

A KNOWLEDGE REPRESENTATION LANGUAGE FOR NATURAL LANGUAGE PROCESSING, SIMULATION AND REASONING

MARJORIE McSHANE^{*,†} and SERGEI NIRENBURG^{†,§}

*Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County, 1000 Hilltop Circle
Baltimore, MD 21250, USA*

**marge@umbc.edu*

†sergei@umbc.edu

‡<http://ilit.umbc.edu/PubMcShane.htm>

§<http://www.trulysmartagents.org/>

OntoAgent is an environment that supports the cognitive modeling of societies of intelligent agents that emulate human beings. Like traditional intelligent agents, OntoAgent agents execute the core functionalities of perception, reasoning and action. *Unlike* most traditional agents, they engage in extensive “translation” functions in order to render perceived inputs into the unambiguous, ontologically-grounded knowledge representation language (KRL) that is used to model their knowledge, memory and reasoning. This paper describes the KRL of OntoAgent with a special focus on the many runtime functions used to translate between perceived inputs and the KRL, as well as to manipulate KRL structures for reasoning and simulation.

Keywords: Knowledge representation; simulation; natural language processing; computational semantics.

1. Introduction

The ultimate goal of agent modeling is to contribute to attaining the original goal of research in artificial intelligence: developing automatic intelligent agent systems that will be able — individually and in teams consisting of other such agents and people — to perform tasks that, at present, can be performed only by people. Consider some of the core tasks of a simulated, embodied, language-enabled intelligent agent: behaving in a physiologically realistic manner; experiencing, interpreting and remembering its own mental and physical states; communicating with people in natural language; learning through experience, communication and reasoning; managing its memory, including forgetting, memory consolidation, etc.; making decisions to further the plans that will fulfill its goals; and collaborating with human and intelligent-agent colleagues. In order to minimize development effort, maximize resource reuse and avoid knowledge incompatibilities, all of these processes should be supported by an integrated knowledge substrate and a single knowledge representation language (KRL).

This paper describes the KRL that supports all of these processes in the OntoAgent environment. Special attention is devoted to the functions that translate various modes of agent perception into interpreted KRL structures and use KRL structures to support agent reasoning and action.

An Introductory Example. Before proceeding to an overview of OntoAgent and the specific KRL phenomena to be discussed, let us begin with an example that illustrates the OntoAgent KRL. The following is a representation of the meaning **Pirates hijacked a ship on March 3, 2009**. As will be described in more detail below, the elements of the meaning representation are language-independent ontological concepts that look like English words simply for the benefit of knowledge engineers. Each frame (here, HIJACK-1, PIRATE-1, SHIP-1) is headed by an OBJECT or EVENT instance, described using ontologically defined RELATIONS (such as AGENT) and ATTRIBUTES (such as ABSOLUTE-DAY). The indices are used to distinguish among instances since there could be, for example, more than one instance of hijacking reported in a language input or remembered in an agent’s memory.

HIJACK-1	
AGENT	PIRATE-1
THEME	SHIP-1
ABSOLUTE-MONTH	MARCH
ABSOLUTE-DAY	3
ABSOLUTE-YEAR	2009
PIRATE-1	
AGENT-OF	HIJACK-1
SHIP-1	
THEME-OF	HIJACK-1

The representation of this meaning will be the same (apart from some details of indexing) whether it is located in the agent’s persistent memory, whether it is derived from the semantic interpretation of a new textual input, whether it is derived from agent reasoning about other known facts, or — in a future implementation — whether it is derived from the agent’s interpretation of simulated vision. In short, all meaning that an OntoAgent agent stores and manipulates is rendered in the same, unambiguous, ontologically-grounded KRL, and it is the agent’s responsibility to translate into, translate out of, and manipulate structures in this KRL in order to carry out high-level reasoning and naturally interface with the outside world.

A Sample Application of OntoAgent. In the upcoming discussion of KRL-related phenomena, it will be useful to cite examples from an actual application configured in the OntoAgent environment. The application used for this purpose is Maryland Virtual Patient (MVP), a simulation and tutoring system developed to support training cognitive decision making in clinical medicine [9–11, 18–20]. MVP is implemented as a society of agents, with one role — that of the trainee — played by a human and other roles played by artificial intelligent agents. At the core of this

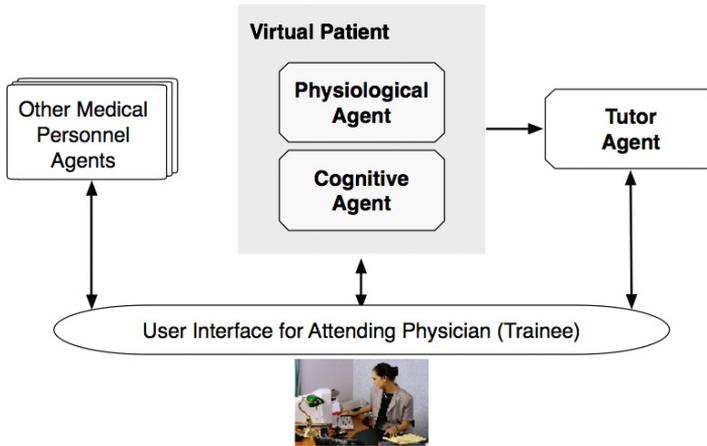


Fig. 1. The network of agents in MVP.

network is the virtual patient (VP) — a knowledge-based model and simulation of a person suffering from one or more diseases. The virtual patient is a “double agent” in that it models and simulates both the physiological and the cognitive functionality of a human. Physiologically, it undergoes both normal and pathological processes in response to internal and external stimuli. Cognitively, it is implemented as a collection of knowledge-based models of simulated human-like perception, reasoning and action processes. Figure 1 shows a high-level view of the agent network in MVP.

