instance of a relation between contents (see SDRT: Segmented Discourse Representation Theory [1]). Argumentation can be viewed as an example of the second category of the illocutionary act described by the following type: argue(α, β). The speech acts may be also divided into simple acts and compound acts containing (constituted from) distinct kinds of acts. The dialogue is an example of the second category: it may be built from consecutive speech acts. For instance, a dialogue may be a sequence $\langle$claimα, whyα, argue(α, β), concede$\alpha\rangle$, i.e. the first move in the dialogue is an act of claiming that α holds, what is followed by questioning this claim, what is replied by an argumentation supporting α with β, and the final move is conceding the initial claim.

6.1.4 Interrelation between argument$_1$ and argument$_2$

In real-life scenarios, both kinds of argument coexist and interact with each other. Imagine a following dialogue between Bob and Wilma (i.e. an argument$_2$):

Bob: You know what? Harry was in Dundee.

Wilma: How do you know?

Bob: I saw him.

In this dialogue, Bob and Wilma jointly build an argument$_1$: Harry was in Dundee, since Bob saw him in Dundee. Observe that Bob's utterance "Harry was in Dundee" could be left without justification (argument), if Wilma did not question it, i.e. if she did not express doubts whether it is true or false (or whether to accept it or not). In other words, the dialogue triggers the argumentation. Moreover, the context of the argument$_2$ enables to keep track of the agents' interaction which creates the argument$_1$: the argumentation is invoked (the broken-line arrow in Fig. 5) by Wilma's speech act, and provided (the solid-line arrow in Fig. 5) by Bob's speech acts.

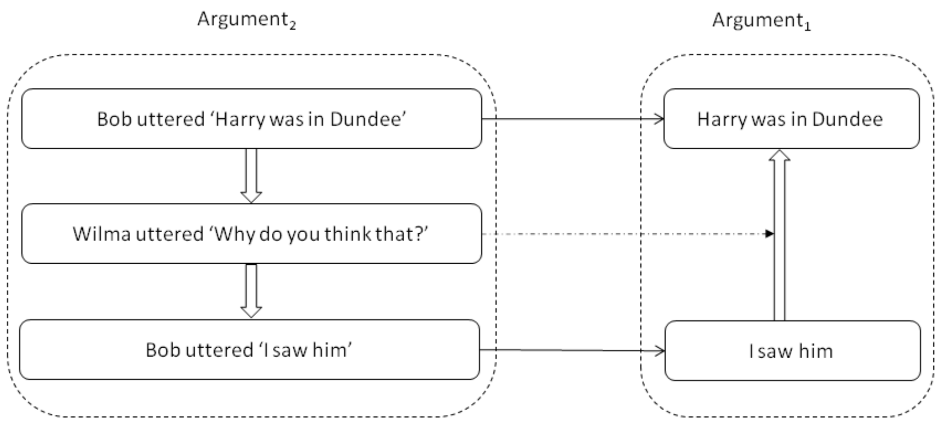

Figure 5: Interrelation between argument$_1$ and argument$_2$.

6.1.5 The general idea of the AIF$^+$ specification

The background framework for the argument$_1$ is some kind of logic (see Table 6). However, its meaning is adjusted to the pragmatic approach as described in Section 6.1.3. Consequently, the argument$_1$ is interpreted as an instance (token) of a reasoning. Its basic types of units are built from propositions which may describe any situation. They can refer to someone's speech act (e.g. Bob's assertion that Harry was in Dundee) as well as to any other action or situation (e.g. Harry's presence in Dundee). The main types of relations between those units can be based on deductive rules (e.g. on *Modus Ponens*: Harry was in Dundee and If someone is in Dundee, then he is in Scotland, therefore Harry was in Scotland), on defeasible rules such as argumentation schemes (e.g. on *appeal to witness testimony*: Bob asserts that Harry was in Dundee and Bob is in a position to know whether Harry was in Dundee, therefore Harry was in Dundee), or on rules for conflicts (e.g. a *logical conflict* among a proposition and its negation: Harry was in Dundee and Harry wasn't in Dundee).

The argument$_2$ has a dialectical interpretation (see Table 6). Its basic types of units are built from propositions restricted to describe speech acts. The types of relations between those units are based on dialogue rules (e.g. on a protocol for the system PPD; for instance, the protocol demands that a move claimα can be followed either by whyα or by concedeα (see e.g. [20])).

