Applied Ontology 2 (2007) 267–303
IOS Press

Ontology negotiation in heterogeneous multi-agent systems: The ANEMONE system

Jurriaan van Diggelen, Robbert-Jan Beun, Frank Dignum, Rogier M. van Eijk and John-Jules Meyer
Institute of Information and Computing Sciences, Utrecht University, The Netherlands
E-mails: {jurriaan,rj,dignum,rogier,jj}@cs.uu.nl

Abstract. In open heterogeneous multi-agent systems, communication is hampered by lack of common ontologies. Ontologies may differ in naming conventions, granularity and scope. In such an environment, the agents must possess the right conversational skills to effectively exchange information even when the speaker's ontology is only approximately translatable to the hearer's ontology. Furthermore, the agents must be able to autonomously establish an ontology translation by exchanging parts of their ontologies. In this paper, we propose a layered communication protocol in which the agents gradually build towards a semantically integrated system by establishing minimal and effective common ontologies. We tested our system, called ANEMONE, on a number of heterogeneous news agents. We show how these agents successfully exchange information on news articles, despite initial difficulties caused by heterogeneous ontologies.

1. Introduction

Research in agent communication languages, such as KQML (Finin et al., 1994) and FIPA ACL (FIPA, 2002), has embraced the notion of ontology to enable effective information exchange between agents. Agents sharing the same ontology can exchange their knowledge fluently as their knowledge representations are compatible with respect to the concepts regarded as relevant and with respect to the names given to these concepts. The most straightforward way to realize a common ontology would be to develop one standardized ontology which is used by all agents. However, this scenario would be very unlikely in open multi-agent systems, as those on the Internet, because it would require all involved system developers to reach consensus on which ontology to use. Moreover, an agent's ontology is highly related to the tasks that an agent is designed to perform (Chandrasekaran, Josephson & Benjamins, 1999). A standardized ontology forces an agent to abandon its own subjective world view and adopt one that is not specifically tailored to its task. This may result in a suboptimal situation.

In an open multi-agent system, communication problems that arise from heterogeneous ontologies should be *solved*, rather than avoided. In this paper, we present a suitable solution, called ANEMONE: An Effective Minimal Ontology Negotiation Environment, first introduced in van Diggelen et al. (2006). The purpose of ANEMONE is to establish *effective* communication using a *minimal* common ontology. We assume that the agents may have different areas of expertise, which is reflected in ontologies of different granularity and scope. In such an environment, effective communication is a subtle issue as a concept in one agent's ontology may not be precisely translatable in another agent's ontology. We will use the formal notions of *sound* and *lossless* communication (van Diggelen et al., 2004) to state the requirement that sufficient information should flow between the agents in a correct manner. The communication protocol detects when communication is ineffective and applies techniques for ontology exchange to build a common ontology of minimal size. In this way, the agents exchange ontological information on an as-need basis. We thus adhere to the emerging paradigm of ontology negotiation (Bailin & Truszkowski, 2002). Agents first try to cope with the situation as it is; when communication fails to be effective, the agents seek a minimal solution which solves their communication problem at hand.

We illustrate our approach with a case study that involves some semantic integration problems that are typical for open systems. We consider an open community of agents that periodically download news articles on different topics from RSS news feeds.[1] Different news providers are represented by different agents; the taxonomy of news topics supplied by a provider forms the agent's ontology. The fact that different news providers categorize their news differently gives rise to a proliferation of heterogeneous ontologies. The resulting semantic integration problems crop up once the agents start to exchange news articles with each other. For example, consider the agent Ag-M that represents the news provider *Moreover*, and the agent Ag-Y that represents the news provider *Yahoo*. Driven by a user's request to Ag-M for articles on the topic *Basketball*, Ag-M decides to ask Ag-Y for *Basketball* articles. Although Ag-Y has relevant articles on this topic, in Ag-Y's ontology they are classified under *NBA*.[2] Because Ag-Y does not know *Basketball* and Ag-M does not know *NBA* their communication fails.

Traditional approaches, such as standardization and ontology alignment (Noy & Musen, 2000), are not suited for these kinds of semantic integration problems. Standardization efforts in this domain are almost doomed to fail, because the news providers deliberately distinguish themselves from others by using different ontologies. Ontology alignment has been proposed as a technique that enables agents to keep their individual ontologies by making use of *mappings* between the different ontologies. Although this is a step in the right direction, it assumes that the mappings can be pre-defined before the agents start interacting. In our case, it is not known beforehand which ontology mappings are needed due to the openness of the system and the fact that ontologies change from time to time. For example, one of Yahoo's news topics, *Asian Tsunami Disaster*, is clearly a temporary topic.

Drawing on related work on ontology negotiation and on our experiences with the news agent case, we adopt a standpoint on issues as what kind of common ontology should be built up, and when and where this should occur. These standpoints are incorporated in ANEMONE and make it a novel system for ontology negotiation. By implementing ANEMONE and applying it to the news agent case, we show that ANEMONE provides a solution for some semantic integration problems which, to the best of our knowledge, cannot be solved using existing techniques.

An overview of the communication mechanism is presented in Fig. 1. The communication mechanism consists of three layers. The upper layer of the protocol is the Normal Communication Protocol (NCP) which deals with the kind of social interaction that agents normally exhibit when no ontology problems exist in the system. Every conversation starts in the NCP layer. If the agents fail to understand each other, the agents switch to the middle layer in the protocol which is the Concept Definition Protocol (CDP). In this layer, the agents explain the meaning of a concept to each other by exchanging concept definitions. The meaning of a concept is explained in terms of other concepts. If the communication difficulties are so severe that the agents do not even understand each other's concept definitions, the agents switch to the

[1] RSS is a popular XML format for the syndication of news content on the Internet.

[2] *NBA* is the National Basketball Association.

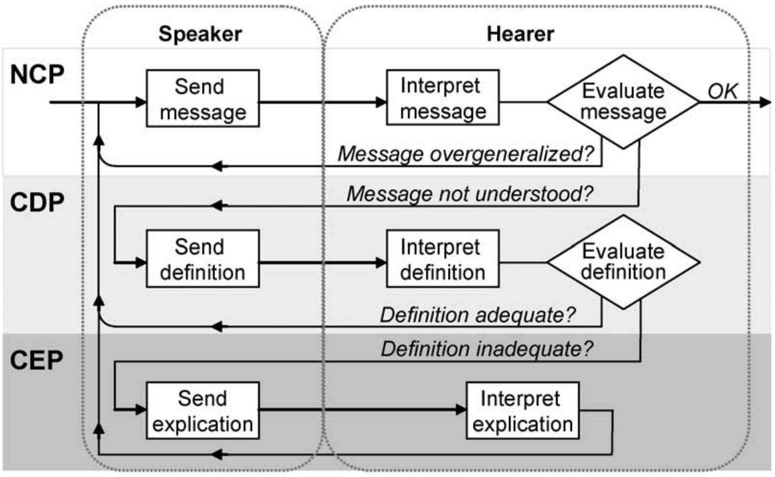

Fig. 1. Overview.

lowest layer in the protocol, i.e. the Concept Explication Protocol (CEP). In CEP, the agents exchange the meaning of a concept using non-symbolic communication, e.g. by pointing to examples of the concept. In this paper, we will design, formalize and validate each operation and decision of the communication mechanism (the boxes and diamonds in Fig. 1, respectively).

The paper is organized as follows. First, we will discuss ontologies and stress those aspects that are important in the context of ontology negotiation. The design objectives of ANEMONE are discussed in Section 3. In Section 4, the communication protocols are discussed. We will give a formal specification for each of the eight boxes and diamonds of Fig. 1. Furthermore, we will prove that the proposed mechanisms actually possess the properties we have claimed. Section 5 presents a case study where we apply ANEMONE to a system with heterogeneous news agents. Although the software implementation regards a specific case, we believe that it contains many aspects that are typical for multi-agent systems. Therefore, the findings of this case study are generalizable to many other multi-agent applications. In Section 6, we describe related work and discuss the relation with our approach. Section 7 presents a discussion, followed by conclusions in Section 8.

2. Ontologies

An ontology is usually understood as a specification of a *shared* conceptualization (Gruber, 1993). As argued in the Introduction section, the sharedness property is difficult to maintain in open heterogeneous multi-agent systems. We therefore do not regard sharedness as an intrinsic property of ontologies, but rather as a useful property. ANEMONE does not make any initial demands on agents with regard to their amount of common concepts. We will discuss an initial situation which is the most difficult. In this situation, the agents do not share any concepts with each other besides the trivial concept *top* ($\top$), which is defined as a superconcept of every other concept.

Also ontologies that are not shared play an important role in an agent's knowledge representation, namely by specifying the agent's subjective *world view* (Davis, Shrobe & Szolovits, 1993). The ontology describes the agent's commitment on how to view the world, which is influenced by the tasks for which the agent will *use* the ontology. For example, an agent that is specialized in Internet news may view the world as consisting of *sports news site* and *politics news site*. By itself, the agent's ontology does not represent any knowledge that is of practical use to the agent. Rather, it introduces a vocabulary which the agent uses to represent and reason with its *assertional knowledge*, i.e. dynamic knowledge regarding facts about the world. For example, the agent's assertional knowledge base may specify that "http://news.bbc.co.uk/sport" is a *sports news site*. This is the type of knowledge that actually matters to the agent. The same phenomenon can be observed in the context of databases. The database schemas themselves, like ontologies, do not contain any useful information. The important information is included in the records of the database, as with the assertional knowledge base. The purpose of agent communication is to exchange assertional knowledge. The only reason to exchange ontological knowledge is to make (parts of) an ontology shared to enable communication of assertional knowledge.

Ontologies may take a variety of forms, but almost without exception they contain concepts in a subsumption hierarchy as their core elements. In our analysis, we avoid introducing unnecessary additional constructs and focus on simple ontologies consisting of sets of concepts and concept relations. A concept relation is one of the following: $\sqsubset$ (strict subconcept relation), $\sqsupset$ (strict superconcept relation), $\equiv$ (equivalence), $\perp$ (disjointness), $\oplus$ (overlap). We will write $c \sqsubseteq d$ as a shorthand for $c \sqsubset d \vee c \equiv d$. Figure 2 presents a graphical representation of two example ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ of agents Ag-1 and Ag-2, respectively.

An arrow between two concepts represents a subconcept relation (and against the flow a superconcept relation). Two concepts in two different branches in the ontology are disjoint. Concepts that are equivalent or overlap are connected with a line with the $\equiv$ or $\oplus$ symbol in it (e.g. Fig. 3). For readability, we have left out concept relations that are derivable from other relations. Formally, an ontology is defined as a labeled graph:

Definition 1. An ontology is defined as $\mathcal{O} = \langle \mathcal{C}, \mathcal{R} \rangle$, where $\mathcal{C}$ is a set of concepts, and $\mathcal{R} = \langle \sqsubset, \sqsupset, \equiv, \oplus, \perp \rangle$ describes the relations between concepts, i.e.

- $\sqsubset$ is a transitive, anti-symmetric relation describing the subconcept relation;
- $\sqsupset$ is a transitive, anti-symmetric relation describing the superconcept relation;
- $\equiv$ is a transitive, symmetric, reflexive relation describing the equivalence relation;
- $\oplus$ is a symmetric relation describing the overlap relation;
- $\perp$ is a symmetric relation describing the disjoint relation.

We will write $\mathcal{C}_i$ to refer to the set of concepts in $\mathcal{O}_i$ (the ontology of agent Ag-i). We will leave out indexes when no confusion arises. We assume that every agent's ontology contains the top concept ($\top$). For all other concepts, we assume that the agents start with a distinct set of concept names. In this way,

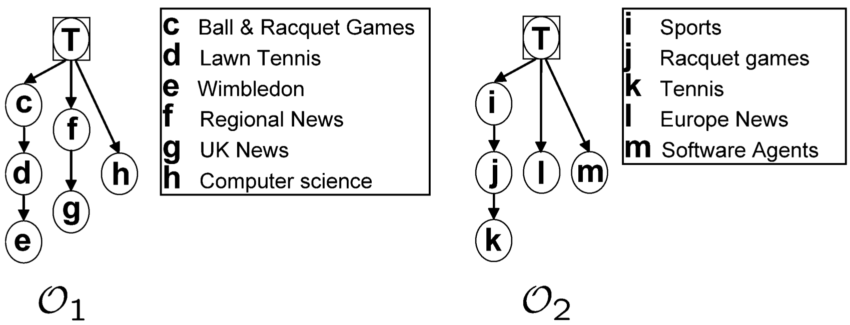

Fig. 2. Initial example ontologies.

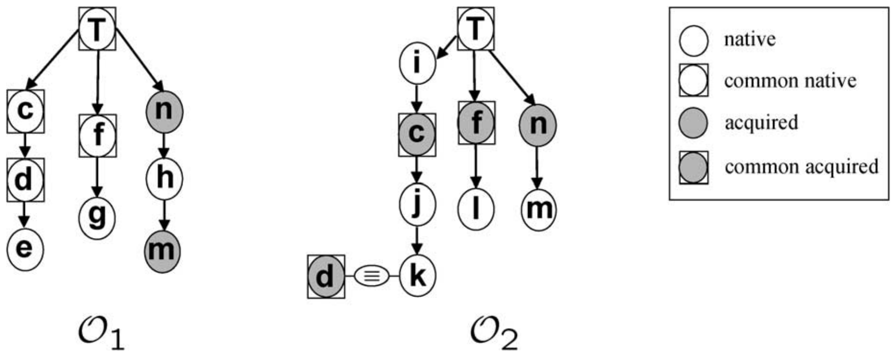

Fig. 3. Evolved example ontologies.

we avoid naming conflicts by ensuring that two concepts in different ontologies with the same name also have the same meaning. This can be easily achieved by using namespaces, i.e. prefixing the agents' concept names with a unique identifier.

In the context of ontology negotiation, some other aspects of ontologies are relevant, which will be discussed in the following two sections.

2.1. Grounded ontologies

In concept explication, one agent must be able to find out how a concept from another agent's ontology relates to the concepts in its own ontology. For example, Ag-2 must be able to find out where Ag-1's concept **c** should be placed in its own ontology. For humans, knowing that concept **c** stands for "Ball & Racquet Games", this problem can be solved with relative ease: concept **c** should be placed as a subconcept of **i** (Sports), as a superconcept of **j** (Racquet Games), and as a disjoint concept of **l** (Europe News) and **m** (Software Agents). This solution is based on human knowledge about the meanings of the natural language descriptions of these concepts. Humans know what we call the *intended interpretations* of these concepts (Genesereth & Nilsson, 1987). To a computer that is not familiar with natural language, the name **c** is just as meaningless as the name "Ball & Racquet Games". Cleverly chosen concept names do not add anything to what is represented for the machine (McDermott, 1981). As a result, the computer is left ignorant about the intended interpretation of its ontological concepts.[3] How the intended interpretation can be modeled in a computer using different means will be the topic of our discussion below.