Human users of MVP can carry out all actions expected of a physician: interview the VP, teach it about its condition, suggest tests and treatments, try to convince the VP to agree to them if the VP is not immediately amenable, receive the results of tests, observe the results of treatments, follow a patient over a long period of simulated time, optionally receive help from an automatic mentor, and optionally rerun a given patient any number of times to practice different management strategies.

A Summary of the Content and Organization of the Paper. Figure 2 shows the architecture of an agent — its knowledge resources as well as its perception, reasoning and action capabilities. The figure provides a visual anchor for each of the eight KRL-oriented phenomena to be discussed in the subsections below. The short descriptions of each phenomenon point to respective nodes in the figure, with terms used in the latter indicated by italics.

- (1) The *OntoAgent* KRL is the unambiguous, formal metalanguage in which all of the agent’s semantically interpreted *Knowledge Resources* are stored: the ontology (the memory of types), the fact repository (the memory of tokens) and the semantic interpretation of words stored in the sem-struct zone of lexicon entries. We begin with an overview of these resources and a more detailed description of the KRL than was provided in the brief pirates example above.

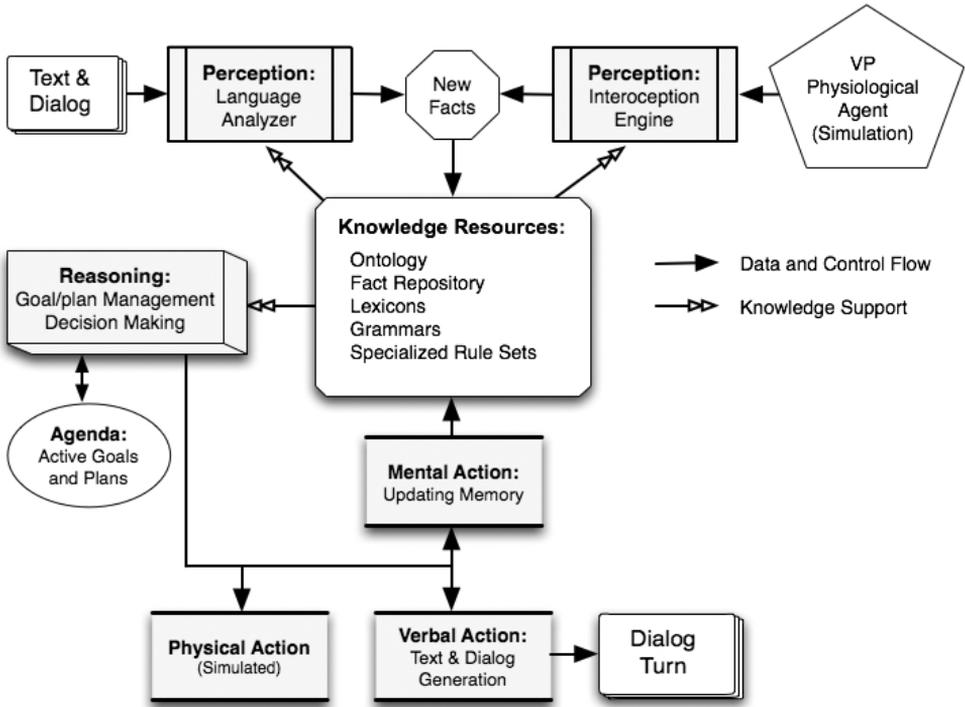


Fig. 2. The architecture of an agent.

- (2) When an agent receives natural language input (*Perception: Language Analyzer*), it interprets it, translating it into the unambiguous KRL before remembering it as *New Facts* in its persistent memory. This process of translation — which is not attempted by most agents and natural language processing systems — involves a large suite of NLP resources and processors.
- (3) Memory management (*Mental Action: Updating Memory*) involves linking new memories to existing memories, which can involve the interpretation of KRL paraphrases.
- (4) The simulation of the physiological side an agent — in our example, the virtual patient (*VP Physiological Agent Simulation*) — is carried out using ontologically recorded scripts written in the same KRL as all other OntoAgent knowledge.
- (5) When an agent experiences interoception (*Perception: Interoception Engine*), it must interpret what it is experiencing, which is modeled as translating from the KRL-encoded expert model that drives the simulation into a “naïve” KRL interpretation that is understandable by the agent.
- (6) Agents carry out plan- and goal-style decision making (*Reasoning: Goal/plan Management, Decision Making*), which is modeled using decision functions that take as input ontologically recorded property values expressed in the KRL.

- (7) As shown in (2), when agents receive natural language (NL) input, they must carry out $NL \rightarrow KRL$ translation in order to remember and subsequently employ the meaning of that input. Similarly, when interfacing with the world as language producers (*Verbal Action: Text & Dialog Generation*), they must carry out $KRL \rightarrow NL$ translation before generating a *Dialog Turn*.
- (8) When agents decide to take physical actions, they make the decision using decision functions in the KRL, then translate the KRL output of those decisions into simulation action (*Physical Action (Simulated)*).