The interesting question is how exactly the units of argument$_2$ are connected to the units of argument$_1$. So for example, what is the relationship between a proposition p (e.g. "Harry was in Dundee") and the proposition "x asserted p" (e.g. "Bob asserted that Harry was in Dundee")? The link between a locution (or, more precisely, a proposition that reports a locution) and its propositional content (i.e. the proposition or propositions to which the locution refers) is determined and authorized by constitutive rules for performing speech acts (see Section 6.1.3). The constitutive rules can be of a number of types depending on the type of illocutionary force which the performer

	Background	Types of Units	Main Relations
Argument$_1$	logic	propositions describing the world (in particular locutions)	- deductive rules e.g. Modus Ponens - argumentation schemes e.g. appeal to witness testimony - conflict schemes e.g. logical contradiction
Argument$_2$	dialectics	propositions describing locutions	- dialogue rules e.g. protocols for PPD$_0$
Interaction between arg$_1$ and arg$_2$	pragmatics	elements of arg$_1$ and arg$_2$	- illocutionary schemes

Figure 6: The properties of arguments considered in the AIF$^+$.

of the speech act assumes. In this way, it bears more than a passing resemblance to scheme structure. These schemes are not capturing the passage of a specific inferential relation, but rather than passage of a specific linguistic relation dependent on the type of illocutionary force used in a speech act. As a result, we refer to these schemes as illocutionary schemes (see Table 6).

In the AIF$^+$, argument$_1$ is represented by two kinds of nodes: (1) *information nodes* (denoted by I$_i$ in Fig. 7), which refer to data, and (2) scheme nodes, which refer to the passage between information nodes. The scheme nodes are classified into three groups: (2.1) *rule application nodes* which correspond to inference or support (denoted by RA$_i$ in Fig. 7), (2.2) conflict application nodes which correspond to conflict or refutation, and (2.3) preference application nodes which correspond to value judgments or preference orderings. In this section, we focus on the I- and RA-nodes. Intuitively, the I-nodes are the units of argument$_1$, and the RA-nodes are the elements of relations on a set of the units. The relations are governed by deductive or defeasible rules of inference. In other words, the RA-nodes links I-nodes. Thus, a particular RA-node can be described as a tuple:

$$\mathrm{RA}_i = (\mathrm{I}_1, ..., \mathrm{I}_k), \text{ where } \mathrm{I}_1, ..., \mathrm{I}_k \text{ are certain information nodes.}$$

The argument$_2$ is also represented by two types of nodes: (1) *locution nodes* (denoted by L$_i$ in Fig. 7), which refer to utterances and constitute a subclass of information nodes: $\mathcal{L} \subseteq \mathcal{I}$ (where $\mathcal{L}$ is a set of all locution nodes and $\mathcal{I}$ is a set of information nodes), and (2) *transition application* nodes (denoted by TA$_i$ in Fig. 7), which refer to the passage between locutions and constitute a subclass of rule application nodes: $\mathcal{TA} \subseteq \mathcal{RA}$ (where $\mathcal{TA}$ is a set of transition application nodes and $\mathcal{RA}$ is a set of rule application nodes). Intuitively, the L-nodes are the units of argument$_2$, and the TA-nodes are the elements of relations on a set of the units. The relations are governed by protocols of dialogues. A particular TA-node can be described as a tuple:

$$\mathrm{TA}_i = (\mathrm{L}_1, ..., \mathrm{L}_k), \text{ where } \mathrm{L}_1, ..., \mathrm{L}_k \text{ are certain locution nodes.}$$

The interaction between argument$_1$ and argument$_2$ is captured by means of *illocutionary application nodes* (denoted by YA$_i$ in Fig. 7). There are two kinds of YA-nodes: (1) the YA-nodes between I-Nodes in argument$_1$ and argument$_2$ (denoted by u-YA$_i$ in Fig. 7), and (2) the YA-nodes between S-Nodes in argument$_1$ and argument$_2$ (denoted by r-YA$_i$ in Fig. 7). A set of illocutionary application nodes YA can be described as follows:

$\mathcal{YA} = u\text{-}\mathcal{YA} \cup r\text{-}\mathcal{YA}$, where elements of the sets $u\text{-}\mathcal{YA}$ and $r\text{-}\mathcal{YA}$ are pairs such that

- u-YA$_i$ = (L$_j$, I$_k$) where L$_j$ is a locution node, and I$_k$ is an information node,
- r-YA$_i$ = (TA$_j$, RA$_k$) where TA$_j$ is a transition application node, and RA$_k$ is a rule application node.