The intended interpretation is mathematically specified as a function $\mathcal{I}^{INT}$ which maps concept names in $\mathcal{C}$ to subsets of the domain of discourse (Genesereth & Nilsson, 1987). The domain of discourse contains all objects that the agents may wish to speak about. In our example, the domain of discourse consists of the set of URL's on the Internet. Thus, the intended interpretation of concept **c** is the set of all URL's of webpages about "Ball & Racquet Games". We assume that the domain of discourse is shared among all agents. The intended interpretation of an ontological concept can be defined using a formal language, such as description logic (Nardi & Brachman, 2003). The formal semantics of the language specifies how statements in the language relate to the interpretations of concepts. The use of

[3]When we speak of the intended interpretation of an ontological concept, we mean the interpretation that is intended by the *developer* of the ontology.

description logics for ontology negotiation is investigated in van Diggelen et al. (2007). For the purposes of this paper, it is sufficient to state how the concept relations in Definition 1 determine the intended interpretation function.

Assumption 1.

- $c \sqsubset d$ iff $\mathcal{I}^{INT}(c) \subset \mathcal{I}^{INT}(d)$,
- $c \sqsupset d$ iff $\mathcal{I}^{INT}(d) \subset \mathcal{I}^{INT}(c)$,
- $c \equiv d$ iff $\mathcal{I}^{INT}(c) = \mathcal{I}^{INT}(d)$,
- $c \oplus d$ iff $\mathcal{I}^{INT}(c) \not\subseteq \mathcal{I}^{INT}(d) \wedge \mathcal{I}^{INT}(d) \not\subseteq \mathcal{I}^{INT}(c) \wedge \mathcal{I}^{INT}(c) \cap \mathcal{I}^{INT}(d) \neq \emptyset$,
- $c \perp d$ iff $\mathcal{I}^{INT}(c) \cap \mathcal{I}^{INT}(d) = \emptyset$.

This assumption is useful in concept explication for two reasons.

The first reason concerns interpreting concept explications. In CEP, the hearer is provided with examples of the explicated concept, i.e. it is shown the intended interpretation of a concept. Using this information and Assumption 1, it can derive the correct relations between the explicated concept and the concepts in its own ontology. For example, suppose that Ag-1 explicates **c** by showing all webpages about "Ball & Racquet Games", i.e. it communicates $\mathcal{I}^{INT}(\mathbf{c})$. Ag-2 finds out that this set of webpages is a subset of the set of webpages about "Sport", i.e. $\mathcal{I}^{INT}(\mathbf{c}) \subset \mathcal{I}^{INT}(\mathbf{i})$. Using Assumption 1, Ag-2 knows that it may add $\mathbf{c} \sqsubset \mathbf{i}$ to the relations in its ontology $\mathcal{O}_2$.

The second reason concerns sending concept explications. In CEP, the speaker must provide examples of a concept (as in the above example, Ag-1 shows examples of **c**). Therefore, it must know the intended interpretation of the concept. Assumption 1 enables an agent that knows the relations between concepts to have knowledge about the intended interpretation of a concept. For example, if Ag-1 knows that $\mathbf{c} \sqsupset \mathbf{d}$ and that $a \in \mathcal{I}^{INT}(\mathbf{d})$, it may derive that $a \in \mathcal{I}^{INT}(\mathbf{c})$. Unfortunately, the intended interpretation cannot be unambiguously defined by specifying concept relations alone. Because this has important consequences for ontology negotiation, we will elaborate on this issue below.

Suppose that an ontology contains two concepts *even* and *odd* which are interpreted on a domain of discourse containing the natural numbers 1 and 2. Suppose that the ontology specifies that *even* $\perp$ *odd*. Obviously, this concept relation excludes the interpretation that 1, 2 are *even* numbers and that 1 is an *odd* number (because this would yield a non-empty intersection of *even* and *odd*). However, the intended interpretation, i.e. $\mathcal{I}^{INT}(even) = \{2\}$ and $\mathcal{I}^{INT}(even) = \{1\}$ is not uniquely specified either. For example the interpretation that 1 is *even* and 2 is *odd* is not excluded by the statement that *even* is disjoint with *odd*. No other concept relations between *even* and *odd* can be specified to exclude this non-intended interpretation. Even in this very simple example, it is not possible to uniquely capture the intended interpretation by stating concept relations alone.

In symbolic AI, the impracticability to precisely state the relation of a symbol and its meaning is known as the *symbol grounding problem* (Harnad, 1990). In ontology negotiation, the symbol grounding problem is relevant, as the agents must know the precise meaning of their ontological concepts in order to derive the correct relations between different ontologies. A pragmatic solution to the symbol grounding problem is a *grounded ontology* (Steels, 1998; Vogt, 2002). A grounded ontology contains, besides concept relations, also concept classifiers. Agents are well suited to use a grounded ontology as agents perceive the world they inhabit. This perceptive information can be used by an agent to recognize objects in the world as belonging to an ontological concept. This function is performed by the classifier. In semiotics, the idea that symbols require an interpreter in order to be meaningful is well established.

Peirce has formulated this as follows: "The symbol is connected with its object by virtue of the idea of the symbol-using mind, without which no such connection would exist" (Peirce, 1894; par. 7). In our approach, agents are the interpreters which use their classifiers to connect symbols (concepts in the ontology) to objects (webpages).

Technically, a classifier can be implemented using classification techniques that are well-studied in the machine learning community. Formally, a classifier is a function that returns true when an object belongs to a concept, and false when an object does not belong to a concept (according to the concept's intended interpretation). This can be specified as follows.

Specification 1 (*Classifier*). An object a is classified as member of concept c iff $a \in \mathcal{I}^{INT}(c)$.

Note that this specification concerns a *perfect* classifier. In practical situations, the classifiers may be noisy and sometimes return incorrect results. This topic will be discussed in Section 5.

2.2. Native, acquired and common concepts

Figure 2 shows the ontologies of two heterogeneous agents in the initial situation. Figure 3 shows the evolved ontologies of the same agents after they participated in ontology negotiation. For example, one of the things that changed is that Ag-2 has learned the concept **d** from Ag-1 and placed it in its own ontology as an equivalent concept with **k** (representing that *lawn tennis* means *tennis*). For evolved ontologies, some aspects of ontologies are important, which are described below.

Firstly, there is a difference between a *native* concept and an *acquired* concept. In Fig. 3, the former is represented as non-shaded and the latter as shaded. For example, the concept **d** is an acquired concept of Ag-2. An agent that has not yet exchanged any ontological knowledge with other agents, has only native concepts in its ontology, i.e. as in Fig. 2. The native ontology of an agent is implemented by its system developer to enable it to store and reason with assertional knowledge (knowledge that is used by the agent to carry out its task). Because the assertional knowledge base is defined in terms of the native ontology, this ontology is constrained to be static. During ontology negotiation, the agents occasionally teach each other new concepts. These are the agent's acquired concepts and are defined additionally to its native concepts. Acquired concepts are not used for storing knowledge, neither are they accompanied with concept classifiers, i.e. they are not grounded. They only serve the purpose of enabling communication. This allows the acquired ontology to be dynamic. Basically, they specify a *concept mapping*. For example acquired concept **d** in $\mathcal{O}_2$ specifies that Ag-1's concept **d** is mapped to Ag-2's concept **k**. Acquired concepts may also state that one agent's concept is a subconcept of another agent's concept, for example **c** is a subconcept of **i** in $\mathcal{O}_2$. In this way, we avoid the introduction of special mapping operators, such as into- and onto-mappings (Borgida & Serafini, 2003; Stuckenschmidt & Timm, 2002).

Another important issue in ontology negotiation is keeping track of which concepts the agents have in common. In Fig. 3, a concept that is common to Ag-1 and Ag-2 is represented by a box around the concept. When we speak of a concept that is common to two agents, we mean that the concept is common knowledge between the agents, i.e. the agents know that they share the concept (Lewis, 1969). In ontology negotiation, a concept may also be *unknowingly shared*. In this case, both agents know the concept, but do not know this of each other. Examples of unknowingly shared concepts are **n** and **m** in Fig. 3. Such a situation might arise when two agents learn the same concept from a third agent. When we speak of a non-common concept, we refer to a concept that is not common knowledge but may be unknowingly shared.

Formally, we will refer to these issues as follows:

Definition 2. Given a set of concepts $\mathcal{C}$, we distinguish between the following subsets of $\mathcal{C}$

- $\mathcal{C}^{\mathcal{N}}$: The set of native concepts;
- $\mathcal{C}^{\mathcal{A}}$: The set of acquired concepts;
- $\mathcal{C}^{\mathcal{C}\cdot j}$: The set of concepts that the agent has in common with agent j.

In the context of two agents, we will sometimes write $\mathcal{C}^{\mathcal{C}}$ to refer to the set of concepts that the agent has in common with the *other* agent. The following is assumed to hold with respect to these definitions:

Assumption 2.

- $\mathcal{C}_i^{\mathcal{C}\cdot j} = \mathcal{C}_j^{\mathcal{C}\cdot i}$;
- $\mathcal{C}_i^{\mathcal{C}\cdot j} \subseteq \mathcal{C}_j \cap \mathcal{C}_i$;
- $\mathcal{C}_i^{\mathcal{N}} \cap \mathcal{C}_j^{\mathcal{N}} = \{\top\}$;
- $\mathcal{C}^{\mathcal{N}} \cap \mathcal{C}^{\mathcal{A}} = \emptyset$.

The first assumption states that the concepts that Ag-i knows to be shared with Ag-j are the same concepts as those that Ag-j knows to be shared with Ag-i. The second assumption states that concepts that are common to Ag-i and Ag-j occur in the ontologies of both agents, but that not necessarily all concepts that occur in both agents' ontologies are common concepts. The third assumption states that, as argued before, the native concepts of two different agents have different names (except for $\top$). The last assumption states that no concept is both native and acquired.

Having introduced a conceptual framework for ontology negotiation, we will now turn our attention to the precise design objectives of ANEMONE.

3. Design objectives

The purpose of this section is to describe the domain independent design choices of the ontology negotiation protocol (Fig. 1). For example, how do the agents traverse through the protocol layers and what general techniques can be used in the individual layers. These design choices are motivated by a number of design objectives that state what the ontology negotiation protocol should be able to achieve.

Before we discuss the design objectives, we introduce some terminology that is used for describing communication issues throughout the rest of the paper. We distinguish between what the speaker intends to convey, what is actually stated in the message, and how this is interpreted by the hearer. We use the following Latin verbal forms to describe this difference for each layer of the communication protocol. The *transferendum* is the concept that the speaker intends to convey in NCP. It translates this concept to the *transferens*, which is the concept used in the message. The hearer receives the *transferens* and translates it to the *translatum*, which is the hearer's interpretation of the message. In CDP, the speaker intends to define the *definiendum*. It uses the *definiens* in the message, which is interpreted by the hearer as the *definitum*. Likewise, in CEP, the speaker intends to explicate the *explicandum*, which it sends in the message using the *explicans*, which is interpreted by the hearer as the *explicatum*. This is summarized in Fig. 4.

	Speaker		Message		Hearer
NCP	transferendum	$\rightarrow$	transferens	$\rightarrow$	translatum
CDP	definiendum	$\rightarrow$	definiens	$\rightarrow$	definitum
CEP	explicandum	$\rightarrow$	explicans	$\rightarrow$	explicatum

Fig. 4. Terminology.

The goal of an ontology negotiation protocol is to enable the agents to build up something that enables them to reach a desired level of coordination. This raises questions like *what* should be built up during ontology negotiation, *where* should they build it up and *when* should they do that? The design objectives of ANEMONE are stated by answering these questions:

- What? *Minimal & Effective*.
- When? *Lazy*.
- Where? *Decentralized*.

These objectives are explained in more depth below.

3.1. Minimal and effective

The ontology negotiation protocol should enable the agents to build up a common ontology that is minimal and effective. The objective of minimality applies to both the size and the use of the common ontology. An ontology that is minimal in size can be processed efficiently as it does not contain bulks of superfluous concepts. A case study (Giovannucci & Rodríguez-Aguilar, 2005) points out that this is a serious issue in practice, as management of large ontologies might cause significant overload. An ontology that is minimal in use enables agents to keep their conversations short, leading to more efficient interactions.

The counterpart of minimality is effectiveness. The common ontology should contain enough concepts to enable the agents to convey sufficient information in a sound manner. Particularly, communication should be *lossless*, meaning that, from the perspective of the hearer, no information is lost in the communication process. When communication is lossless, spending effort on extending the common ontology does not contribute to the amount of information that can be conveyed between the agents.

According to Grice (1975), cooperative communication is guided by four basic rules. One of these rules concerns the *quantity of information exchange* and another rule concerns the *quality of information exchange*. Both of these rules are incorporated in approach.

Sound communication concerns the quality of information exchange. This means that the translatum should be a correct translation of the transferendum. A large amount of literature exists on which translations in various domains qualify as correct (Janssen, 1998). For the purpose of communicating agents with heterogeneous ontologies, we simply define correctness as logical consequence. We have defined this notion more precisely in a related article (van Diggelen et al., 2007). Related work that captures a similar idea is based on channel theory (Schorlemmer & Kalfoglou, 2005). For here, it suffices to state that the translatum should be a consequence of the transferendum. Formally, this is defined as follows:

Definition 3 (*Sound communication*). Let $c \in \mathcal{C}_i^{\mathcal{N}}$ be the transferendum, and $e \in \mathcal{C}_j^{\mathcal{N}}$ be the translatum. Communication is sound iff $\mathcal{I}^{\mathit{INT}}(c) \subseteq \mathcal{I}^{\mathit{INT}}(e)$.

Lossless communication concerns the quantity of information exchange. It states that the translatum is the most specific consequence of the transferendum. This means that the hearer cannot distinguish the translatum from the transferendum, because its ontology cannot express the difference. This notion is inspired by the observation that for different heterogeneous agents, different indiscernibility classes apply (Doherty, Lukaszewicz & Szalas, 2003). To define lossless communication, the following terminology is useful.