To summarize, this paper describes all of these different manifestations of KRL in the OntoAgent environment and suggests that taking a unified approach to modeling such diverse agent functionalities will lead to significant benefits in modeling multi-functional agent systems that can be infinitely extended (we anticipate no ceiling of results associated with our approach) and ported across domains and applications.

2. KRL as Manifest in Knowledge Resources

The OntoAgent **ontology** is a formal model of the world encoded in an unambiguous KRL. A description of, and rationale for, the form and content of the ontology is available in [21], Sec. 7.1; an axiomatic definition is presented in Sec. 7.1.6 of that work; and additional theoretical and practical issues are discussed in [12], in preparation. Here, we present some highlights for purposes of orientation.

The ontology is organized as a multiple-inheritance hierarchical collection of frames headed by concepts that are named using language-independent labels. It currently contains approximately 9000 concepts, most of which belong to the general domain. The number of concepts in the ontology, currently around 8000, is far fewer than the number of words or phrases in any language for several reasons: (1) Synonyms and hyponyms are mapped to the same ontological concept, with semantic nuances recorded in the corresponding lexical entries. (2) Many lexical items are described using a combination of concepts. (3) Lexical items that represent a real or abstract point of a scale all point to a single property that represents that scale. (4) Concepts are intended to be cross-linguistically and cross-culturally relevant so we tend not to introduce concepts for notions like *recall* in the sense of a recalling a purchased good because it is highly unlikely that all languages/cultures use this concept. Instead, we describe the meaning of such words compositionally in the lexicons of those languages that do use it. (For further discussion of the lexicon/ontology split, see [13].)

Concepts divide up into events, objects and properties. Properties are primitives, which means that their meaning is grounded in the real world with no further ontological decomposition. The expressive power of the ontology is enhanced by multivalued fillers for properties, implemented using facets. Facets permit the ontology to include information such as “the most typical colors of a car are white, black, silver and gray; other normal, but less common, colors are red, blue, brown

and yellow; rare colors are gold and purple.” The inventory of facets includes: *default*, which represents the most restricted, highly typical subset of fillers; *sem*, which represents typical selectional restrictions; *relaxable-to*, which represents what is, in principle, possible but is not typical; and *value*, which represents not a constraint but an actual, non-overridable value. Select properties from the ontological frames for the event DRUG-DEALING illustrates the use of facets.

DRUG-DEALING		
IS-A	value	CRIMINAL-ACTIVITY
AGENT	default	CRIMINAL, DRUG-CARTEL
	sem	HUMAN
	relaxable-to	SOCIAL-OBJECT
THEME	default	ILLEGAL-DRUG
INSTRUMENT	sem	MONEY
HAS-EVENT-AS-PART	sem	BUY, SELL
LOCATION	default	CITY
	sem	PLACE
	relaxable-to	PHYSICAL-OBJECT
...		

Objects and events are defined using an average of 16 properties each, many of whose fillers are inherited rather than locally specified. In short, the *meaning* of an object or event is the set of its property-facet-value triples. Unlike most ontologies, the OntoAgent ontology includes complex events, otherwise known as scripts (cf. [29]), that support both simulation and reasoning about language and the world.

The main benefits of writing an ontology in a KRL rather than an NL are the absence of ambiguity in KRL and its potential reuse across all natural languages. Cut to twenty years from now, when the OntoAgent ontology should contain tens of thousands of well-described concepts, thousands of scripts, and reasoners that can leverage both of these to support the work of intelligent agents. Since the ontology is language independent, this knowledge infrastructure will be accessible to intelligent agents that communicate in any language, as long as a compatible lexicon for that language has been developed.

Since the ontology is language independent, its link to a natural language must be mediated by a **lexicon**. Semantically, each lexical sense specifies what concept, concepts, property or properties of concepts defined in the ontology must be instantiated in the text-meaning representation to account for the meaning of a given lexical unit of input. For example, the English lexicon indicates that the one sense of *dog* maps to the concept DOG (a type of CANINE); another sense maps to HUMAN, further specified to indicate a negative evaluative modality (e.g. a woman can call her cheating ex-boyfriend a dog); and yet another sense maps to the event PURSUE. Senses for argument-taking words and modifiers are encoded along with their typical syntactic configurations, such that word in the configuration is described both


```

value LECTURE-12
(and possibly hundreds or thousands of other events,
each of which will be described with all of the relevant
properties and values)
LOCATION-OF value TOWER-1 ; Tower of London
...

```

Every agent in an agent network can have its own ontology, lexicon and fact repository, all of which can be augmented during its simulated life through learning and experience.

3. Natural Language → KRL Translation During Language Processing

Although agent reasoning is most effectively carried out in a knowledge representation language (KRL), natural language (NL) is the ideal medium of communication with people. An intelligent agent with NL capabilities will not only communicate, it will also be able to convert vast amounts of knowledge available in NL into KRL, store it and use it for further reasoning about language and the world. In order to achieve this best of both worlds scenario — permitting intelligent agents to reason in KRL but having them communicate with people, and create knowledge bases from NL sources — it is important both to establish the formal relationship between NL and KRL and to provide intelligent agents with the facility to translate between them. An important issue in designing KRLs is how to make them as close as possible to NL, so as to facilitate the NL-KRL translation. Before describing NL → KRL translation in *OntoAgent*, let us first consider the bigger picture of the KRL vs. NL debate in AI-NLP.