As noted above, the link between a speech act and its propositional content is determined and warranted (authorized) by the constitutive rules. For instance, the assertion "Bob asserted that Harry was in Dundee" is related to the proposition "Harry was in Dundee" (see u-YA$_1$ in Fig. 7), if the constitutive rules for assertives are satisfied. In natural contexts, the most important types of rules are the preparatory and sincerity rules. This is related to the distinction between a successful and nondefective speech act [29]. That is, an assertion may be successful but still defective, if its performer did not have enough evidence for the statement or he declared what he actually disbelieves. Thus, the passage between "Bob asserted that Harry was in Dundee" and "Harry was in Dundee" could be blocked, if Bob was insincere. Unlike u-YA$_1$ (see Fig. 7), u-YA$_2$ refers to directive illocutionary force. Therefore, those passages are regulated and warranted by different constitutive rules.

Now, we can describe the example from Section 6.1.4 within the AIF$^+$ framework. The dialogue between Bob and Wilma consists of three locutions: L$_1$ - Bob's utterance 'Harry was in Dundee', L$_2$ - Wilma's utterance 'Why do you think that?', and L$_3$ - Bob's utterance 'I saw him' (see Fig. 7). They are related to each other by means of transitions TA governed by dialogue rules: TA$_1$ is an instance of a rule determining that an assertion can be followed by a challenge, and TA$_2$ is an instance of a rule determining that a challenge can be followed by an argument.

The argumentation, which Bob and Wilma jointly build, consists of two units: I$_1$ - Harry was in Dundee, I$_2$ - Bob saw Harry in Dundee. They are related to each other by means of RA governed by an argumentation scheme, i.e. RA$_1$ is an instance of the scheme from perception: having a percept with content α is a reason to believe that α.

The interaction between the dialogue and the argumentation is described with the use of the YA-nodes. Three of them are illocutions between units. The speech acts L$_1$ and L$_3$ have the assertive illocutionary force and the propositional contents I$_1$ and I$_2$, respectively. The passage between L$_1$ (resp. L$_3$) and I$_1$ (resp. I$_2$) is represented by YA$_1$ (resp. YA$_3$). The node L$_2$ describes the directive speech act. Its representation is troublesome. Its propositional content is denoted by I$_1$. However, the link captured by the YA-nodes means more than simply a "has content" relation. Intuitively, the "has content" relation would be an operation of extracting a propositional content out of a speech act independently of the illocutionary force performed in this speech act.

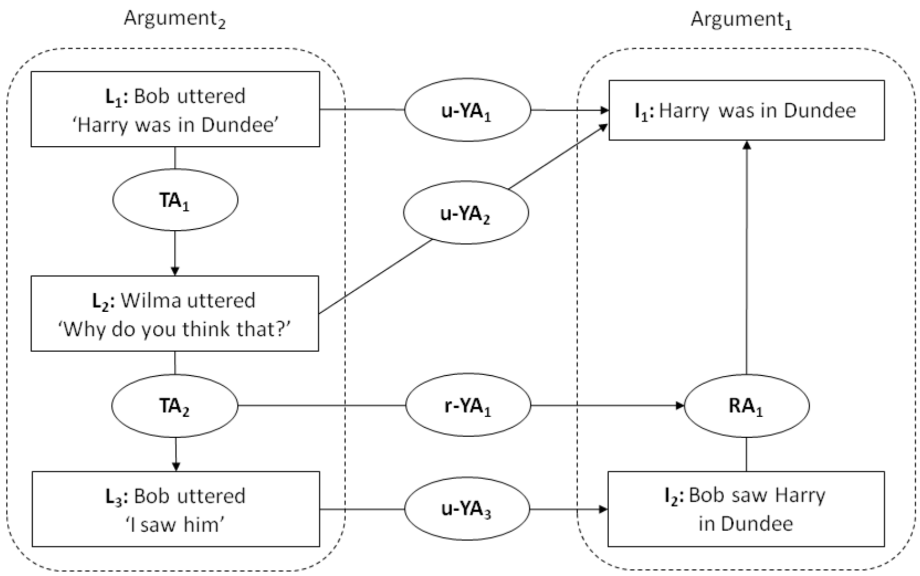

Figure 7: Interrelation between argument$_1$ and argument$_2$.