Definition 4. Let S be a set of concepts, c is most specific in the set S iff $c \in S \wedge \neg\exists c' \in S.\mathcal{I}^{INT}(c') \subset \mathcal{I}^{INT}(c)$.

Lossless communication can then be defined as follows:

Definition 5 (*Lossless communication*). Let $c \in \mathcal{C}_i^{\mathcal{N}}$ be the transferendum, and $e \in \mathcal{C}_j^{\mathcal{N}}$ be the translatum. Communication is lossless iff e is most specific in the set $\{e' \in \mathcal{C}_j^{\mathcal{N}} | \mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e')\}$.

A channel theoretic investigation of communication between heterogeneous agents is given in Schorlemmer & Kalfoglou (2005).

Example 1. Consider the evolved ontologies of Ag-1 and Ag-2 in Fig. 3,

1. The following conversation is *sound and lossless*: Ag-1 "translate" transferendum **j** (Racquet games) to transferens **c** (Ball & Racquet games) which Ag-1 interprets as translatum **c** (Ball & Racquet games). This is because **c** (Ball & Racquet games) is the most accurate translation of **j** (Racquet games) that is possible in Ag-1's ontology.
2. The following conversation is *not sound*: Ag-2 translates transferendum **i** (Sports) to transferens **c** (Ball & Racquet games) which Ag-1 interprets as translatum **c** (Ball & Racquet games). This is because something that qualifies as Sports does not necessarily qualify as Ball & Racquet games.
3. The following conversation is *not lossless*: Ag-2 translates transferendum **k** (Tennis) to transferens **c** (Ball & Racquet games) which Ag-1 interprets as translatum **c** (Ball & Racquet games). This is because **k** (Tennis) can be translated more accurately in Ag-1's ontology, namely as **d** (Lawn Tennis).

Creating a common ontology that is only minimal in size is trivial: adopt an empty ontology. Creating a common ontology that is only effective is also trivial: make the common ontology the union of all native ontologies in the system. The combination of a minimal and effective ontology, however, is non-trivial. The trade-off between the two is what makes this objective challenging.

3.2. Lazy

The lazy[4] objective states that ontology exchange should occur on an as-need basis. The agents should only put effort into building a common ontology when strictly necessary. Ontology exchange is not a goal in itself, but a means to enable communication.

To meet the lazy objective, the agents should only leave NCP when communication fails to be effective and should try to return as soon as possible. Furthermore, the agents should only exchange ontological information in CEP when this cannot be done in CDP. This is because defining a concept in CDP is

[4]The term *lazy* is borrowed from the computer programming community where *lazy evaluation* refers to a technique that postpones the computation of an expression until it is known that the results of the computation are actually needed.

less resource-consuming than learning a concept in CEP. In other words, for the agents to be lazy, they should stay as high up as possible in the layered communication protocol.

3.3. Decentralized

In line with the ontology negotiation paradigm, no central location exists where a common ontology is built up. Every agent increments its own ontology with the necessary concepts. In this way, they collectively address the semantic integration problem, solving communication problems between themselves when they arise. Multi-agent systems are usually thought to lack a central control. This justifies a decentralized solution to interoperability problems.

4. Communication protocols

4.1. Normal communication protocol

In this section, we will specify the upper layer of the layered communication mechanism (Fig. 1). In the light of the design objectives stated in the previous section, we will propose a way to send messages, to interpret messages and to evaluate messages. We will start our discussion with sending messages.

4.1.1. Sending messages

Sending a message requires the speaker to translate the transferendum to a transferens. Below, we will discuss two issues involved in this process, namely approximate message composition, and speaking in non-common concepts.

A naive ontology negotiation protocol would require the transferens to be equal to the transferendum in order to ensure lossless communication. In this way, the speaker does not cause any information loss in the communication process. For example, to communicate transferendum **k**, Ag-2 should choose concept **d** which is equal to concept **k**. However, in order to make maximal use of common concepts, the speaker should also be allowed to choose a transferens which is more general than the transferendum. We call this feature approximate message composition. For example, to communicate transferendum **j**, Ag-2 may choose transferens **c**. This conversation leads to lossless communication, as shown in Example 1.1 and is therefore perfectly acceptable. A minimal common ontology can only be effectively applied if the agents make use of this possibility and try to get information across even if this can only be done in more general terms. Marvin Minsky has nicely summarized this aspect of communication as follows: “Whatever we may want to say, we probably won’t say exactly that” (Minsky, 1988).

Another important issue in ontology negotiation concerns speaking in non-common concepts. For the agents, it is beneficial to use common concepts as these concepts are guaranteed to be known by the other agent. However, a concept may also be unknowingly shared, as argued in Section 2.2. For example, suppose that Ag-2 wishes to communicate the (unknowingly shared) concept **m**. If it chooses the non-common concept **m** as transferens, the conversation will be effective because concept **m** is known by Ag-1. To avoid unnecessary ontology exchange (i.e. to be lazy), the agents should be allowed to convey a message using a non-common concept, which may turn out to be an unknowingly shared concept. In human communication this is very common (Barr, 2004). For example, consider two people communicating who have never met. The speaker assumes that his words will be understood by his conversation partner, but does not know this for certain.

The following specification formalizes these ideas.

Specification 2 (*Send Message*). Let c be the transferendum, let d be most specific in the set $\{cc \in \mathcal{C}^{\mathcal{C}} | c \sqsubseteq cc\}$, the transferens is

- d, or
- d', where $c \sqsubseteq d'$ and $d' \sqsubset d$ and $d' \in \mathcal{C}$.

In the above specification, the transferens is either d or d'. The specification allows for approximate message composition because the transferens (both d and d') is a superconcept or equivalent concept of transferendum c. The specification allows speaking in non-common concepts because the speaker may choose between the common concept d, and the non-common concept d'. However, the specification limits the range in which the speaker may compose approximate messages or speak in non-common concepts.

Firstly, the speaker is allowed to use a common transferens that is more general than the transferendum but which is not needlessly more general. Therefore, the specification states that if there are multiple common superconcepts of the transferendum, the speaker must choose the *most specific* one. In this way, the specification minimizes information loss.

Secondly, the speaker is allowed to use a non-common transferens only when this enables the transferendum to be conveyed more specifically than would be possible using a common concept. Therefore, the specification states that if the agent uses non-common transferens d' instead of common transferens d, d' must be a strict subconcept of d. In this way, the specification avoids incomprehension by the hearer as much as possible.

Example 2. Consider the evolved ontologies of Ag-1 and Ag-2 in Fig. 3,

1. The transferendum is **k**. Ag-2 may only use **d** as a transferens.
2. The transferendum is **e**. Ag-1 may use **e** or **d** as a transferens, but not **c**.
3. The transferendum is **m**. Ag-2 may use **m** or **n** or $\top$ as a transferens.

In the second example, it would be better for Ag-1 to use **d** instead of **e** because **d** is known by Ag-2 and not overgeneralized. However, Ag-1 does not know that **d** is not overgeneralized for Ag-2. In the third example, it would be better for Ag-2 to use **m** instead of **n** or $\top$, because **m** is unknowingly shared and not overgeneralized (such as $\top$). In both examples, the speaker faces the choice to use a common transferens with the risk of losing too much information, or to use a more specific non-common transferens, with the risk of not being understood by the hearer. Because there is no general way to make this decision, as shown by the examples, we have left both options open in the specification. Nevertheless, the agents can often make an informed guess on which option is more likely to be successful. Such heuristics are studied in van Diggelen, Wiering & de Jong (2006) and van Diggelen et al. (2005).

4.1.2. Interpreting messages

Interpreting a message requires the hearer to translate the transferens to a translatum. As argued before, the translatum must be a native concept, because the agent's assertional knowledge base is defined in terms of native concepts. To ensure sound communication, the translatum must be a super- or equivalent concept of the transferens. To minimize information loss, the hearer must use the most specific translatum. This can be specified as follows.

Specification 3 (*Interpret message*). Let d be the transferens, the translatum is e, where e is most specific in the set $\{nc \in \mathcal{C}^{\mathcal{N}} | d \sqsubseteq nc\}$.

Example 3. Consider the evolved ontologies of Ag-1 and Ag-2 in Fig. 3,

1. The transferendum is **e**. Ag-1 translates **e** to transferens **d**. Ag-2 translates **d** to translatum **k**.
2. The transferendum is **m**. Ag-2 translates **m** to transferens **m**. Ag-1 translates **m** to translatum **h**.
3. The transferendum is **m**. Ag-2 translates **m** to transferens $\top$. Ag-1 translates $\top$ to translatum $\top$.
4. The transferendum is **g**. Ag-1 translates **g** to transferens **f**. Ag-1 translates **f** to translatum $\top$.

Given Specification 2 and 3 for sending and receiving messages, we can prove that communication is sound as defined by Definition 3.

Property 1. *Communication is sound.*

Proof. Suppose $c \in \mathcal{C}_i^{\mathcal{N}}$ is the transferendum, d the transferens and $e \in \mathcal{C}_j^{\mathcal{N}}$ the translatum. Specification 2 for sending a message ensures that $c \sqsubseteq d$. Using Assumption 1, it follows that $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(d)$. Specification 3 for interpreting a message ensures that $d \sqsubseteq e$. Using Assumption 1, it follows that $\mathcal{I}^{INT}(d) \subseteq \mathcal{I}^{INT}(e)$. Thus $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e)$, conforming to Definition 3. $\square$

The communication mechanism as discussed until now, however, does not always give rise to lossless communication as shown by Example 3.

Thus, an additional check must be performed by the hearer to recognize such non-lossless conversations.

4.1.3. Evaluating messages

Message interpretation as described by Specification 3 is not always possible. When the speaker uses a non-common transferens, the concept may be unknown to the hearer. In this case, the hearer cannot interpret the message and evaluates the message as "Message not understood" (see Fig. 1).

When the transferens is known by the hearer, a check must be performed if the conversation was lossless. There are two methods for recognizing lossless communication.

The first method is performed by the speaker. If it translates the transferendum to an *equivalent* transferens, it lets this be known by sending a message of the "ExactInform" type. If it applies approximate message composition, i.e. the transferens is more general than the transferendum, it sends a message of the "Inform" type. In case an "ExactInform" message is sent, communication is guaranteed to be lossless, as follows straightforwardly from Definition 5 and Specification 3.

The second method applies when an "Inform" message is sent, in which case it is more difficult to assess lossless communication. The hearer must distinguish between the lossless communications using "Inform" and the non-lossless ones. Non-lossless communication means that the message is overgeneralized. The hearer recognizes overgeneralization by reasoning as follows. Upon hearing that an object belongs to the transferens, it knows that the object is a member of every super- or equivalent concept of the transferens and that the object is not a member of all concepts that are disjoint with the transferens. This knowledge cannot be a symptom of overgeneralization. However, the hearer remains ignorant about the sub- and overlapping concepts of the transferens. This ignorance may be a symptom of overgeneralization. Thus, concepts that may indicate overgeneralization are native sub- or overlapping concepts of the transferens.

However, not all native sub- or overlapping concepts of the transferens indicate overgeneralization. Remember that Specification 2 requires the speaker to translate the transferendum to the *most specific* common transferens (or to a non-common subconcept of that). The aim of this measure was to prevent the speaker from becoming more general than necessary. However, it can also be used to enable the hearer to form a belief about what the speaker intended to convey and what it did not. In philosophy of language, such a derivation is known as a *conversational implicature* (Grice, 1975). In ANEMONE, it works as follows: the hearer knows that the speaker does *not* intend to convey the common subconcepts of the transferens, otherwise it would have used a different transferens. It therefore knows that these concepts do not indicate overgeneralization. Moreover, the hearer can reason as follows. It also knows that the speaker did *not* intend to convey any non-common subconcepts of the common subconcept of the transferens. This particular type of conversational implicature, which concerns the quantity of information exchange, is also known as a scalar implicature (Levinson, 1983). These ideas are formalized as follows:

Specification 4 (*Evaluate message*). Let d be the transferens, and p be the type of the message

- If $d \notin \mathcal{C}$, then *Message not understood*;
- If $p =$ "Inform" and $\exists nc \in \mathcal{C}^{\mathcal{N}}$ for which
  1. $nc \sqsubset d$ or $nc \oplus d$, and
  2. $nc \notin \{cc \in \mathcal{C}^{\mathcal{C}} | cc \sqsubset d\}$, and
  3. $nc \notin \{scc \in \mathcal{C} | \exists cc \in \mathcal{C}^{\mathcal{C}}.cc \sqsubset d \wedge scc \sqsubset cc\}$,

  then *Message overgeneralized*;
- Else *OK*.

Example 4. Consider the evolved ontologies of Ag-1 and Ag-2 in Fig. 3,

1. The transferendum is **e**. Ag-1 translates **e** to transferens **d** and sends an Inform message. Ag-2 translates **d** to translatum **k**. Ag-2 correctly determines that communication was lossless because no sub- or overlapping concept of **d** exists.
2. The transferendum is **g**. Ag-1 translates **g** to transferens **f** and sends an Inform message. Ag-2 translates **f** to translatum $\top$. Ag-2 determines that communication may not have been lossless as **l** is a native subconcept of **f** and replies "Message overgeneralized". Ag-1 tries again to get the transferendum **g** across by "translating" to transferens **g** and by sending an ExactInform message. Ag-2 does not understand **g** and replies "Message not understood". The agents traverse through the CDP and CEP layers of the protocol after which Ag-2's ontology is extended with a common acquired concept **g** which is a subconcept of **l**. Ag-2 translates transferens **g** to translatum **l** and assesses lossless communication because the message was sent using an ExactInform message.
3. The transferendum is **j**. Ag-2 translates **j** to transferens **c** and sends an inform message. Ag-1 "translate" **c** to translatum **c**. Ag-1 detects native subconcepts of **c**, namely **d** and **e**. However, it does not regard these as indicators of overgeneralization. Ag-1 reasons as follows: if Ag-2 had intended to convey **d** or **e**, it should have used **d** as the transferens instead of **c**. In the specification, **d** and **e** do not trigger a "Message overgeneralized" response because **d** is a common subconcept of **c** and **e** is a subconcept of **d**. Thus, Ag-1 responds "OK" and the conversation finishes.

The following property states that the communication mechanism leads to lossless communication, as defined by Definition 5.