The relationship between the fields of NLP and automatic reasoning has historically not been tightly interwoven. Core research on reasoning has not traditionally concentrated on NL issues. Conversely, though the kind of NLP that is required to build an intelligent agent must itself involve reasoning, the bulk of system building in NLP and computational linguistic research has concentrated on issues other than text meaning. Finally, reasoning in most systems that did treat meaning in NL used methods different from those employed for general reasoning. This division between NLP and reasoning was recognized already in the 1950s by Bar Hillel:

“...The evaluation of arguments presented in a natural language should have been one of the major worries of logic since its beginnings. However, [...] the actual development of formal logic took a different course. It seems that [...] the almost general attitude of all formal logicians was to regard such an evaluation process as a two-stage affair. In the first stage, the original language formulation had to be rephrased, without loss, in a normalized idiom, while in the second stage, these normalized formulations would be put through the grindstone of the formal logic evaluator. [...] Without

substantial progress in the first stage even the incredible progress made by mathematical logic in our time will not help us much in solving our total problem.” (quoted from [1], pp. 202–203).

Once one substitutes “knowledge representation language” (KRL) for “normalized idiom” and “reasoner” for “formal logic evaluator,” it becomes clear that the current state of affairs is quite similar to that of half a century ago.

The automatic creation, from NL text, of realistically broad-coverage knowledge bases that can adequately support reasoning has not become a central concern of NLP, in part, probably, because this task is AI-complete. Instead, most NLP work in the past 15 years or so has concentrated on so-called “knowledge-lean” methods, which aim at broad coverage but accept limited-quality results as long as some quantified progress can be demonstrated over time. Experimental accountability is undoubtedly a very desirable goal but there remains the issue of whether the currently dominant knowledge-lean methods are pursuing local maxima, and whether high-quality results in high-end applications — such as developing intelligent agents — are attainable by these methods even in principle. Our own work in ontological semantics [21] differs from most of the current work in NLP in that, while making use of stochastic techniques, it concentrates on knowledge-based methods in its pursuit of the long-term goal of building high-end applications that interleave NLP and reasoning. The translation between unconstrained NL and a KRL is a core area of this research.

Table 1 summarizes approaches and opinions about the relationship between NL and KRL encountered in the field. In *OntoAgent*, we take the NL-KRL-NL translation approach, which requires the most development effort but also promises, we believe, the biggest returns. The other approaches seek to obviate the need for this translation or at least to alleviate its complexity. We consider these other approaches in turn before describing our efforts to operationalize the NL-KRL-NL option.

“Pure” NL-as-KRL means using natural language directly as input to automatic reasoners. The NL-as-KRL movement among computational logicians and proponents of controlled languages is relatively recent and has been formulated as a research program: how to make KRL more NL-like. At least one AI/NLP researcher, Yorick Wilks, believes that KRL is already, in the final analysis, NL. Wilks has consistently argued (e.g. [33, 35] where this position is further claimed to ascend back to [36]) that any knowledge representations that are meaningful are by nature ambiguous and vague, and that it is, in principle, impossible to eliminate such “language-like features” as ambiguity from ontologies and conceptual structures. This position, clearly, sounds the death knell for standard automatic reasoning techniques because it essentially states that any automatic reasoning will be indeterminate.

At the same time, Wilks claims, and rightfully so, that ambiguous, incomplete and inconsistent knowledge resources can be and are still useful for NLP (he inevitably cites WordNet [15], a lexical resource widely used by corpus-based NLP

Table 1. Summary of approaches to and opinions about the relationship between NL and KRL.

Approach	Advantages	Difficulties
NL-KRL-NL translation	NL input/output readable by people. KRL readable by trained people. NL knowledge bases available. KRL suitable for reasoners. Multi-lingual: same KRL/ reasoners for all languages.	Needs expressive, NL-influenced KRL. Need large, high-quality KBs. Need high-quality analyzer. Need reasoners about language and the world.
“pure” NL-as-KRL	Easy to use for people. NL knowledge bases (e.g. encyclopedias, diction- aries, gazetteers, etc.) available.	Impractical: engines for reasoning directly in NL do not exist and may never exist.
Controlled NL-as-KRL	Controlled NL automatically trans- latable without loss into KRL. Practical in a subset of applications.	Impractical in the general case of deriving knowledge from text. Quality depends on lack of ambiguity. Difficult to impose at the lexical level (users have to become lexicon acquirers).
“pure” KRL-as-NL	Easy to use for reasoners.	Impractical: too heavy a burden on people to learn/use the KRL. KRL knowledge bases unavailable. NL knowledge bases unusable.
KRL-is-NL	Provides justification for modest goals of incremental, knowledge-lean R&D.	As a theoretical stance, does not allow for any constructive proposals about the NL-KRL connection.
NL-NL _c -KRL-NL _c	Controlled NL (NL _c) automatically translatable without loss into KRL. Practical in a subset of applications.	Requires manual translation into the controlled language.

practitioners, though demonstrably incomplete and flawed when used as a knowledge base for NLP [23]). The catch is that the types of applications Wilks has in mind rely only partially on addressing issues of text meaning. Take the example of a personal conversational assistant [34] which can, to a degree, fake understanding and, when needed, can change the subject to lead the conversation to a topic where its ability to hold up its end of the conversation is more secure. This is an admirable sleight-of-hand strategy that works in an application where the main purpose of communication is phatic. Strategically similar approaches can alleviate the complexity of automatic text meaning extraction and manipulation in situations when responsibility for understanding can be passed to the human interlocutor, as when interpreting noisy output from machine translation or summarization systems. However, such detour strategies will not work when understanding by the intelligent agent is crucial. We have commented in detail on the NL-is-KRL opinion in [22] and [17], and here will limit our remarks to a few relevant methodological points.