Then, YA_1 and YA_2 would be the instances of the same YA-operation. However, such specification of the YA-nodes is not satisfactory.

There is also one illocution of a complex type: r-YA_1 links Bob's response to Wilma's challenge (i.e. TA_2) with the argument "Bob saw Harry in Dundee" for the claim "Harry was in Dundee" (i.e. RA_1). Intuitively, r-YA_1 corresponds to the observation that Bob's argument is invoked by Wilma's challenge (see Section 6.1.4).

6.2 Recap: AIF+ Ontological Extensions

A distinguishing feature of dialogue is the interplay between the locutors, as each locutor responds in turn to the utterances made by other participants of the dialogue. This aspect of dialogue is characterised as a form of inference called *transitional inference* and is represented by TA-Nodes, Transition Application nodes, which capture the flow of a dialogue, for example, recording that a given assertion has been made in response to an earlier question.

According to the original specification of the AIF, direct I-node to I-node links are prohibited (and with good reason: to do so would introduce the necessity for edge typing obviating this requirement is one of the advantages of the AIF approach). The answer to this question is already available in the work of Searle [28] and later with Vanderveken [29].The link between a locution (or, more precisely, a proposition that reports a locution) and its propositional content (i.e. the proposition or propositions to which the locution refers) is determined and authorized by *constitutive rules* for performing speech acts. These rules describe in what (successfully) performing a speech act consists.

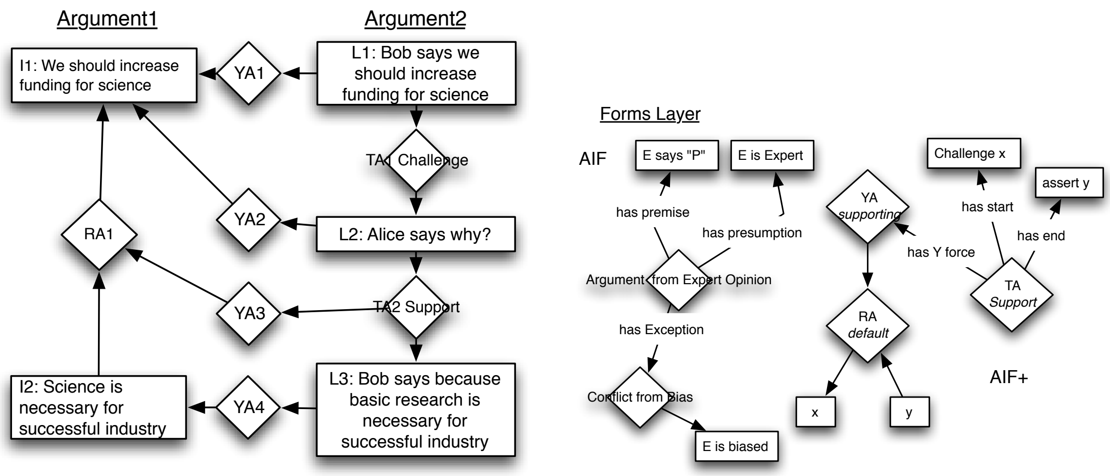

Figure 8: Illustrating, on the left, the link between arguments (argument$_1$) and dialogue (argument$_2$) at the object layer, and on the right, examples of corresponding links between TA-, YA-, and RA-nodes at the AIF$^+$ form layer and their similarity to exception links between Schemes in the AIF *simpliciter*

The constitutive rules can be of a number of types depending on the type of *illocutionary force* which the performer of the speech act assumes. In this way, they bear more than a passing resemblance to scheme structure. These schemes are not capturing the passage of a specific inferential relation, but rather than passage of a specific linguistic relation dependent on the type of illocutionary force used in a speech act. As a result, we refer to these schemes as *illocutionary schemes* or Y schemes. Specific applications of these schemes are then, following the now familiar pattern, YA schemes. YA-nodes describe passage between L-nodes ("elements" of argument$_2$) and I-nodes ("elements" of argument$_1$ or, more rarely, argument$_2$).