Property 2. *If the hearer evaluates a message as OK, then communication is lossless.*

Proof. Suppose $c \in \mathcal{C}_i^{\mathcal{N}}$ is the transferendum, d the transferens and $e \in \mathcal{C}_j^{\mathcal{N}}$ the translatum. We prove the theorem by showing that the situation where the hearer evaluates the message as OK while communication was *not* lossless leads to a contradiction. Non-lossless communication means that e is *not* a most specific concept in the set $\{e' \in \mathcal{C}_j^{\mathcal{N}} | \mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e')\}$ (Definition 5 does not hold). This means that either e is not in the set $\{e' \in \mathcal{C}_j^{\mathcal{N}} | \mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e')\}$ (option A), or that e is not a *most specific* element in that set (option B). We will show that both options lead to a contradiction. Option A is not valid because $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e)$ (Property 1) and $e \in \mathcal{C}_j^{\mathcal{N}}$ (Specification 3). If option B holds, then some e'' exists in the set $\{e' \in \mathcal{C}_j^{\mathcal{N}} | \mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e')\}$, such that $\mathcal{I}^{INT}(e'') \subset \mathcal{I}^{INT}(e)$. Now consider the relation between e'' and transferens d. e'' is not disjoint with d, because they have a common subconcept, i.e. $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e'')$ and $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(d)$. Furthermore, it is not the case that $\mathcal{I}^{INT}(d) \subseteq \mathcal{I}^{INT}(e'')$, because then e'' should be the translatum instead of e, as e'' would be most specific in the set $\{nc \in \mathcal{C}^{\mathcal{N}} | d \sqsubseteq nc\}$ (see Specification 3). Thus, either $e'' \sqsubset d$ or $e'' \oplus d$. Now consider the three conditions of "Message overgeneralized" in Specification 4. If we take $nc = e''$, condition 1 is satisfied. Hence, either condition 2 or 3 must not be satisfied, otherwise the hearer would have assessed "Message overgeneralized". For the condition 2 to fail, there must be a concept $cc \in \mathcal{C}^{\mathcal{C}}$ for which $cc \sqsubset d$, and $e'' \equiv cc$. For condition 3 to fail, there must be a concept $cc \in \mathcal{C}^{\mathcal{C}}$ for which $cc \sqsubset d$, and $e'' \sqsubset cc$. Hence, there is a concept $cc \in \mathcal{C}^{\mathcal{C}}$ for which $cc \sqsubset d$ and $e'' \sqsubseteq cc$. Thus, $\mathcal{I}^{INT}(c) \subseteq \mathcal{I}^{INT}(e'') \subseteq \mathcal{I}^{INT}(cc) \subset \mathcal{I}^{INT}(d)$. Therefore, cc should be the transferens instead of d, because d is not most specific in the set $\{cc' \in \mathcal{C}^{\mathcal{C}} | c \sqsubseteq cc'\}$ (see Specification 2). Hence, option B also leads to a contradiction, which proves the theorem. □

4.2. *Concept definition protocol*

If the hearer assesses "Message not understood" (as in Example 3), the agents switch to the CDP layer of the communication mechanism. In the concept definition protocol, the speaker tries to convey the meaning of a concept by stating the relations with the common concepts in its ontology. In this way, it defines the concept which was not understood.

In the specification below, relations, such as $\sqsubset$, are represented as sets of pairs. Because a concept may be related to a common concept in five different ways ($\sqsubset, \sqsupset, \equiv, \oplus, \perp$), the definiens is specified as a tuple consisting of five relations, i.e. sets of pairs. This is specified in the definition below.

Specification 5 (*Send definition*). Let c be the definiendum. The definiens is: $\langle\{\langle c, cc\rangle \in \sqsubset | cc \in \mathcal{C}^{\mathcal{C}}\}, \{\langle c, cc\rangle \in \sqsupset | cc \in \mathcal{C}^{\mathcal{C}}\}, \{\langle c, cc\rangle \in \equiv | cc \in \mathcal{C}^{\mathcal{C}}\}, \{\langle c, cc\rangle \in \oplus | cc \in \mathcal{C}^{\mathcal{C}}\}, \{\langle c, cc\rangle \in \perp | cc \in \mathcal{C}^{\mathcal{C}}\}\rangle$.

The hearer interprets the definition by adding the concept relations to its ontology. For example, the subconcept relation becomes $\sqsubset \cup \sqsubset'$ after the definition is interpreted. This states that the relation $\sqsubset$ as defined by the ontology before message interpretation, is extended with the relations $\sqsubset'$ that are stated in the definition. The following specification formalizes this.

Specification 6 (*Interpret definition*). Let the definiens be $\langle \sqsubset', \sqsupset', \equiv', \oplus', \perp' \rangle$, the definitum is: $\mathcal{O} = \langle \sqsubset \cup \sqsubset', \sqsupset \cup \sqsupset', \equiv \cup \equiv', \oplus \cup \oplus', \perp \cup \perp' \rangle$.

An agent regards the meaning of an acquired concept *complete*, if it knows the relation with every other concept in its ontology. If the definition enables the hearer to derive the complete meaning of a concept (i.e. the definitum is complete), the hearer regards the definition as adequate and switches back to NCP. If the hearer is left with an incomplete meaning, it regards the definition as inadequate and switches to CEP. This is specified as follows:

Specification 7 (*Evaluate definition*). Given a definiens for concept c. Let the definitum be $\mathcal{O} = \langle \mathcal{C}, \mathcal{R} \rangle$:

- If $\forall c' \in \mathcal{C}.\, \exists R \in \mathcal{R}.\langle c, c' \rangle \in R$, then *Definition adequate*;
- Else *Definition inadequate*.

Example 5. Consider the evolved ontologies of Ag-1 and Ag-2 in Fig. 3,

1. The definiendum is **e**. Ag-1 composes a definiens which states that **e** $\sqsubset$ **d** and **e** $\sqsubset$ **c** and **e**$\perp$**f**. Ag-2 adds these relations to $\mathcal{O}_2$ which becomes the definitum. Ag-2 derives that **e** is disjoint with **f**, **l**, **n**, **m**, that **e** is subconcept of $\top$, **i**, **c**, **j**, **k**, **d**. It therefore assesses that the meaning of **e** is complete and evaluates the definition as adequate.
2. The definiendum is **j**. Ag-2 composes a definiens which states that **j** $\sqsubset$ **c** and **j** $\sqsupset$ **d** and **j**$\perp$**f**. Ag-1 adds these relations to $\mathcal{O}_1$ which becomes the definitum. Ag-1 derives that **j** is disjoint with **f**, **g**, **n**, **h**, **m**, that **j** is subconcept of $\top$, **c**, and that **j** is a superconcept of **d**, **e**. It therefore assesses that the meaning of **j** is complete and evaluates the definition as adequate.
3. The definiendum is **g**. Ag-1 composes a definiens which states that **g** $\sqsubset$ **f** and **g**$\perp$**d** and **g**$\perp$**c**. Ag-2 adds these relations to $\mathcal{O}_2$ which becomes the definitum. Ag-2 derives that **g** is disjoint with **i**, **c**, **j**, **k**, **d**, **n**, **m** and that **g** is a subconcept of **f**, $\top$, but it can not derive the relation with **l**; it therefore assesses that this meaning is incomplete and evaluates the definition as inadequate.

The first two examples show the successful application of the concept definition protocol to establish lazy ontology reconciliation. The speaker succeeds in conveying the complete meaning of an unknown concept by giving a concept definition. Thus, the speaker has obviated the need to visit the concept explication protocol. The last example shows that the concept definition protocol enables the hearer to correctly recognize an incomplete meaning. It determines that concept definitions provide insufficient means to convey the complete meaning of a concept, and it switches to CEP.

4.3. Concept explication protocol

When no adequate definition of the concept in terms of other concepts can be given, the agents convey the meaning of a concept non-symbolically in CEP.

In CEP, the speaking agent communicates a number of positive and a number of negative examples of the concept to be explicated. In the specification below this is formalized using a pair of two sets, i.e. one set containing the positive examples (objects belonging to the concept), and one set containing the negative examples (objects not belonging to the concept).

Specification 8 (*Send explication*). Let c be the explicandum, the explicans is: $\langle \{p | p \in \mathcal{I}^{INT}(c)\}, \{n | n \notin \mathcal{I}^{INT}(c)\} \rangle$.

The hearer classifies these examples using the concept classifiers from its own ontology. For each native concept in its ontology, the hearer creates a *confusion matrix*, containing the information from Fig. 5 (Kohavi & Provost, 1998):

	Classified as negative	Classified as positive
Actual negative	$n1$	$n2$
Actual positive	$n3$	$n4$

Fig. 5. Confusion matrix.

In this matrix, $n1$ is the number of negative examples of the explicated concept that were classified as negative; $n2$ is the number of negative examples that were classified as positive; $n3$ is the number of positive examples that were classified as negative; $n4$ is the number of positive examples that were classified as positive. This matrix can be used to calculate the true positive rate (TPR), and the true negative rate (TNR) as follows:

- $\text{TPR} = \frac{n4}{n3+n4}$;
- $\text{TNR} = \frac{n1}{n1+n2}$.

The true positive rate represents the proportion of positive examples that have been positively classified as belonging to a concept. The true negative rate represents the proportion of negative examples that have been classified as not belonging to a concept. In case the number of positive examples is zero, we assume that the value for TPR equals 1. Likewise, when the number of negative examples is zero, the TNR value is assumed to be 1.

4.3.1. CEP under ideal circumstances

In the ideal situation, when the agent's classifiers are perfect and every element in the domain of discourse is used as an example, TPR and TNR offer strict criteria for determining the correct concept relation. These criteria are described next. Consider an ideal situation, where TPR and TNR result from applying a classifier of a native concept nc to examples of the explicated concept c. The relation between concept nc and c can be determined as follows:

- $nc \equiv c$ if $\text{TPR} = 1$ and $\text{TNR} = 1$;
- $nc \perp c$ if $\text{TPR} = 0$;
- $nc \sqsubset c$ if $0 < \text{TPR} < 1$ and $\text{TNR} = 1$;
- $nc \sqsupset c$ if $\text{TPR} = 1$ and $\text{TNR} < 1$;
- $nc \oplus c$ if $0 < \text{TPR} < 1$ and $\text{TNR} < 1$.

The first condition states that nc is equivalent with c if all members of c are positively classified as nc, and all non-members of c are negatively classified as nc. The second condition states that nc is disjoint with c if all members of c are negatively classified as nc. The third condition states that nc is a subconcept of c if some, but not all, members of c are positively classified as nc and all non-members of c are negatively classified as nc. The fourth condition states that nc is a superconcept of c if all members of c are positively classified as nc and some non-members of c are positively classified as nc. The last condition states that nc overlaps with c if some, but not all, members of c are negatively classified as nc and some non-members of c are positively classified as nc.

In the ideal situation, with a finite domain of discourse, and with disregard to computational costs, this concept explication protocol guarantees correct results. However, in most practical situations, the classifiers of the agents are not perfect. Furthermore, the domain of discourse may be infinite, in which case it is not possible to exchange every element in the domain of discourse. In this case, these criteria for determining concept relationships are too strict. In the next section, we show how they can be loosened.

4.3.2. CEP under noisy circumstances

There are actually many ways to loosen the criteria for determining concept relations. We propose a simple approach by introducing some threshold parameters θ_1, θ_2 and θ_3. Using these parameters, the TPR and TNR values can be divided into five regions that correspond to different concept relations as done in Fig. 6.

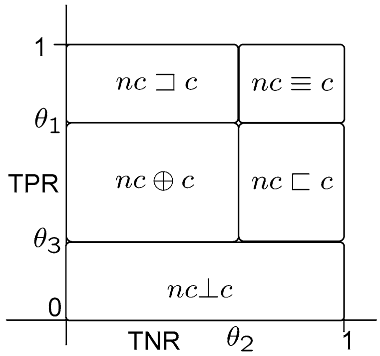

Fig. 6. CEP boundaries.

Note that, by using values $\theta_1 = 1$, $\theta_2 = 1$ and $\theta_3 = 0$, we obtain the same strict criteria for ideal concept explication described in the previous section. By using a looser set of values (i.e. $\theta_1 < 1$, $\theta_2 < 1$, $\theta_3 > 0$), the effects of noisy concept classifiers can be taken into account. The next specification describes the concept explication protocol for noisy circumstances.

Specification 9 (*Interpret explication*). Let $\langle P, N \rangle$ be explicans for concept c. For every $nc \in \mathcal{C}^{\mathcal{N}}$: let TPR and TNR result from applying the classifier of nc to examples of concept c; add $\langle nc, c \rangle$ to relation R, where

- $R = \equiv$ if TPR $\geqslant \theta_1$ and TNR $\geqslant \theta_2$;
- $R = \perp$ if TPR $\leqslant \theta_3$;
- $R = \sqsubset$ if $\theta_3 <$ TPR $< \theta_1$ and TNR $\geqslant \theta_2$;
- $R = \sqsupset$ if TPR $\geqslant \theta_1$ and TNR $< \theta_2$;
- $R = \oplus$ if $\theta_3 <$ TPR $< \theta_1$ and TNR $< \theta_2$.

The more strict the threshold parameters are set, the more often an agent determines the concept relation $\oplus$. This has a number of consequences for the future interactions between the agents. With regard to message interpretation, an acquired concept that overlaps with most of the agent's native concepts can be said to be less meaningful than an acquired concept that is subset, equivalent or disjoint with most native concepts. This is because a concept that overlaps with a native concept does not enable the agent to derive any positive or negative information about the native concept. On the other hand, an acquired concept that is subset or equivalent with a native concept, enables the agent to derive positive information about the native concept. A concept that is disjoint with a native concept enables the agents to derive negative information about the native concept.

Thus, the agents interpret messages as containing more information when they use loose threshold parameters in concept explication than when they use strict threshold parameters. A loose set of thresholds also increases the number of incorrect message interpretations, as agents might derive too much information from their acquired concepts. In information retrieval, such a trade-off between the correctness and the quantity of information transfer is characterized by the measures *precision* and *recall* (Robertson, 2000). Precision measures how much of the information that is transferred is actually correct. Recall measures how much of the correct information is actually transferred. Stated in these terms, a loose set of threshold values leads to interactions with low precision, but with high recall. A strict set of threshold values leads to interactions with high precision, but with low recall.

4.4. Message protocol

Until now, we have only discussed the general principles that underlie the communication mechanism. To establish successful agent communication, a protocol is required which specifies which messages can be sent at which times. The message protocol of ANEMONE is defined using a finite state machine, depicted in Fig. 7. This figure regards the same communication mechanism as Fig. 1, but the focus here is on the messages that are exchanged between the agents. In this protocol, Ag-i is (initially) the speaker and Ag-j is the hearer.