It is noteworthy that while Wilks cited the extraction and manipulation of text meaning as the major scientific objective, ambiguity of representation was not a central issue for him. The desire to nudge the evaluation results of systems like word sense disambiguation engines into the 90% range (cf. [6]) led him to claim that, for NLP, only word sense distinctions at the coarse-grained level of homographs are important.^a Such a claim may work in the world of Semeval and other similar competitions but, in reality, the situation with sense delimitation is much murkier. For example, the English word *operation* has 11 senses in the American Heritage dictionary and (so far) 3 senses in the OntoSem semantic lexicon (roughly, *military operation*, *surgery* and *general state of functioning*). Different structures/concepts correspond to these meanings in the metalanguage, each with its own set of properties and value sets. If this three-way ambiguity is retained in the representation, then, to gain more information about *the operation*, the reasoner will not know whether to ask “Was general anaesthesia administered?” or “Was a general in command?” It is, however, entirely possible that at any given time in the continuous process of knowledge acquisition for NLP/reasoning (that is, knowledge to support an intelligent agent) a subset of distinctions that are necessary to avoid confusing the reasoning engine have not yet been introduced. It follows that if certain distinctions are not required for any reasoning purposes, such “benign” ambiguities may be retained in the representation. This is clearly an operational, application-oriented approach but we have to live with it because the field has not yet come up with a universal theoretical criterion for sense delimitation.

Today it is reasonable to hope that the balance between short-term and long-term research in NLP and reasoning is on the road to being restored. Even in the empiricist research paradigm, currently dominant in NLP, researchers recognize that the core prerequisite for the improvement of their application systems (which currently achieve only modest results) is not developing better machine learning algorithms that operate on larger sets of training data, but rather enhancing the types of knowledge used in the processing (the terminology they prefer is judicious selection of distinguishing features on which to base the comparisons and classifications of texts). As [7] notes, “[i]n the context of language, doing ‘feature engineering’ is otherwise known as doing linguistics. . . [T]he space of interesting and useful features that one could extract is usually effectively unbounded. All one needs is enough linguistic insight and time to build those features. . .” These features are, in practice, elements of the metalanguage of representation of text meaning and, therefore, of KRL.

Controlled NL (NL_c) as KRL. The notion of controlled languages ascends at least to the “basic English” of Ogden (1930). These are languages with a restricted

^aThe psycholinguistic evidence that Wilks and Ide cite to support this position is irrelevant because systems do not operate the way people do. In fact, Wilks’ own famous “theorem” to the effect that there is no linguistic theory, however bizarre, that cannot be made the basis of a successful NLP system (Wilks actually said, MT system) seems to argue for discounting psycholinguistic evidence for NLP.

lexicon, syntax and semantics. Dozens of controlled languages, deriving from many natural languages, have been developed over the years to serve computer applications. A typical application is using a controlled language to write a document (e.g. a product manual) in order to facilitate automatic translation of the text into other languages; presumably, the controlled-language text will contain fewer lexical, grammatical, semantic and other ambiguities capable of causing translation systems to make mistakes. Controlled languages are usually discussed together with authoring tools of various kinds (spelling checkers, grammar checkers, terminology checkers, style checkers, etc.) that alleviate the difficulties that authors face in conforming to the controlled language in their writing. Controlled languages can also be used as programming languages: according to [30, 31], COBOL is a controlled English. The controlled languages especially relevant to this paper are computer processable controlled languages (CPCLs, [32]), modeled after [28] Computer Processable English. The defining constraint of a CPCL is “to be capable of being completely syntactically and semantically analyzed by a language processing system” [28]. Work in this area involves building tools to facilitate two aspects of the process: authoring texts in the controlled language and carrying out specialized types of knowledge acquisition — for example, compiling NL-NL_c dictionaries that provide a specification of the particular senses of NL words that are included in NL_c. One benefit of this approach is that such dictionaries can, if desired, be completely user-dependent, which means that different kinds of reasoning will be supported by the same general apparatus using these “idiolects” of NL_c.

A large number of CPCLs have been proposed in the last decade, among them the language used in the KANT/KANTOO MT project [24], Boeing’s Computer Processable Language [4], Processable English [27], the Controlled English to Logic Translation (CELT) language (e.g. [26]), SUMO [16], Common Logic Controlled English [30, 31] and Attempto [5]. While there are differences between them, strategically all of them conform to the methodology of using people, not machines, to disambiguate text. This disambiguated text can be represented in a variant of FOL — possibly with some extensions — and used as input to reasoning engines. To our knowledge, only CELT and KANT attempt to ground their symbols in an ontology. The other approaches keep their ontological commitment to a minimum. Having no ontological commitment broadens the opportunity for user-defined applications that can bypass the automatic analysis of open text. However, the research and development devoted to the use of controlled languages, in our opinion, contributes little to the long-term goal of creating truly automatic intelligent agents, which is predicated on the capability of understanding unconstrained language.