In addition to delineating the relationship between locutions uttered and the underlying structure of arguments expressed during a dialogue, AIF$^+$ also supports representation of dialogue protocol. Protocols are described using Locution Description (LDesc-Nodes) nodes. For each locution, represented by an L-Node, there is a corresponding LDesc-Node which can in turn be linked to corresponding PreCondDesc and PostCondDesc nodes that describe, respectively, the pre-conditions and post-conditions for legally uttering a locution. Pres and post-conditions can be represented in a number of ways and rather than create a new protocol specification language to account for this, AIF$^+$ supports specification of pre- and post-conditions using fragments of appropriate dialogue protocol description languages such as the Dialogue Game Description Language (DGDL) [33]. For example, the post-condition associated with the challenge move of ASD, that the content "p" of the move should be added to the listener's commitment store, can be expressed using a fragment of DGDL as follows: *store(add,* $\{p\}$*, CS, listener)*. The development of a dialogue, as an ordered sequence of locutions, is handled by transitional inference schemes that describe, for a given locution, the available responding locutions.

The revised upper ontology of the AIF+ is thus as in Figure 9:

6.3 Calculated Properties

The AIF cannot handle arguments based on the results of computation over AIF structures. Arguments based on counting other arguments, weighing other arguments, comparing other arguments and evaluating other arguments all involve processes (counting, weighing, comparing, evaluating) that cannot be captured in the AIF itself (and nor should they be, for otherwise the AIF would swell to some general purpose programming language). These various processes might collectively be thought of as ways of calculating properties about the arguments that the AIF represents. It is not that such arguments cannot be represented at all. But rather, if arguments are based on these *calculated properties* – arguments such as "the prosecution has not provided sufficient evidence for a conviction, so the accused is released" – then they can only be represented in the same way as normal propositions. The AIF has no way of capturing the link between such a statement and, say, the existence or non-existence of a set of other nodes. For monologic arguments this is a relatively small problem, but excludes, as the previous example demonstrates, some relatively common forms of legal argument. It may be that most or all of these cases can be handled with preference applications. But for dialogue, the matter is more serious. Protocol rules are very often defined on the basis of calculated properties of dialogue histories: the existence or non-existence of particular claims, the current status of claims, and most significantly, commitments.

Our current approach to tackling this problem is pragmatic. Arguments based upon calculated properties do occur, so we must represent them. To be more accurate, such arguments do occur naturally, and therefore the AIF *will* be employed by its users to represent them. The connection between these propositions and the AIF structures from which their truth values can be determined are no more represented in the AIF than are the connections between other, more conventional, propositions and the real world phenomena that establish their truth values (if such phenomena exist). It is not the job of the AIF to represent or calculate some objective truth (or even multiple subjective truthes). It is possible, in contrast, to imagine a process that runs over a given fragment of AIF, and can determine whether or not certain calculated properties hold. There might be any number of such processes – one to calculate what counts as sufficient evidence for a prosecution case in a given court system, another to determine which arguments are justified according to a given acceptability semantics [10], etc. The results of these computational processes may then be compared either automatically or manually with claims about calculated properties that are represented explicitly in the AIF. But the job of ensuring that all and only true calculated properties are represented is not and cannot be the responsibility of the AIF alone.

With calculated properties representable as conventional I-nodes, their frequency in dialogue protocols becomes slightly less of a problem. So, for example, requiring that an opponent is committed to a proposition p before challenging p might require the fact that 'opponent is committed to p' is represented explicitly as an I-node. Commitment is, from an argumentation theory point of view, the most important calculated property,