The agents start their conversation in state 1, where the speaker composes a message. In case the speaker translates what it intends to convey to an equivalent concept, it sends an "ExactInform" message. Otherwise, it sends an "Inform" message. The hearer interprets the message in state 2 or 3. In case it does not understand the concept used in the message, it sends a "StartCDP" message. In case it regards the message as overgeneralized, which only occurs in state 2, it requests for specification by sending a "ReqSpec" message. In case the hearer understands the message and does not regard it as overgeneralized, it answers "OK", after which the conversation finishes in state 4.

The concept definition protocol starts in state 5. In case the speaker does not have any common concepts to compose a concept definition, it sends a "ProposeStartCEP" message. Otherwise, it sends a "Define" message. In state 6, the hearer interprets the concept definition. It sends an "ExitCDP" message when it regards the definition as complete and sends a "StartCEP" message when it regards the meaning as incomplete.

In state 7, the speaker explicates the concept by sending an "Explicate" message. After this, the agents return to the CDP layer. In state 6 the hearer sends an "ExitCDP" message and the agents return to the NCP layer.

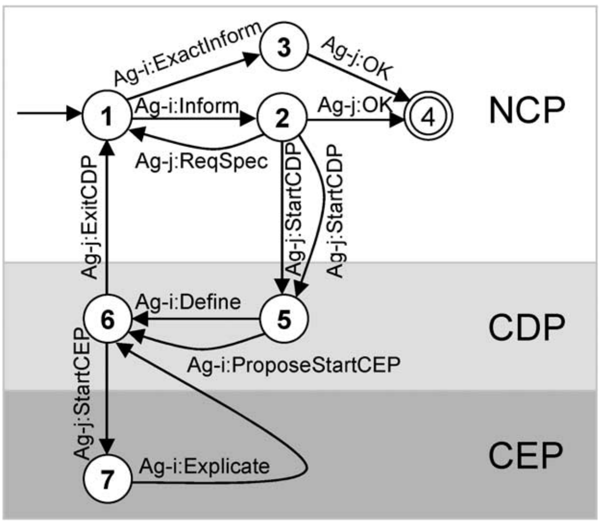

Fig. 7. Message protocol.

5. Case study

This section demonstrates an application of the techniques discussed in this paper.[5] We will focus on a case with heterogeneous RSS news feeds on the Internet. The ontologies that are involved are real news taxonomies as currently used on the Internet. The structure of these ontologies corresponds to the ontologies discussed in Section 2. The instances are real news articles as published at the time of the experiment. The concept classifiers are implemented by text classifiers which the agents train themselves. We used agents as wrappers around news feed and allowed them to communicate about their news articles. They reconciled their ontologies using ANEMONE. The case study aims at demonstrating the following:

- Whether the assumptions and objectives of ANEMONE are justified in a realistic domain;
- Whether the dialogue mechanisms of ANEMONE are successful in a realistic domain;
- Whether our approach to concept explication is fruitful in a realistic domain.

Although the study regards a single case, we believe that the findings are characteristic for open multi-agent systems in general.

5.1. Application domain

Over the last few years, RSS (Rich Site Summary or Really Simple Syndication) has become a popular format for the syndication of news content on the Internet (Powers, 2005). Using RSS, a news provider distributes its web content as an XML file, called an RSS feed. The advantage of using RSS is that it enables the provider to add machine-readable metadata to its news content. This metadata can be used to identify different parts of the article, such as the title, a summary, a link to the full article, and the date of publication.

An RSS aggregator is a useful tool for people that wish to stay updated on a number of favorite web sites. After the user has selected a number of RSS feeds, the RSS aggregator constantly scans the content of these feeds for updates, and presents any new items to the user. This enables the user to see his or her latest favorite news items at one glance without having to repeatedly visit several Internet sites.

Usually, a news provider provides several news feeds on different categories. A selection of the various news feeds that are provided by four major news providers BBC,[6] MoreOver,[7] Reuters[8] and Yahoo[9] is given in Fig. 8.

As appears from this overview, different news providers have adopted very different ways to classify their news articles. A proliferation of classification schemes which is typical for today's Internet is also characteristic of RSS news feeds. Comparing the names *Science/Nature* by BBC, *Science* by Reuters and *Science News* by Yahoo, we observe that naming conventions are absent. Furthermore, the news providers classify their news at differing levels of granularity. For example, Moreover subdivides sports in 20 different topics, whereas Reuters uses only one news feed to cover sports.

For the user of an RSS aggregator, this heterogeneity of classification schemes has the following consequences. Firstly, a user that is interested in, for example, science, should manually discover and

[5]The prototype presented in this paper is freely available on the first author's home page.

[6]News feeds from the BBC. Available at http://news.bbc.co.uk/1/hi/help/3223484.stm.

[7]Free RSS News Feeds Listing – Moreover Technologies. Available at http://w.moreover.com.

[8]Latest New, Reuters.com. Available at http://today.reuters.com/rss/newsrss.aspx.

[9]Yahoo! News – RSS. Available at http://news.yahoo.com/rss.

BBC	MoreOver	Reuters	Yahoo
News Front Page	Tennis	Top News	U.S. National
World	Boxing	Business News	Sports News
UK	Basketball	Science	NBA
Business	Law news	Bird Flu	Science News
Politics	McDonalds news	International	Asian Tsunami Disaster
Science/Nature	Jokes	Politics	Iraq
around 15 others...	*around 330 others...*	*around 10 others...*	*around 120 others...*

Fig. 8. RSS News Feeds.

sign up for the different science related news feeds, e.g. the *Science/Nature* feed at BBC, the *Science* feed at Reuters and *Science News* feed at Yahoo. Considering the amount of news feeds available, this may be a very laborious task. Secondly, after the user has discovered and selected the desired news feeds, the news feeds covering the same topic are represented in a poorly organized list. No integrated view is provided that combines news feeds belonging to the same topic. For example, we would like one topic header "Science" in which all science related articles stemming from any news feed are collected.

For our case study, we propose a prototype of a "next generation" RSS aggregator: a personal news agent (PNA). The purpose of this application remains presenting news articles to the user in a well organized way. However, compared to an RSS aggregator, a personal news agent provides some additional functionality:

- The PNA automatically discovers new news sources.
- The PNA does not simply *aggregate* RSS news feeds but also *integrates* them.

To realize these goals, we implement a multi-agent system consisting of news agents that represent different RSS news providers. The categorizations of topics used by the news providers are the agents' ontologies. The ontologies are grounded by text classifiers. A user signs up to one of the news agents which then becomes its personal news agent. The PNA not only acquires news articles by downloading them from its own RSS news source, but also communicates with other agents to obtain news articles from other sources. For example, a PNA that represents Reuters communicates with a news agent representing Yahoo to obtain articles belonging to the *Science News* feed. Thus, a PNA can be said to discover new news sources by communicating with other news agents in the system.

When a PNA acquires a news article from another news agent, it arranges this news article according to its own list of topics. For example, when Reuters' agent acquires a news article stemming from Yahoo's *Science News* feed, it does not create another topic *Science News*, but places the article under the already existing topic *Science*. In this way, the list of news topics provided to the user remains conveniently arranged. As a result, the PNA can be said to integrate news feeds, instead of simply aggregating them.

By using agents to represent news providers, a multi-agent system is obtained that is open, dynamic, and heterogeneous (Weiss, 2001). The system is open because news providers may start and finish their services at any time. The system is dynamic as the agents may change from time to time, e.g. Yahoo's topic "Asian Tsunami Disaster" is clearly a temporary topic. The heterogeneous lists of news topics used by the news providers give rise to heterogeneous ontologies in the system.

To enable the news agents to communicate, some semantic integration problems crop up which are typical for open heterogeneous MAS's. By making every agent follow the ANEMONE communication mechanism, the agents are capable of overcoming these problems.

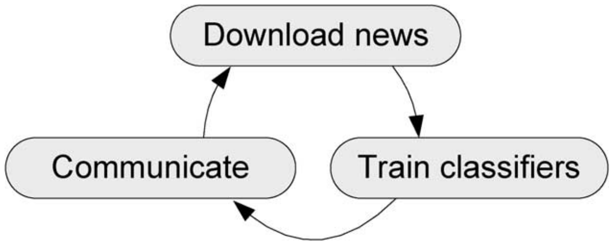

Fig. 9. Main cycle of a news agent.

5.2. *Implementation*

We have implemented the system as a multi-threaded Delphi[10] application. A single news agent corresponds to an RSS aggregator. The main cycle of a news agent is depicted in Fig. 9. We will discuss each step in the cycle below.

5.2.1. *Download news*

A news agent periodically checks the Internet whether any new news items have arrived on one of the RSS feeds it has been subscribed to. These subscriptions are specified in a separate text file which can be altered by the user. If any new item has arrived, it uses an XML parser to extract the fields: title, description, link, and pubDate. It stores this information in a local database which constitutes the assertional knowledge base of the agent. This database can be consulted by the person who uses the PNA to stay updated on the latest news articles. Basically, this component allows for the same functionality as an RSS aggregator.

5.2.2. *Train classifiers*

To obtain the native concept classifiers, the agent trains text classifiers using the items that it has downloaded from its news provider. As training examples, we use the description field of a news item. Because the agent knows to which topic these descriptions belong, we can apply a supervised learning algorithm.

Before the text in the description is used as training data for the classification algorithm, some preprocessing is performed. The texts are preprocessed using common techniques in natural language processing (Jackson & Moulinier, 2002), such as *stemming* and *stopword removal*. Subsequently, the texts are transformed in *term frequency vectors* which indicate, regardless of word order, the number of times that each word occurs in the input sample.

After the description fields are preprocessed, they are used as training data for the classification algorithm. To train a classifier to recognize instances of a certain class, it must be provided with positive and negative examples of the class. Positive examples are the (preprocessed) descriptions of the news items belonging to the topic. Negative examples are the (preprocessed) descriptions of the news items of other topics. For example, to train the classifier for BBC's *Science/Nature* concept, the positive examples are taken from *Science/Nature* and the negative examples from *business* and *UK*.

In our implementation, we have used a support vector machine (SVM) with a linear kernel function, i.e. corresponding to a linear classification (Vapnik, 1995). An in-depth discussion of this algorithm is beyond the scope of this paper. The motivation of using an SVM is that this algorithm has been shown to yield good results with limited amounts of training samples (Joachims, 1999). This is a relevant issue

[10]http://www.borland.com/us/products/delphi/index.html.

in the news agent case as most RSS news providers only publish around four news articles per topic per day. If neural networks were used to learn the concept classifiers, it would probably take too long before the agents could participate in ontology negotiation. We have adopted the implementation SVM^{light} (Joachims, 1999).

5.2.3. Communication

Communication between the agents proceeds according to the communication mechanisms described in this paper. The NCP and CDP layers can be readily implemented in the news agent system. Below, we will describe the implementation of the CEP layer in the communication protocol, where agents learn concepts from other agents.

Agents are capable of participating in CEP after the agents have sufficiently trained their concept classifiers on the news articles they have downloaded themselves. In CEP, the teacher sends descriptions of news articles belonging to the concept to be explicated as positive examples, and descriptions of news articles belonging to the concepts that are disjoint with the concept to be explicated as negative examples. The hearer transforms the descriptions to term frequency vectors as described in the previous section. For every concept, it applies the corresponding support vector machine to classify these examples. It computes TPR and TNR values according to the rules described in Section 4.3 and derives the concept relations.

Because in our application, a small amount of misclassifications is acceptable, we used relatively tolerant criteria to assess concept relations, i.e. $\theta_1 = 0.75$, $\theta_2 = 0.75$, $\theta_3 = 0.4$.

5.3. Results

We demonstrate the system using four agents. The agents represent news publishers BBC, Moreover, Reuters and Yahoo. The agents' ontologies consist of relatively small subsets of the news feeds provided by the news publishers, as depicted in Fig. 10. After the agents have collected news articles for a period of two months, their knowledge bases were filled with approximately 200 news articles per topic. This enabled them to train their classifiers and participate in ontology negotiation. The following examples

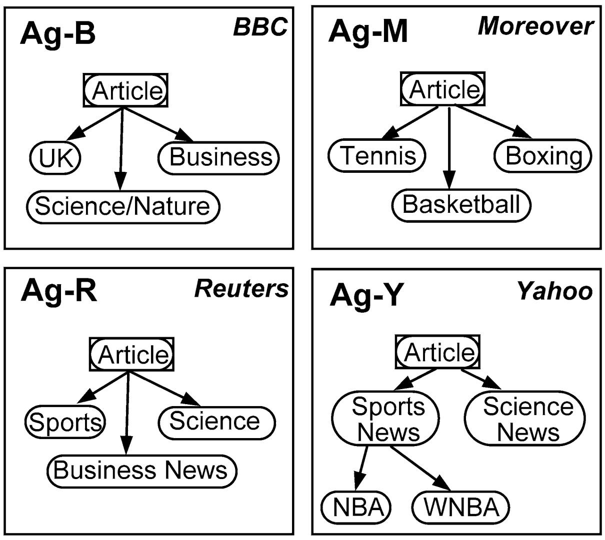

Fig. 10. Initial situation.

present a series of successive dialogues which were generated by the agents. For demonstration purposes, the agents make some of their internal workings public. Sentences preceded by a * are not part of the dialogue, but are a log of the agent's internal reasoning. Due to space limitations, we abbreviated long path names of news articles with `[]`. The agents begin in a situation where they do not share any concept with each other (Fig. 10). The dialogue that follows is shown below. After the agents have had this dialogue, the situation is as in Fig. 11.