Pure KRL-as-NL refers to having human users interact with a system directly using a KRL, which — although in some senses ideal for system development and artificial agent functioning — puts such a high cognitive load on users as to make it infeasible in all but the most constrained of domains and tasks.

KRL-is-NL represents a philosophical view ultimately ascending to [36]. It maintains, roughly, that the symbols used in KRLs are ultimately taken from NLs and cannot be taken from anywhere else; as a result, KRLs retain NL-like features, such as ambiguity and vagueness, no matter how carefully researchers may design them. In practice, this opinion can be used to justify knowledge-lean NLP and reasoning on the grounds of the philosophical impossibility of attaining better results.

NL-NL_c-KRL-NL_c is a multi-step translation method favored by some practitioners of approaches based on first-order logic (FOL). The goal is to constrain NL to whatever can be automatically translated into FOL. So, the process involves human translation from NL to a controlled NL (NL_c), for which a parser into KRL is available. As such, having a controlled NL and its associated parser becomes equivalent to having a KRL (e.g., [5, 26, 32, 8, 28]). It is assumed that system output in a controlled language will pose no problems for people.

NL-KRL-NL Translation. As mentioned above, in OntoAgent we believe that NL-KRL-NL translation is mandatory for sophisticated, reasoning-oriented applications. The NL → KRL translation is carried out by the OntoSem text analyzer, which is a semantically-oriented text analysis system that has been in development for over 20 years [21, 2, 3; and others]. The results of text processing are text meaning representations (TMRs), which are disambiguated representations of text meaning written in the same knowledge representation language that is used in the ontology, the sem-struct of the lexicon and the fact repository. (See [21], Chapter 6, for motivation for the structure and content of TMRs; see [12] for practical details and examples.) TMRs represent propositions connected by discourse relations. Propositions are headed by instances of ontological concepts, parametrized for modality, aspect, proposition time, overall TMR time, and style. Each proposition is related to other instantiated concepts using ontologically defined relations (which include case roles and many others) and attributes. Coreference links form an additional layer of linking between instantiated concepts. Figure 3 shows the architecture of the OntoAgent text analysis system.

If the analyzer receives the input *Charlie watched the baseball game*, it will use disambiguation techniques to determine that the first verbal sense of *watch* (watch-v1) should be selected.

```
(watch-v1
  (def "voluntary visual event")(ex "John was watching a movie.")
  (syn-struct
    ((subject ((root $var1) (cat n)))
     (root $var0) (cat v)
     (directobject ((root $var2) (cat n) (opt +))))))
  (sem-struct
    (VOLUNTARY-VISUAL-EVENT(AGENT (value ^$var1)) (THEME (value ^$var2))))))
```

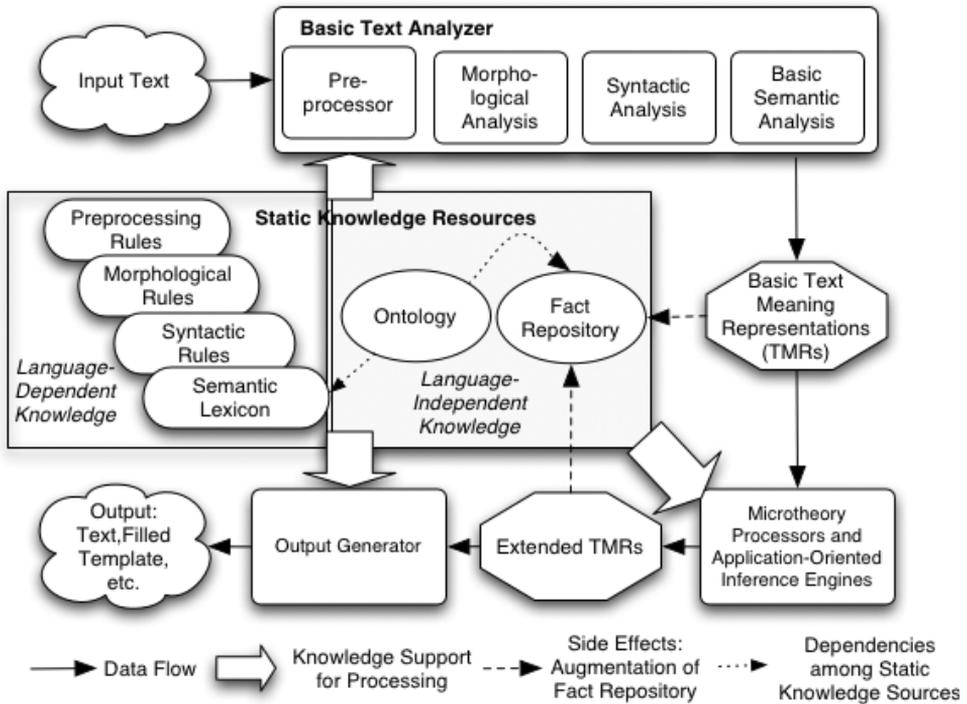


Fig. 3. The OntoAgent text analysis system.