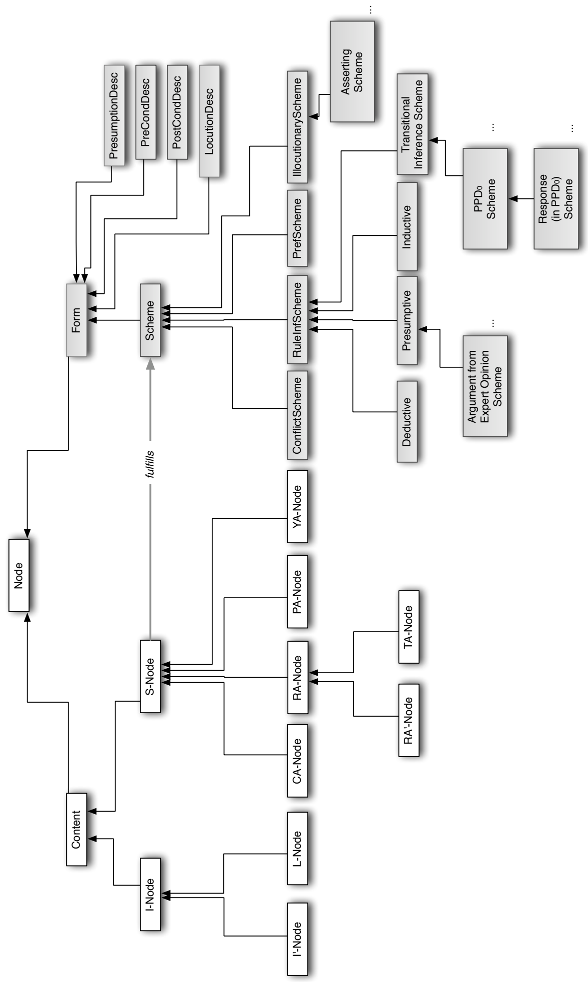

Figure 9: Upper levels of AIF extended to handle dialogue

as it forms a core part of many dialogue games. Statements about (that is, propositions which refer to) commitment are I-nodes like any other. This is reminiscent of Locution nodes, or L-nodes, which are also a special type of I-node. Though L-nodes have a particular role to play in their most common application in representing dialogue, they can also be used as propositions simpliciter: the fact that Bob said q might be used as a premise in an argument for Bob's sentience, for example. So too I-nodes that correspond to statements about commitment can be used in contexts beyond the dialogue game at hand. A very common example of such commitment-based arguments is in circumstantial *ad hominem* arguments, where a speaker's commitments (or in some cases, the locutions themselves) are used to form an argument against the speaker's current claim. As with the rest of the AIF, claims about commitment (that a particular individual is committed to a particular claim, possibly in a particular context) occur just once in the AIF structure, and can be connected into multiple arguments. These commitment-oriented propostions seem entirely reasonable to add in to the AIF structure as a dialogue proceeds. Unfortunately, there are many other types of calculated property that one can imagine being used by a dialogue protocol that are much less appealing. One would not want a proliferation of nodes such as 'the opponent has not spoken in seven turns,' or 'the last-but-two move was not an assertion,' despite the fact that such propositions may very well be required for application of a transition scheme (that is, for the application of a structural rule in a protocol). The solution to this problem is actually already built in to the AIF. As described in [25], scheme nodes (there, rule of inference schemes in particular) have within them definitions of both implicit premises and exceptions. They are defined by the critical questions associated with a scheme and provide stereotypical growth points that allow an argument to be extended through challenge or questioning. Implicit premises, in particular are important to us here. In rule of inference schemes, implicit premises are often included in the scheme but rarely questioned. If they are not made explicit, then they do not occur in the AIF graph – they simply remain an un-exploited potential growth point in the scheme. The same thing happens for the calculated properties that are so often used in protocol definitions. If the fact that 'the opponent has not spoken in seven turns' is not disputed (by either the original participants in the dialogue or by any subsequent analyst or contributor) then it remains an implicit part of the transition scheme, and never becomes an explicit part of the AIF graph. In this way, the AIF provides a mechanism for representing these calculated properties when necessary, but does not clutter analyses with enormous numbers of implicit premises.

It is not difficult to imagine dialogic computational processes, akin to the monologic processes above, that perform various tasks on an AIF structure: verifying that a given AIF fragment conforms to the rules of a protocol; listing alternative ways in which a given interlocutor could legally respond at a given point in a dialogue, etc. But again, these computational process are run on top of AIF; they are not a part of it. Even the process of having a dialogue itself is separate from the AIF. Such a process (which could well involve an interface which allows a set of users to contribute to a dialogue according to the rules laid out in the protocol) would be responsible for updating AIF structure, most certainly, but at each stage in the process that static AIF structure records the history of the dialogue to that point, with the connections between the locution nodes and the I-nodes to which they refer.

In both the monological and dialogical settings, this approach to handling calculated properties is consistent with Finocchiaro's (monological) work on meta-arguments and Krabbe's earlier work on metadialogue [15], in that the meta-moves are explicitly representable. What the AIF does not do however (or at least not in its current incarnation) is develop an infinitely extensible language of meta- and meta-meta-arguments as has been advocated elsewhere in the AI literature [35], as there is not yet any convincing need in realistic scenarios. This is certainly an avenue for further research, however.