Ag-Y intends to inform Ag-R about Science News.

```
1 Ag-Y:<Ag-R,<ExactInform,Science News,[...]>>
2 Ag-R:<Ag-Y,<StartCDP>>
3 Ag-Y:<Ag-R,<ProposeStartCEP>>
4 Ag-R:<Ag-Y,<StartCEP>>
5 Ag-Y:<Ag-R,<Explicate,Science News,[...],[...]>>
* Ag-R Business News: TPR=0.15 TNR=0.86
* Ag-R Science: TPR=0.76 TNR=1
* Ag-R Sports: TPR=0 TNR=0.24
* Ag-R derives equivalent(Science News,Science)
6 Ag-R:<Ag-Y,<ExitCDP>>
7 Ag-Y:<Ag-R,<ExactInform,Science News,[...]>>
* Ag-R translates to Science
8 Ag-R:<Ag-Y,<OK>>
```

Ag-R intends to inform Ag-Y about Science.

```
9 Ag-R:<Ag-Y,<ExactInform,Science News,[...]>>
10 Ag-Y:<Ag-R,<OK>>
```

Ag-Y intends to inform Ag-M about Science News.

```
11 Ag-Y:<Ag-M,<ExactInform,Science News,[...]>>
12 Ag-M:<Ag-Y,<StartCDP>>
13 Ag-Y:<Ag-M,<ProposeStartCEP>>
14 Ag-M:<Ag-Y,<StartCEP>>
15 Ag-Y:<Ag-M,<Explicate,Science News,[...],[...]>>
* Ag-M Basketball: TPR=0.29 TNR=0.32
* Ag-M Tennis: TPR=0.21 TNR=0.84
* Ag-M Boxing: TPR=0.26 TNR=1
* Ag-M derives disjoint(Science News,Basketball;Tennis;Boxing)
16 Ag-M:<Ag-Y,<ExitCDP>>
17 Ag-Y:<Ag-M,<ExactInform,Science News,[...]>>
* Ag-M translates to Article
18 Ag-M:<Ag-Y,<OK>>
```

Ag-R intends to inform Ag-M about Science.

```
19 Ag-R:<Ag-M,<ExactInform,Science News,[...]>>
* Ag-R translates to article
20 Ag-M:<Ag-R,<OK>>
```

Ag-R intends to inform Ag-Y about Sports.

```
21 Ag-R:<Ag-Y,<ExactInform,Sports,[...]>>
22 Ag-Y:<Ag-R,<StartCDP>>
23 Ag-R:<Ag-Y,<Define Sports,<disjoint Science News>>>
24 Ag-Y:<Ag-R,<StartCEP>>
25 Ag-R:<Ag-Y,<Explicate,Sports,[...],[...]>>
* Ag-Y NBA: TPR=0.17 TNR=0.94
* Ag-Y WNBA: TPR=0.13 TNR=0.99
* Ag-Y Science News: TPR=0.08 TNR=0.22
* Ag-Y Sports News: TPR=0.90 TNR=0.81
* Ag-Y derives equivalent(Sports,Sports News)
26 Ag-Y:<Ag-R,<ExitCDP>>
27 Ag-R:<Ag-Y,<ExactInform,Sports,[...]>>
* Ag-Y translates to Sports News
28 Ag-Y:<Ag-R,<OK>>
```

Ag-Y intends to inform Ag-R about NBA.

```
29 Ag-Y:<Ag-R,<Inform,Sports,[...]>>
30 Ag-R:<Ag-Y,<OK>>
```

Ag-R intends to inform Ag-Y about Business News.

```
31 Ag-R:<Ag-Y,<ExactInform,Business News,[...]>>
32 Ag-Y:<Ag-R,<StartCDP>>
33 Ag-R:<Ag-Y,<Define,Business News,<disjoint Sports,disjoint Science News>>>
34 Ag-Y:<Ag-R,<ExitCDP>>
35 Ag-R:<Ag-Y,<ExactInform,Business News,[...]>>
* Ag-Y translates to Article
36 Ag-Y:<Ag-R,<OK>>
```

Ag-Y intends to inform Ag-M about Sports News.

```
37 Ag-Y:<Ag-M,<ExactInform,Sports,[...]>>
38 Ag-M:<Ag-Y,<StartCDP>>
39 Ag-Y:<Ag-M,<ProposeStartCEP>>
40 Ag-M:<Ag-Y,<StartCEP>>
41 Ag-Y:<Ag-M,<Explicate,Sports,[...],[...]>>
* Ag-M Basketball: TPR=0.47 TNR=0.70
* Ag-M Tennis: TPR=0.27 TNR=0.78
* Ag-M Boxing: TPR=0.03 TNR=0.73
* Ag-M derives overlaps(Sports,Basketball)
* Ag-M derives disjoint(Sports,Tennis;Boxing)
42 Ag-M:<Ag-Y,<ExitCDP>>
43 Ag-Y:<Ag-M,<ExactInform,Sports,[...]>>
* Ag-M translates to Article
44 Ag-M:<Ag-M,<OK>>
```

Ag-Y intends to inform Ag-M about NBA.

```
45 Ag-Y:<Ag-M,<Inform,Sports,[...]>>
* Ag-M translates to Article
46 Ag-M:<Ag-Y,<ReqSpec>>
47 Ag-Y:<Ag-M,<ExactInform,NBA>>
48 Ag-M:<Ag-Y,<StartCDP>>
49 Ag-Y:<Ag-M,<Define NBA,<subset Sports News>>>
50 Ag-M:<Ag-Y,<StartCEP>>
51 Ag-Y:<Ag-M,<Explicate,NBA,[...],[...]>>
* Ag-M Basketball: TPR=1 TNR=0.51
* Ag-M Tennis: TPR=0 TNR=0.83
* Ag-M Boxing: TPR=0 TNR=0.85
* Ag-M derives subset(NBA,Basketball)
* Ag-M derives disjoint(NBA,Tennis; Boxing)
52 Ag-M:<Ag-Y,<ExitCDP>>
53 Ag-Y:<Ag-M,<ExactInform,NBA,[...]>>
* Ag-M translates to Basketball
54 Ag-M:<Ag-Y,<OK>>
```

After this conversation, the agents have built up knowledge about each other's ontologies, as shown in Fig. 11. The agents have built up a common ontology that enabled them to convey what they intended to convey (effectiveness), but have not made the common ontology larger than required (minimality). The acquired concepts are not only mapped to equivalent native concepts but also to native superconcepts (e.g. Ag-M's *NBA* and *Basketball*) and to disjoint native concepts (e.g. Ag-M's *Science News*). Approaches that only deal with equivalence mappings would have failed to solve the semantic integration problems of the news agents.

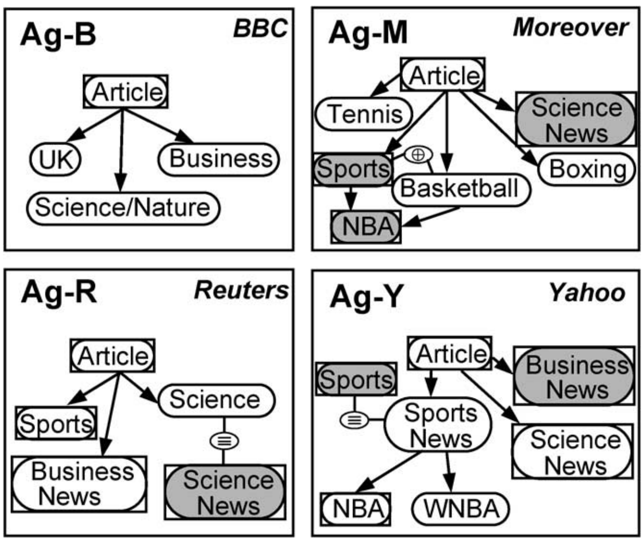

Fig. 11. Situation after 9 conversations.

5.4. Evaluation

In this section, we will evaluate the results of the case study in the light of the three objectives stated at the beginning of Section 5. We will begin our discussion with the first objective which was to justify the assumptions and objectives of ANEMONE.

5.4.1. Justification of assumptions and objectives

The proliferation of heterogeneous ontologies in the news agent system is inherent to the way the system is built up. Standardization efforts in this domain are almost doomed to fail, because the news providers deliberately distinguish themselves from others by using different ontologies.

Whereas ontology standardization is not feasible in this domain, the domain of discourse has already been standardized by the RSS standard. This accomplishes that all agents can easily access the summaries and URL's of the news articles about which they communicate. Furthermore, it establishes a uniform way to represent the news texts. For example, it prevents misunderstandings caused by one agent representing its texts in ASCII and the other in PDF.

This situation confirms our intuition that the domain of discourse is easier to standardize than the agents' ontologies. We believe that it is typical in open systems that some things are standardized while others are not. Most likely, without any standardization at any level, no interaction at all would be possible.

One of the main motivations behind ANEMONE was to apply concept explication as little as possible in order to save resources. In the news agent case, it took around 10 seconds (on a Pentium 4, 2.5 GHz) for the agents to finish concept explication. Although the CEP layer worked well enough for this relatively simple case, in a scenario with more complex ontologies or where the correctness of concept mappings is of critical importance, the CEP layer needs improvements. Most likely, these improved versions will be even more time consuming. This strengthens our motivation for lazy ontology negotiation to reduce the occurrence of concept explication to a minimum.

5.4.2. Dialogue mechanisms

The second objective of the case study was to investigate whether the dialogue mechanisms of ANEMONE are successful in a realistic domain. As appears from the dialogue presented in Section 5.3, many of the ontology negotiation techniques have actually been used. We mention the following:

- The use of mappings other than equivalence mappings (11–18, 45–54);
- The use of approximate message composition (29–30);
- Detecting non-lossless communication (45–46);
- The use of speaking in *unknowingly shared* concepts (19);
- Using acquired concepts for communication in *both* directions (9, 29);
- The use of communicating concept definitions (31–36);
- Detecting inadequate concept definitions (23–24).

Whereas an ontology negotiation protocol provides a nice solution to incrementally establish a communication vocabulary between a *pair* of heterogeneous agents, how this solution scales to *large* multi-agent systems is not straightforward. For example, a desirable property is that every agent eventually uses the same acquired concept to communicate the same meaning. In related work, we have investigated such macro-level properties using simulation experiments with large groups of agents (van Diggelen, Wiering & de Jong, 2006; van Diggelen et al., 2005). These experiments show that convergence of the language, i.e. every agent uses the same acquired concept to communicate the same meaning, may occur within reasonable time.

5.4.3. Concept explication

We believe that the third objective of the case study can be answered positively, i.e. our approach to concept explication is fruitful in a realistic domain. As appears from Fig. 11, most of the derived mappings seem correct. Only some of them, such as *Sports* $\perp$ *Tennis* and *Sports* $\perp$ *Boxing* in Ag-M's ontology, seem incorrect. Nevertheless, also other techniques for ontology mapping have their limitations in this domain, such as invoking Wordnet (Fellbaum, 1998). Firstly, this technique is incomplete, e.g. Yahoo's concept "NBA" is not defined in Wordnet. Secondly, the technique may be incorrect. For example, the "International" concept of the American company Reuters is really a superset of the "UK" concept of the (British) BBC. From the perspective of the BBC agent, Wordnet would suggest differently.

Without going into technical details, we will make some general remarks on how concept explication can be improved. Firstly, it should be clear what qualifies as a correct mapping. What if the concept *Sports*, that was taught by Ag-Y to Ag-M, does not cover tennis news? In that case *Sports* $\perp$ *Tennis* would be a correct mapping. In another experiment, Ag-Y taught the concept *Science News* to Ag-B, after which Ag-B derived that *Science/Nature* $\perp$ *Science News*. Although this mapping seems incorrect, it may also be the case that the *Science News* articles of Yahoo do not meet the scientific standards maintained by BBC's *Science/Nature* topic. Suppose that a French news agent explains its science news topic by showing French example news articles. Clearly, Ag-Y would not regard this concept as equivalent with *Science News*, as its concept classifier was trained on English news articles. It is not self-evident whether this would be correct.

To deal with these issues, we will return to Assumption 1 which relates concept relations to intended interpretations. For the news agent case, this means that the ultimate criterion for correct concept mappings arises from the interpretations of these concepts *as intended by the news providers*. This means that the human developer should be taken into account to evaluate the results. Furthermore, it indicates that the results are highly dependent on the quality of the concept classifiers. The better the classifiers capture the intended interpretations (see Specification 1), the better the results will be. Below, we will elaborate on how the concept classifiers can be improved in our application.

Whereas in Section 2 we have only argued that agents should possess concept classifiers, in this section, we have actually shown a way to obtain these classifiers by training them on news articles. In this training process, we expect that significant improvements can be made. Currently, the agents' concept classifiers are only trained using examples from their own news ontologies. For example, Ag-B has trained its classifiers by only using examples from *UK*, *Business* and *Science/Nature*. When these classifiers are used to classify examples of Ag-Y's concept *Sports News*, they may not work correctly as the classifiers have never been trained using sports articles. This situation can be improved by providing the agents with a wide range of different news articles during their training phase.

6. Related work

In this section, we will discuss four different approaches to deal with semantic interoperability problems in multi-agent systems. Many of these techniques have their origins in more conventional systems such as distributed databases and knowledge systems. We will argue to what extend these techniques are applicable in a multi-agent environment. Finally, we will discuss the emerging paradigm of ontology negotiation which corresponds to the approach presented in this paper.

6.1. Ontology integration

Ontology integration is the most straightforward way to solve interoperability problems in distributed systems. This approach aims at overcoming difficulties with heterogeneous ontologies by replacing them with one common ontology. The common ontology is obtained by integrating every distinct ontology in the system so that it satisfies the needs of all users. Usually, this process involves more than simply putting the distinct ontologies together. This is because of two reasons. Firstly, the ontologies may contain overlapping areas. The same concept in one ontology may be modeled differently in another ontology. In this case, the integration of two ontologies demands one uniform way to model the concept. Secondly, putting two ontologies together does not make explicit the relations between the different ontologies, as these are not described in any of the two ontologies. Therefore, for every two concepts that stem from different ontologies, an additional concept relation must be specified.

Ontology integration has its roots in the database community, where the ontology reconciliation problem is known as the schema integration problem (Batini, Lenzerini & Navathe, 1986). The most widely used schema integration technique resolves conflicts by modifying the initial database schemas, analogous to ontology integration. Because ontologies are more complex structures than database schemas, the techniques that are developed for schema integration require some revision to become applicable to ontology integration. However, the main idea underlying the two approaches is the same.

In the IS community, ontology integration has become a popular technique to deal with heterogeneous ontologies. The main reason for this is that it gets rid of all ontology related interoperability problems at once. The heterogeneous ontologies are simply no longer present. Ontology integration is one of the steps in the On-To-Knowledge Methodology (OTKM) (Sure, Staab & Studer, 2004). OTKM is a methodology for designing and maintaining ontology-based knowledge management applications. In the early modeling stages, the domain experts and potential users propose a set of *concept islands*, i.e. ontologies that model a subdomain of the corresponding domain. Later, these concept islands are integrated to form one uniform ontology. An alternative methodology for constructing shared domain ontologies has been proposed by Aschoff, Schmalhofer & van Elst (2004). The ONIONS methodology (Steve, Gangemi & Pisanelli, 1997) is an example of the ontology integration approach applied to a medical domain.

Because the integration of ontologies is often a laborious task, tools have been developed which assist people in merging ontologies. Examples of such tools are Chimaera (McGuinness et al., 2000), Prompt (Noy & Musen, 2000) and FCA-Merge (Stumme & Maedche, 2001). These tools merge ontologies semi-automatically, i.e. they propose suggestions but leave the ultimate decisions to the system developer. An overview of semi-automatic tools for matching database schemas is given by Rahm & Bernstein (2001).