It will then assign the semantic analysis of *Charlie* to \$var1 and the semantic analysis of *the baseball game* to \$var2, leading to the output TMR shown below. Comments indicate the word in the text that instantiates each concept as well as the lexical senses selected.

```

VOLUNTARY-VISUAL-EVENT-1                ; "watched"; uses watch-v1
AGENT                                     HUMAN-1
THEME                                     BASEBALL-GAME-1
TIME                                     (< find-anchor-time) ; indicates past tense
HUMAN-1                                  ; "Charlie"; uses
                                           *personal-name* processing

AGENT-OF                                  VOLUNTARY-VISUAL-EVENT-1
HAS-PERSONAL-NAME                        "Charlie"
BASEBALL-GAME-1                          ; "baseball game"
                                           ; uses baseball_game-n1

THEME-OF                                  VOLUNTARY-VISUAL-EVENT-1
  
```

It can be seen that the presentation format of TMRs, shown above, reads rather easily and is quite natural-language like. In addition, the OntoAgent environment

has a DEKADE toolset that supports graphical viewing of TMRs as well as pre-final stages of text analysis (<http://www.trulysmartagents.org/dekade.php>).

4. Updating Memory: Paraphrase Interpretation

In order for language-endowed agents to operate intelligently — as when answering questions posed by a human user or learning new facts — they must be able to **interpret** language input, **remember** the content of that input, and attempt to **match/link** that content with memories already stored in their fact repository. Linking new information to old memories is a standing goal of all intelligent agents, and a core capability enabling such linking is the recognition and resolution of paraphrase.

Having generated a meaning representation (MR) from input — be that input language, experience, etc. — the intelligent agent must consider at least the following potential eventualities in deciding on how to remember the content of this input. (1) The new MR is identical to a stored memory. (2) The new MR is identical to a stored memory except for metadata values: the identity of the speaker, the time stamp, etc. (3) The new MR contains a subset or a superset of properties of a stored memory. (4) The new MR is similar to a stored memory but one or more properties has a different value. (5) The new MR — or a component of it — is related to a stored memory via ontological subsumption, meronymy or location. (6) The new MR is related to a stored memory as the latter’s precondition or effect. (7) The new MR is related to a stored memory via “ontological paraphrase.” (8) The new MR is not related to any stored memory because different concepts are used, there are conflicting property values, etc. The first two eventualities, which are simple to detect, represent confirmations of existing memories; the rest require reasoning to determine whether or not the memories match. [14] presents algorithms for analyzing correspondences of types (5)–(7): e.g. it explains how a virtual patient in the MVP environment can match the question *Do you have any discomfort in your esophagus?* with a memory that is stored in the fact repository as (SYMPTOM-1 (LOCATION CHEST-1) (EXPERIENCER HUMAN-1)), which can be glossed as “the person has a symptom in his chest”.

The main point to understand is that using a KRL does not free us from issues of paraphrase because there are so many different ways of *thinking about* a given phenomenon and carving out the world, and these can be realized by different KRL structures. As in our example above, one person can talk about “discomfort in the esophagus”, which is realized in KRL as DISCOMFORT (LOCATION ESOPHAGUS) and another person can refer to “a symptom in the chest”, realized in KRL as SYMPTOM (LOCATION CHEST) and they can actually be talking about the same thing. This is not a flaw in the KRL; it is a reflection of the complexity of the world and the way we choose to think and talk about it.

5. KRL-Recorded Scripts to Support Physiological Simulation

The physiological side of the agents in the OntoSem environment is modeled as a set of interconnected ontological objects representing human anatomy. Each object is

described by a set of ontological properties and their associated value sets. Crucial among the properties are those that link the objects to typical events in which they participate. These events are usually complex — that is, they include other, possibly also complex, events as their components. Following [29], we call these complex events scripts, and we encode them in the ontology using the property HAS-EVENT-AS-PART.

Physiological modeling covers both normal and pathological processes. As an example, let us consider disease models used in MVP, which are ontologically encoded cognitive models of diseases that reflect the mental models of practicing physicians. Disease models break down into two major classes based on whether or not the physiological causal chains underlying the disease are well understood. In cases in which physiological causal chains are relatively poorly understood, the simulation is primarily driven by temporal causal chains. Each disease is divided into conceptual stages, with each stage being associated with clinically observed physiological changes and symptom profiles. As simulated time passes, the patient’s state changes incrementally, calculated using an interpolation function that incorporates the start value of each property at the beginning of the disease and the end value for each conceptual stage. The other class of diseases modeled in the system are those for which physiological causal chains are quite well understood. Such diseases are particularly interesting because they can be spontaneously generated or cured based on the physiological preconditions being fulfilled. For example (and to simplify the state of affairs for exposition), if a person’s lower esophageal sphincter (the sphincter between the esophagus and the stomach) is hypotensive (too loose), he will develop gastroesophageal reflux disease, GERD. So if a user wants to give a healthy virtual patient GERD, all he has to do is carry out a surgical procedure to cut the lower esophageal sphincter and render it hypotensive.

Returning to issues of KRL, all of these simulation-supporting scripts are written in the same ontological metalanguage as the rest of the ontology. (See [9, 10] for more in-depth descriptions of the physiological models.) Furthermore, they are not only used to support simulation but are also used to support the reasoning virtual tutors and advisors. For example, if a physician-in-training using the MVP system wants to know what events occur during the progression of a disease, what paths the disease can take, etc., the tutor has the answer to these questions because the tutor’s ontological knowledge is the same expert knowledge that drives the simulation. Implementing this question-answering capability in MVP is on our agenda.