Finally, an interesting difference between inference schemes in monological argument structures and transitions schemes in dialogical argument structures is that the former will rarely constrain their premises to be specific types of calculated properties. There will be few rules of inference, for example, that are based upon whether one argument defeats another. In contrast, transition schemes will very often be based upon calculated properties such as commitments. There are exceptions, however. We have seen how some legal argumentation makes stereotypical use of calculated properties; such usage may very well be suited to being captured in argumentation schemes. And similarly, transition schemes will sometimes make use of purely structural (i.e. non-calculated) properties – 'always assert after a challenge,' for example. When a dialogue protocol is made up of transition schemes that are exclusively of this type, the protocol is Markovian[15] (in the general case, protocols, and especially those expressed as dialogue games, are not intrinsically Markovian).

Papers: AIF+: Dialogue in the AIF, Reed et al., COMMA 2008

7 sAIF: Abstract Extensions to the AIF

Scruffy AIF or sAIF (pronounced 'safe', to remind us that it is not at all safe) is a version of the core AIF with the addition of explicit representation of abstract arguments. We introduce A nodes which are linked to I and S nodes via comprises links. So an argument p, scheme99, q might have an edge from each of its three component nodes, p, scheme99 and q to an A node a1. Argument a1 thus comprises the argument p, scheme99, q. A nodes are handy, particularly when we want to perform analysis (we can identify the original argument as a whole, as it appeared in the newspaper or whatever), and when we care about intertextuality (whereby one arguer incorporates or refers to the argument of another. On the other hand, it is not clear exactly what A nodes are: it is arguable they should be seen as a type of proposition [13], and therefore of I-node, but as this would require I-node to I-node links, it may be better to view A nodes as sui generis.

Once one has an explicit representation of arguments, it seems a small step to represent abstract argumentation frameworks, and thereby allow the AIF to benefit directly from the many computational algorithms and results that have been developed with them. To do this, we also need to represent Attacks. The sAIF introduces K nodes which connect A nodes in the way you would expect. K nodes were introduced simply as a quick way of allowing OVAgen to interact with other AIF-based tools. The

[15] i.e. the legitimate future moves depend only upon the current move and not upon other features of the dialogue history.

problem is that K nodes are almost certainly a (special) type of calculated property, i.e. one abstract argument attacks another precisely in virtue of a relationship or set of relationships holding between their parts. If attack relations are indeed calculated properties, then ceteris paribus they should not be represented explicitly, for otherwise, AIF graphs could be intrinsically inconsistent.

As one would expect, sAIF currently allows representation of abstract argument frameworks without having to also represent the structured arguments from which they are composed.

8 Conclusions

This report has surveyed the current state of the AIF core ontology, the range of ways in which the core AIF has been reified and deployed in software applications, and has introduced a number of concrete proposals for ways in which an AIF 2.0 could be developed.

References

[1] N. Asher and A. Lascaride. *Logics of Conversation*. Cambridge University Press, 2003.

[2] J. L Austin. *How To Do Things With Words*. Oxford University Press, 1962.

[3] F. J. Bex, H. Prakken, C. Reed, and D. N. Walton. Towards a formal account of reasoning about evidence: argumentation schemes and generalisations. *Artificial Intelligence and Law*, 11:125–165, 2003.

[4] F.J. Bex, S.W. van den Braak, H. van Oostendorp, H. Prakken, B. Verheij, and G. Vreeswijk. Sensemaking software for crime investigation: how to combine stories and arguments? *Law, Probability and Risk*, 6:145–168, 2007.

[5] S.W. van den Braak. *Sensemaking software for crime analysis*. PhD thesis, Utrecht University, 2009.

[6] W. Brockriede. Characteristics of arguments and arguing. *Journal of the American Forensic Association*, 13, 1977.

[7] W. Brockriede. Where is argument? *Journal of the American Forensic Association*, 11:179–182, 1977.

[8] S. Buckingham Shum. Cohere: Towardsweb 2.0 argumentation. In *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA)*, pages 97–108. IOS Press, 2008.

[9] C. Chesnevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, and S. Willmott. Towards an argument interchange format. *Knowledge Engineering Review*, 21(4):293–316, 2006.