Despite its popularity in the IS community, it is unlikely that ontology integration provides a suitable solution for ontology reconciliation among agents. This claim is motivated by the following two arguments.

The first argument is that one multi-purpose ontology, which perfectly suits the needs of all agents is not always realizable in practice. An agent that is specialized in a specific task needs an ontology that is tailored to its task. Therefore, agents with different tasks require different ontologies. Ontology integration forces an agent to make concessions in which world view to adopt which could be disadvantageous for its performance.

Our second argument concerns system development. The division of a system in autonomous agents enables a distributed development. System developers that do not even know each other can independently develop autonomous agents without worrying about how this fits in a larger architecture. This advantage does not hold when the system developers must reach consensus on one integrated ontology.

6.2. Ontology alignment

Ontology alignment allows agents to preserve their individual ontologies, leaving their autonomy unaffected. Communication is enabled by a set of pre-defined *mappings*, which specify the relations between the agents' ontologies. Similar to ontology integration, it is assumed that the work of creating ontology mappings is done by humans. Most ontology integration tools (discussed in the previous section) can also be used to create ontology mappings.

In the context of databases, this approach corresponds to global-as-view schema integration (Calí, De Giacomo & Lenzerini, 2001). In the global-as-view approach, the independently developed schemas persist and are integrated by constructing a global view, i.e. a virtual schema that makes the mappings between the individual schemas explicit.

Ontology mappings can be implemented by *point-to-point mappings* and by an *intermediate ontology*. Point-to-point mappings are defined for every pair of ontologies. The agents use these mappings to directly translate to and from each other's ontologies. The disadvantage of this approach is that many mappings have to be defined in order to align the agents' ontologies. A way to reduce the number of mappings is to use an intermediate ontology or interlingua (Ciocoiu, Gruniger & Nau, 2001). The intermediate ontology indirectly aligns the agents' ontologies. The agents use the intermediate ontology by translating the message stated in terms of the speaker's local ontology to the intermediate ontology which the receiver translates back to its own local ontology. In this respect, the approach resembles our approach to ontology reconciliation.

Regardless of whether a point-to-point or an intermediate topology is adopted, the use of ontology alignment in multi-agent systems suffers a major disadvantage: it presumes that the mappings can be established at design-time, before the agents start interacting. This assumption is not very realistic in multi-agent systems because of two reasons. Firstly, in an *open* system, it is not known beforehand which agents will constitute the system. It is therefore impossible to state at design-time which ontology mappings are needed at agent interaction time. Secondly, it is not clear when design-time stops and interaction time starts. As many development methodologies point out (Schreiber et al., 2000), system development should proceed in a number of iterative cycles. According to the so-called *development cycle*, improvements are implemented in each cycle based on the findings from the previous cycle. This means that agents, and their ontologies in particular, also evolve at interaction time (Noy & Klein, 2004). Ontology alignment does not provide the flexibility to deal with changing ontologies, as it assumes that one moment exists after which the agents' ontologies no longer change.

6.3. Ontology mediation services

A more flexible approach to ontology reconciliation is based on *mediation services* (Wiederhold & Genesereth, 1997). In the agent literature, such services are usually referred to as *ontology agents*. According to the FIPA Ontology Service Specification (FIPA, 2000), an ontology agent is expected to facilitate agent communication by registering ontologies, generating mappings between ontologies, and by making translations between ontologies. In other words, an ontology agent provides a central point which can be consulted by agents with communication problems. This approach opens the possibility to reconcile heterogeneous ontologies at agent interaction time, thus being more flexible than ontology alignment.

The KRAFT architecture (Preece et al., 2000) is an example of the ontology mediation approach. In this architecture, knowledge sources, knowledge fusion entities and users are all represented by independent agents. The heterogeneity of the distributed knowledge sources is solved by specialized agents that, among other things, resolve ontology mismatches. OBSERVER (Mena et al., 2000) is another example of a distributed architecture that is based on the ontology mediation approach. This approach makes use of so-called *Interontology Relationships Managers* to process queries in multi-ontology environments.

Different approaches for ontology agents or mediation services differ greatly in how these components are positioned and implemented. Invariable in these approaches is the idea that a mediator is capable of translating between the different ontologies in the system. This establishes that other components are relieved from the necessity of having this capability themselves. The ontology mediation approach can therefore be viewed as a *centralized solution* to semantic interoperability problems.

Although the mediation approach might appear as an attractive solution for interoperability problems in multi-agent systems, it does not solve the problem of how to establish these mappings themselves. Basically, it relocates the problem from the individual agents to the ontology agent. One might argue that an advantage of this relocation is that it is less work to establish these mappings at one location (at the ontology agent), than to establish ontology mappings for every agent. However, the contrary may also hold because at the time the mappings are created, it is not known who will communicate with whom about what. It is therefore also unknown which mappings are actually needed and which are not, which may lead to the establishment of superfluous mappings.

An additional disadvantage of this approach is that it deprives agents of control over establishment and use of ontology mappings. This needs not be a problem in systems with a carefully designed structure where every agent trusts the ontology agent and knows how to reach it. However, in open heterogeneous multi-agent systems, where the need for semantic integration is most pressing, such a well-defined structure is usually absent. It is not self-evident that agents are willing to delegate the task of finding ontology mappings to this ontology agent. An ontology translation that is considered sufficiently accurate by the ontology agent, may be regarded as unacceptable by the individual agents. Furthermore, in open and heterogeneous systems it cannot be guaranteed that the ontology agent does not have malicious intentions. The ontology agent might have an agenda of its own, and deliberately make the wrong translations between ontologies. Basically, these problems arise because a *centralized* solution for ontology reconciliation is applied to an inherently *decentralized* multi-agent system.

6.4. Ontology negotiation

Ontology negotiation (ON) tackles the ontology reconciliation problem in a fully decentralized way. Unlike ontology alignment, no pre-defined mappings are required. Unlike the ontology mediator approach, no intervention of third agents is required. The agents solve the problem between themselves, i.e. the ontology reconciliation problem is tackled decentralized. Another remarkable difference is that this solution does not aim at solving all ontology mismatches at once. Instead, an *incremental* solution is established which deals with ontology problems when they arise. In this way, only those mappings are established which are actually needed. Ontology negotiation thus overcomes the two main objections raised in the previous section against the mediator approach.

Whereas ON is a promising approach, it is also a very ambitious approach. According to Uschold & Gruninger (2002), ontology negotiation can be regarded as *the Holy Grail of semantic integration*. This is mainly because the technique requires agents to be capable of both *detecting* ontology mismatches and of *resolving* them. Whereas other approaches for ontology reconciliation keep the possibilities for human involvement open, detecting and resolving mismatches must proceed fully automatically in ON. Another complication in ON is that the reconciliation process must proceed in a distributed way. The previously described approaches for ontology reconciliation proceeded from a god's eye view over the heterogeneous ontologies. In ON, such a god's eye view does not exist, i.e. two agents that wish to align their ontologies do not have access to each other's ontologies. Agents cannot "look inside each other's head". Because the area is also relatively new, much work remains to be done.

The term *ontology negotiation* was coined in Bailin & Truszkowski (2002). In this paper, the authors present a communication mechanism that enables agents to exchange parts of their ontology in a pattern of successive clarifications. The approach has been applied in the context of scientific archives. Many approaches that investigate ontology reconciliation from an agent perspective, have a strong focus on the automatic generation of ontology mappings. As this is a necessary component of ON, we will review some of these techniques below. The DOGGIE approach (Williams, 2004) focuses on machine learning techniques that can be used to make the meaning of a concept clear to another agent. Burnstein et al. (2003) proposes a technique for meaning exchange between agents that is based on the formal semantics of ontologies, defined in terms of a neutral topic domain. The approach described by Wiesman & Roos (2004) proposes to find ontology mappings based on corresponding instances of concepts and joint attention. Furthermore, many of the automatic ontology mapping techniques that have been proposed outside the agent community are relevant for ON (Maynard et al. (2004) presents an extensive survey).

Automatic ontology mapping is not the only component that is required for successful ON. Other relevant work that focuses on the use and properties of ontology mappings in agent communication is reported by Doherty, Lukaszewicz & Szalas (2003) and Stuckenschmidt & Timm (2002). A formal treatment of ontology mappings in agent communication from a programming perspective is reported by van Eijk et al. (2001). Soh & Chen (2005) have proposed a framework which aims at minimizing ontology exchange during ON. Their approach is based on the idea that ontology exchange is only useful when it improves operational efficiency. The work by Wang & Gasser (2002) describes how mutual ontology alignment in a group of agents can be of benefit for the whole group.

Another important component of ON is a detection mechanism for ontology mismatches. An overview of possible ontology mismatches is described by Visser et al. (1997). A study on how ontology mismatches can be detected is reported by Beun, van Eijk & Prüst (2004). The main focus of this work is on homonymy problems that arise when multiple ontologies are used. For example, a word in one ontology might mean something different than the same word in another ontology. The approach describes a way to detect such mismatches.

Other related work for ON can be found in the literature on negotiation, of which, as its name suggests, ontology negotiation is a special kind. Negotiation protocols are well-studied interaction mechanisms that enable agents with different interests to cooperate (Rosenschein & Zlotkin, 1994). For example, they may be used to make a buyer and a seller agree on prices for certain goods, or in air traffic control, to decide which airplanes are allowed to land first. What ontology negotiation protocols have in common with these protocols is their distributed nature. As this is one of the most striking differences with other ontology reconciliation techniques, *ontology negotiation* is an appropriate name to characterize this approach. There is no central coordinating entity that manages the interactions between the agents, but the agents reach an agreement among themselves. In ontology negotiation the agreement is about a (piece of) common ontology. Similar to other negotiation protocols, the agents' interests may be conflicting. It is easiest for an agent to make other agents adapt to its own ontology, saving the costs of learning foreign ontologies. Of course, not every agent should maintain that policy. The ON protocol serves to resolve such issues.

7. Discussion

7.1. The role of symbolic communication

ANEMONE enables agents to overcome symbolic communication problems by building up a common ontology. One may observe that the things that are symbolically communicable after the common ontology is built up were already non-symbolically communicable before the common ontology was built up. The critic could interpret this observation as evidence that the whole communication protocol is useless. The critic would raise: to inform that individual a is member of concept **c**, why not just send a to the hearer, and let the hearer classify this individual in its own ontology? Apparently, this would solve the problem that concept **c** might not be understandable to the hearer. We will give three reasons why this objection does not hold.

Firstly, the proposal of the critic requires the hearer to use its classifiers. Computationally, this is more costly than symbolic communication. Seen from this perspective, symbolic communication serves a purpose of computational efficiency. Secondly, non-symbolic communication as proposed by the critic requires the individual to be accessible for the hearer at the time of communication. This is not always the case. For example, a sender intends to convey that it possesses a news article, but is only willing to show the article in exchange for money. Thirdly, the proposal of the critic, i.e. to communicate everything by pointing to instances, only works with *inform* messages. In our discussion we have only considered *inform* messages in the NCP layer. For most applications, however, the NCP layer should be extended to deal with other types of messages as well, e.g. *request* or *query*. In ANEMONE, such extensions are rather straightforward, because the agents have built up a communication vocabulary which allows them to communicate symbolically. However, it is not clear how the critic's approach can be extended to deal with *request* or *query* messages.

7.2. Using ANEMONE for hybrid ontology reconciliation

The layered structure of ANEMONE allows for a reasonable amount of flexibility. In this section, we will discuss two hybrid approaches for ontology reconciliation in which ANEMONE is combined with other techniques.

7.2.1. Combination 1

It is always a good investment to standardize, when possible, parts of the agents' ontologies at design-time. Those parts of the ontologies that remain heterogeneous after the standardization effort can be aligned at agent interaction time using ontology negotiation. This is a hybrid approach to ontology reconciliation where standardization is combined with ontology negotiation. Such a hybrid approach also fits well within ANEMONE.

One possibility to start with a standardized ontology is to make the agents aware of this. In this case, the agents start with an ontology that partly consists of common concepts, namely those stemming from the standardized ontology. The NCP layer of the protocol will effectively apply these concepts in communication. In case the agents intend to convey concepts that do not belong to the standardized ontology, the other layers in the protocol will prove themselves useful.

Another possibility to use standardized ontologies in ANEMONE is to namespace the concepts in these ontologies with the (unique) names of the standard ontologies. In this way, the ontologies of agents contain many unknowingly shared concepts. The communication protocol ensures that these unknowingly shared concepts will be effectively applied in communication.

7.2.2. Combination 2

The concept explication method described in this paper may not be satisfactory for all applications. This is when the objects in the domain of discourse are abstract or non-existent. Furthermore, our approach requires an agent to recognize an object in the world as belonging to a concept in its knowledge representation. Therefore, it must be able to perceive the world about which it wishes to represent knowledge. The idea that things can only be known that are, at least some of the times, accessible to the senses has a long philosophical tradition (Kant, 1781). Situated agents fit well in this tradition. Internet agents have access to the Internet, and mobile robots have cameras to perceive the physical world. Nevertheless, for many AI systems, the requirement is not readily met. To quote Roy (2005), many current computer systems are trapped in sensory deprivation tanks, cut off from direct contact with the physical world. For example, a knowledge representation about hotels may be implemented in a computer system that is incapable of perceiving a hotel in the real world. Our approach to concept explication is difficult to apply in such systems.

In these cases, another concept explication method can be used by replacing the CEP layer of the communication protocol. This may involve another technique for ontology reconciliation, leading to a hybrid approach.

One possibility is to incorporate ontology mediation services in the system. Instead of switching to the CEP layer of the protocol, the agents consult the mediation service to find the appropriate ontology mapping. The mediation service may be implemented as a database of ontology mappings or may consist of human ontology engineers. Another possibility is to use translation dictionaries in the CEP layer of the protocol.

Whichever replacement of CEP is chosen, it remains a resource-consuming activity. Therefore, the features of laziness and minimality that are incorporated in the upper layers of the protocol remain useful, even when a different concept explication method is chosen.

8. Conclusion

In this paper, we have presented the ANEMONE system for solving semantic integration problems. Instead of trying to solve all ontology problems at one stretch at design time, ANEMONE provides agents with the tools to overcome ontology problems at agent interaction time. The layered communication mechanism tackles semantic integration problems when needed and only when needed.