[10] P. M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning and logic programming and *n*-person games. *Artificial Intelligence*, 77:321–357, 1995.

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

[12] D. Hitchcock. Informal logic and the concept of argument. In D. M Gabbay, P. Thagard, and J. Woods, editors, *Handbook of the Philosophy of Science*, volume 5 of *American Philosophical Quarterly Monograph Series*, pages 101–129. Elsevier, 2006.

[13] J. Katzav and C. Reed. On argumentation schemes and the natural classification of arguments. *Argumentation*, 18(2):239–2259, 2004.

[14] K. Korta and J. Perry. Pragmatics. In Edward N. Zalta, editor, *Stanford Encyclopedia of Philosophy*. Center for the Study of Language and Information, Stanford University, 2006.

[15] E. C. W. Krabbe. Metadialogues. In *Proceedings of the Fifth International Conference of the International Society for the Study of Argumentation*, pages 641–644. International Centre for the Study of Argumentation, 2003.

[16] A. Letia, I.A. amd Groza. Contextual extension with concept maps in the argument interchange format. In I. Rahwan and P. Moraitis, editors, *Proceedings of ArgMas 2008*, 2008.

[17] H. Lindgren. Towards using argumentation schemes and critical questions for supporting diagnostic reasoning in the dementia domain. In *Proceedings of CMNA 2009*, 2009.

[18] S. Modgil and J. McGinnis. Towards characterising argumentation based dialogue in the argument interchange format. In *Proceedings of the 4th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS2007)*, page to appear, 2007.

[19] D. J. O'Keefe. Two concepts of argument. *The Journal of the American Forensic Association*, 13(3):121–128, 1977.

[20] H. Prakken. On dialogue systems with speech acts, arguments, and counterarguments. In *JELIA'2000, The 7th European Workshop on Logics in Artificial Intelligence*, number 1919 in Springer Lecture Notes in AI, pages 224–238. Springer Verlag, Berlin, 2000.

[21] I. Rahwan and B. Banihashemi. Arguments in owl: A progress report. In *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA)*, 2008.

[22] I. Rahwan, B. Banihashemi, C. Reed, D. Walton, and S. Abdallah. Representing and classifying arguments on the semantic web. *The Knowledge Engineering Review.*, to appear, 2010.

[23] I. Rahwan and C. Reed. *Argumentation in Artificial Intelligence*, chapter The Argument Interchange Format. Springer, 2009.

[24] I. Rahwan and P. V. Sakeer. Towards representing and querying arguments on the semantic web. In *Proceedings of the 2006 International Conference on Computational Models of Argument (COMMA)*, 2006.

[25] I. Rahwan, F. Zablith, and C. Reed. Laying the foundations for a world wide argument web. *Artificial Intelligence*, 171:897–921, 2007.

[26] C. Reed and G. Rowe. Araucaria: Software for puzzles in argument diagramming and xml. Technical report, University Of Dundee, 2001.

[27] C. Reed, S. Wells, G. W. A. Rowe, and J. Devereux. Aif+: Dialogue in the argument interchange format. In *2nd International Conference on Computational Models of Argument (COMMA 2008)*, 2008.

[28] J. R. Searle. *Speech Acts*. Cambridge University Press, 1969.

[29] J. R. Searle and D. Vanderveken. *Foundations of Illocutionary Logic*. Cambridge University Press, 1985.

[30] D. Walton. What is reasoning? what is an argument? *The Journal of Philosophy*, 87(8):399–419, 1990.

[31] D. Walton. *A Pragmatic Theory of Fallacy*. The University of Alabama Press, 1995.

[32] S. Wells, C. Gourlay, and C. Reed. Argument blogging. In *9th International Workshop on Computational Models of Natural Argument (CMNA 9)*, 2009.

[33] S. Wells and C. Reed. A domain specific language for describing diverse systems of dialogue. *Journal of Applied Logic*, to appear, 2010.

[34] S. Willmott, G. Vreeswijk, M. South, C. Chesnevar, G. Simari, J. McGinis, I. Rahwan, C. Reed, and S. Modgil. Towards an argument interchange format for multiagent systems. In N. Maudet and I. Parsons, S. Rahwan, editors, *Third International Workshop on Argumentation in Multi-Agent Systems*, 2006.

[35] M. Wooldridge, P. McBurney, and S. Parsons. On the metalogic of arguments. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-05)*, 2005.