We have applied ANEMONE to the domain of news articles. Despite the fact that agents used different ontologies to classify news articles, they were able to overcome their communication problems and successfully exchanged news articles with each other. The performance of ANEMONE can be improved by incorporating more sophisticated machine learning techniques in the lower layer of the protocol. For our application, where the agents use relatively simple ontologies, the concept explication layer worked sufficiently well. The focus in this paper, however, has been on the communication mechanisms which serve to embed the machine learning techniques in the overall communication architecture, i.e. the upper two layers in the protocol. Even in our relatively simple case, these communication mechanisms proved to be useful. The agents refrain from applying resource consuming machine learning techniques as much as possible. When necessary, they gradually contribute to a common ontology which is minimal in size and enables them to convey sufficient information. There is no need for a central coordinating agent; the agents find out by themselves if their communication is not satisfactory and solve the problem between themselves. These features of ANEMONE established that the news article agents, after having participated in only a few conversations, became part of a reasonably semantically integrated system.

In the future, we plan to perform more experiments with personal news agents. In this way, we can evaluate the results more thoroughly. We plan to investigate the influence of the ontology's size, scope and granularity on the results. Furthermore, we plan to experiment with different threshold parameters in concept explication. To assess the correctness of the derived concept relations, we should also take the user into account.

Also, we plan to apply ANEMONE to more complex domains. This may require us to replace the current concept explication protocol with one that is better suited for that domain. Furthermore, the ontologies in that domain may be more complex than the concept hierarchies we have focused on in this paper. This requires an extension of the other layers in the communication protocol to deal with, for example, the alignment of concept attributes. Nevertheless, the use of concept attributes raises the same issues as we have discussed in this paper. Not every concept attribute should necessarily be communicable to other agents in order to achieve effective communication. The agents should find out when communication is not lossless, and add extra concept attributes to their common ontology on an as-need basis.

References

Aschoff, F.-R., Schmalhofer, F. & van Elst, L. (2004). Knowledge mediation: A procedure for the cooperative construction of domain ontologies. In A. Abecker, L. van Elst & V. Dignum (Eds.), *Proceedings of the ECAI-2004 Workshop on Agent-Mediated Knowledge Management (AMKM-2004)* (pp. 29–38), Valentia, Spain.

Bailin, S.C. & Truszkowski, W. (2002). Ontology negotiation between intelligent information agents. *Knowledge Engineering Review*, *17*(1), 7–19.

Barr, D.-J. (2004). Establishing conventional communication systems: Is common knowledge necessary? *Cognitive Science*, *28*, 937–962.

Batini, C., Lenzerini, M. & Navathe, S.B. (1986). A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, *18*(4), 323–364.

Beun, R.J., van Eijk, R.M. & Prüst, H. (2004). Ontological feedback in multiagent systems. In *Proceedings of Third International Conference on Autonomous Agents and Multiagent Systems* (pp. 110–117). New York: ACM Press.

Borgida, A. & Serafini, L. (2003). Distributed description logics: Assimilating information from peer sources. In S. Spaccapietra (Ed.), *Journal of Data Semantics*, LNCS (Vol. 1, pp. 153–184). Springer-Verlag.

Burnstein, M., McDermott, D., Smith, D.R. & Westfold, S.J. (2003). Derivation of glue code for agent interoperation. *Autonomous Agents and Multi-Agent Systems*, *6*(3), 265–286.

Calí, A., De Giacomo, G. & Lenzerini, M. (2001). Models for information integration: Turning local-as-view into global-as-view. In *Proceedings of International Workshop on Foundations of Models for Information Integration*, Viterbo, Italy.

Chandrasekaran, B., Josephson, J.R. & Benjamins, V.R. (1999). What are ontologies, and why do we need them? *IEEE Intelligent Systems*, *14*(1), 20–26.

Ciocoiu, M., Gruniger, M. & Nau, D. (2001). Ontologies for integrating engineering applications. *Journal of Computing and Information Science in Engineering*, *1*(1), 12–22.

Davis, R., Shrobe, H.E. & Szolovits, P. (1993). What is a knowledge representation? *AI Magazine*, *14*(1), 17–33.

Doherty, P., Lukaszewicz, W. & Szalas, A. (2003). On mutual understanding among communicating agents. In *Proceedings of the Workshop on Formal Approaches to Multi-Agent Systems (FAMA'03)* (pp. 83–97), Warsaw, Poland.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Finin, T., Fritzson, R., McKay, D. & McEntire, R. (1994). KQML as an agent communication language. In N. Adam, B. Bhargava & Y. Yesha (Eds.), *Proceedings of the 3rd International Conference on Information and Knowledge Management (CIKM'94)* (pp. 456–463). Gaithersburg, MD, USA.

FIPA (2000). FIPA Ontology Service Specification.

FIPA (2002). FIPA ACL Message Structure Specification.

Genesereth, M.R. & Nilsson, N.J. (1987). *Logical Foundations of Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Giovannucci, A. & Rodríguez-Aguilar, J.A. (2005). On the cost of agent-awareness for negotiation services. In B. Henderson-Sellers & M. Winikoff (Eds.), *Seventh International Bi-Conference Workshop on Agent-Oriented Information Systems AOIS-2005*, Klagenfurt, Austria.

Grice, H.P. (1975). Logic and conversation. In P. Cole & J.L. Morgan (Eds.), *Speech Acts* (pp. 41–58). New York: Academic Press.

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, *5*, 199–220.

Harnad, S. (1990). The symbol grounding problem. *Physica D*, *42*, 335–346.

Jackson, P. & Moulinier, I. (2002). *Natural Language Processing for Online Applications: Text Retrieval, Extraction, and Categorization*. Amsterdam, The Netherlands/Philadelphia, PA, USA: John Benjamins Publishing.

Janssen, T.M.V. (1998). Algebraic translations, correctness and algebraic compiler construction. In *AMiLP'95: Proceedings of the first international AMAST workshop on Algebraic methods in language processing* (pp. 25–56). Amsterdam, The Netherlands: Elsevier Science Publishers B.V.

Joachims, T. (1999). Making large-scale support vector machine learning practical. In *Advances in Kernel Methods: Support Vector Learning* (pp. 169–184). Cambridge, MA: MIT Press.

Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *ICML'99: Proceedings of the Sixteenth International Conference on Machine Learning* (pp. 200–209). San Francisco, CA: Morgan Kaufmann Publishers Inc.

Kant, I. (1781). *A Critique of Pure Reason*. Available: http://philosophy.eserver.org/kant/critique-of-pure-reason.txt.

Kohavi, R. & Provost, F. (1998). Glossary of terms. In *Special Issue on Applications of Machine Learning and the Knowledge Discovery Process* (Vol. 30, pp. 271–274). Boston: Kluwer Academic Publishers.

Levinson, S.C. (1983). *Pragmatics*. Cambridge: Cambridge University Press.

Lewis, D. (1969). *Convention*. Oxford: Blackwell Publishing.

Maynard, D., Stamou, G., Zaihrayeu, I., Barrasa, J., Euzenat, J., Hauswirth, M., Ehrig, M., Bouquet, P., Dieng-Kuntz, R., Hernández, R.L., Tessaris, S., Van Acker, S. & Bach, T.-L. (2004). D2.2.3: State of the art on current alignment techniques. Knowledge Web.

McDermott, D. (1981). Artificial intelligence meets natural stupidity. In J. Haugeland (Ed.), *Mind Design: Philosophy, Psychology, Artificial Intelligence* (pp. 143–160). Cambridge, MA: MIT Press.

McGuinness, D.L., Fikes, R., Rice, J. & Wilder, S. (2000). The chimaera ontology environment. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI 2000)*. Austin, Texas.

Mena, E., Illarramendi, A., Kashyap, V. & Sheth, A. (2000). OBSERVER: An approach for query processing in global information systems based on interoperation across pre-existing ontologies. *International Journal on Distributed and Parallel Databases (DAPD)*, *8*(2), 223–272.

Minsky, M. (1988). *The Society of Mind*. New York: Simon & Schuster.

Nardi, D. & Brachman, R.J. (2003). An introduction to description logics. In F. Baader, D. Calvanese, D. McGuiness, D. Nardi, & P. Patel-Schneider (Eds.), *The Description Logic Handbook* (Chapter 1, pp. 1–40). Cambridge: Cambridge University Press.

Noy, N.F. & Musen, M.A. (2000). Prompt: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Austin, TX, USA.

Noy, N.F. & Klein, M. (2004). Ontology evolution: Not the same as schema evolution. *Knowledge Information Systems*, *6*(4), 428–440.

Peirce, C.S. (1894). What is a sign? Available: http://www.marxists.org/reference/subject/philosophy/works/us/peirce1.htm.

Powers, S. (2005). What are syndication feeds. O'Reilly Media.

Preece, A.D., Hui, K.Y., Gray, W.A., Marti, P., Bench-Capon, T.J.M., Jones, D.M. & Cui, Z. (2000). The KRAFT architecture for knowledge fusion and transformation. *Knowledge Based Systems*, *13*(2–3), 113–120.

Rahm, E. & Bernstein, P.A. (2001). A survey of approaches to automatic schema matching. *The VLDB Journal*, *10*(4), 334–350.

Robertson, S. (2000). Evaluation in information retrieval. In F. Crestani, M. Agosti & G. Pasi (Eds.), *Lectures on Information Retrieval: Third European Summer-School, ESSIR 2000*, Lecture Notes in Computer Science (Vol. 1962, pp. 81–93). New York: Springer-Verlag.

Rosenschein, J.S. & Zlotkin, G. (1994). *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*. Cambridge, MA: MIT Press.

Roy, D. (2005). Semiotic schemas: a framework for grounding language in action and perception. *Artificial Intelligence*, *167*(1–2), 170–205. (Special volume on connecting language to the world.)

Schorlemmer, M. & Kalfoglou, Y. (2005). Progressive ontology alignment for meaning coordination: an information-theoretic foundation. In *AAMAS'05: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 737–744). New York, NY: ACM Press.

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W. & Wielinga, B. (2000). *Knowledge Engineering and Management, The CommonKADS Methodology*. MIT Press.

Soh, L.-K. & Chen, C. (2005). Balancing ontological and operational factors in refining multiagent neighborhoods. In *Proceedings of the Fourth International Conference on Autonomous Agents and Multi-Agent Systems*. New York, NY, USA: ACM Press.

Steels, L. (1998). The origins of ontologies and communication conventions in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, *1*(2), 169–194.

Steve, G., Gangemi, A. & Pisanelli, D.M. (1997). Integrating medical terminologies with ONIONS methodology. In J.P. Charrel & H. Kangassalo (Eds.), *Information Modelling and Knowledge Bases* (Vol. VIII). Amsterdam: IOS Press.

Stuckenschmidt, H. & Timm, I.J. (2002). Adapting communication vocabularies using shared ontologies. In *Proceedings of the Second International Workshop on Ontologies in Agent Systems (OAS)*, Bologna, Italy.

Stumme, G. & Maedche, A. (2001). FCA-MERGE: Bottom-up merging of ontologies. In *IJCAI* (pp. 225–234).

Sure, Y., Staab, S. & Studer, R. (2004). On-to-knowledge methodology (OTKM). In S. Staab & R. Studer (Eds.), *Handbook on ontologies* (Chapter 6, pp. 117–132). Springer.

Uschold, M. & Gruninger, M. (2002). Creating semantically integrated communities on the world wide web. *Semantic Web Workshop Co-located with WWW 2002 Honolulu*.

van Diggelen, J., Beun, R.J., Dignum, F., van Eijk, R.M. & Meyer, J.-J.Ch. (2004). Optimal communication vocabularies and heterogeneous ontologies. In *Developments in Agent Communication*, LNAI (Vol. 3396). Springer.

van Diggelen, J., Beun, R.J., Dignum, F., van Eijk, R.M. & Meyer, J.-J.Ch. (2005). A decentralized approach for establishing a shared communication vocabulary. In *Proceedings of the International Workshop on Agent Mediated Knowledge Management (AMKM'05) held with AAMAS'05*, Utrecht, The Netherlands.

van Diggelen, J., Beun, R.J., Dignum, F., van Eijk, R.M. & Meyer, J.-J.Ch. (2006). ANEMONE: An effective minimal ontology negotiation environment. In P. Stone & G. Weiss (Eds.), *Proceedings of the Fifth International Conference on Autonomous Agents and Multi-agent Systems (AAMAS'06)* (pp. 899–906). New York, NY, USA: ACM Press.

van Diggelen, J., Beun, R.J., Dignum, F., van Eijk, R.M. & Meyer, J.-J.Ch. (2007). Ontology negotiation: Goals, requirements and implementation. *International Journal of Agent-Oriented Software Engineering (IJAOSE)*, *1*(1), 63–90.

van Diggelen, J., Wiering, M.A. & de Jong, E.D. (2006). Strategies for ontology negotiation: Finding the right level of generality. In *Agent Communication II, International Workshops on Agent Communication, AC 2005 and AC 2006*, LNCS (Vol. 3859). Springer.

van Eijk, R.M., de Boer, F.S., van der Hoek, W. & Meyer, J.-J.Ch. (2001). On dynamically generated ontology translators in agent communication. *International Journal of Intelligent Systems*, *16*(5), 587–607.

Vapnik, V.N. (1995). *The Nature of Statistical Learning Theory*. New York, NY: Springer.

Visser, P.R.S., Jones, D.M., Bench-Capon, T.J.M. & Shave, M.J.R. (1997). An analysis of ontological mismatches: Heterogeneity versus interoperability. In *AAAI 1997 Spring Symposium on Ontological Engineering*. Stanford, USA.

Vogt, P. (2002). The physical symbol grounding problem. *Cognitive Systems Research*, *3*(3), 429–457.

Wang, J. & Gasser, L. (2002). Mutual online ontology alignment. In *Proceedings of the Workshop on Ontologies in Agent Systems, held with AAMAS 2002*, Melbourne, Australia.

Weiss, G., Ed. (2001). *Multiagent Systems*. Cambridge, MA/London, UK: MIT Press.

Wiederhold, G. & Genesereth, M. (1997). The conceptual basis for mediation services. *IEEE Expert: Intelligent Systems and Their Applications*, *12*(5), 38–47.

Wiesman, F. & Roos, N. (2004). Domain independent learning of ontology mappings. In *Proceedings of Third International Conference on Autonomous Agents and Multiagent Systems* (pp. 844–851). New York: ACM Press.

Williams, A.B. (2004). Learning to share meaning in a multi-agent system. *Autonomous Agents and Multi-Agent Systems*, *8*(2), 165–193.