6. PhysiologicalAgent → CognitiveAgent Paraphrase During Interoception

Interoception is the perception of physiological phenomena. It is a feature of intelligent agents that have both physiological and cognitive aspects. The source of interoception is physiological phenomena, like symptoms of a disease, hunger and sleepiness. Memories of interoceptive experiences are stored using the same ontologically grounded metalanguage as memories gleaned from language processing

(recall that these are the two channels of perception for our agents at this time). And there is another key similarity between the knowledge gleaned from language perception and interoception: in both cases, paraphrase processing can be required to link the new KRL structures to agent memories. Section 4 illustrated this for language perception, we now briefly review it with respect to interoception.

Physiological simulations are driven by ontological scripts that record expert models. As such, they employ such concepts as *DYSPHAGIA* (difficulty swallowing), *PERISTALSIS* (wave-like contractions of a series of muscles), and *BOLUS* (the contents of a single swallow — a chewed piece of food or a gulp of liquid). When a non-physician agent experiences something like “difficulty swallowing”, or it swallows “some food”, or the food “doesn’t go down right”, it is certainly not remembering these things using concepts like *DYSPHAGIA*, *BOLUS* and *PERISTALSIS*. Instead, it uses whatever concepts it has available in its ontology. As a modeling strategy aimed at creating verisimilitude in our agents, we automatically translate from the expert model to a form interpretable by the agent itself using a method we call **Physiological-Agent → CognitiveAgent Paraphrase**.

Let us consider the example of *DYSPHAGIA* in a bit more detail. Whereas the *PhysiologicalAgent* that drives the simulation tracks the *INTENSITY* of the concept *DYSPHAGIA* over time, the *CognitiveAgent* most likely does not know about such a concept, and might not even recognize difficulty swallowing as a differentiated symptom at all, it might perceive it as a vague discomfort after swallowing. Therefore, when the *PhysiologicalAgent* generates a symptom, the *CognitiveAgent* must interpret it using ontological primitives that it understands. The translation from the *PhysiologicalAgent*’s ontological representation to the *CognitiveAgent*’s ontological representation can yield different paraphrases for different patients, though we expect each patient to use just one paraphrase consistently when remembering a given type of symptom. The types of knowledge that are used to automatically generate “agent-interpretable” paraphrases are ontological properties like subsumption and meronymy. For example, *DYSPHAGIA* is a child of *SYMPTOM* in the expert ontology; so when the *PhysiologicalAgent* generates an instance of *DYSPHAGIA*, the *CognitiveAgent* can interpret it using the parent concept, *SYMPTOM*, along with any of a number of different property-value pairs, such as [CAUSED-BY SWALLOW]. We must emphasize that this type of paraphrase represents a modeling strategy deriving from our symbolic approach to generating cognitive simulations; we are not suggesting that humans require such a facility. [18] further describes the automatic process of paraphrasing for agent interoception.

7. KRL-Recorded Knowledge to Support Plan- and Goal-Style Decision Making

All agents in the *OntoAgent* environment carry out dynamic decision making that in style, albeit not complexity, approximates human decision making. For example, whenever a decision needs to be made, an agent first determines whether it has sufficient information to make the decision, an assessment that is based on a

combination of what it actually knows, what it believes to be necessary for making a good decision, and its personality traits. If it lacks some decision-making knowledge, it can posit the goal of obtaining this knowledge, which is a metacognitive behavior that leads to learning (see [26] for more on metacognition in the OntoAgent environment). Formally speaking, a goal is an ontological instance of a property whose domain and range are specified.

Let us consider the fundamentals of planning and decision-making using the specific example of a virtual patient (VP) in the MVP application. The main goal pursued by all VPs is: (BE-HEALTHY-1 (DOMAIN HUMAN-1) (RANGE 1)). This means that for the ontologically defined property BE-HEALTHY applied to the agent itself (HUMAN-1), the range is the highest possible value — here, 1 on the abstract scale $\{0, 1\}$. We assume that this is a universal goal of all humans and, in cases in which it seems that a person is not fulfilling this goal, he is simply prioritizing another goal, like EXPERIENCE-PLEASURE.

In MVP, when a VP begins to detect symptoms, the goal instance BE-HEALTHY-1 is put on the goal and plan agenda. It remains on the agenda and is reevaluated when: (a) its intensity or frequency (depending on the symptom) reaches a certain level; (b) a new symptom arises; or (c) a certain amount of time has passed since the patient’s last evaluation of its current state of health, given that the patient has an ongoing or recurring symptom or set of symptoms: e.g. “I’ve had this mild symptom for too long, I should see the doctor.” At each evaluation of its state of health, the VP can either do nothing or go to see the doctor — a decision that is made based on an inventory of VP character traits, the current and recent disease state and, if applicable, previous doctor’s orders (cf. next section). If it decides to see the doctor (ontologically, SEE-MD), that plan is put on the agenda. All subgoals toward achieving the goal BE-HEALTHY-1 and their associated plans are put on and taken off the agenda based on VP decision functions that are triggered by changes in its physical and mental states throughout the simulation. So when the doctor suggests having a test (goal: HAVE-DIAGNOSIS-1) and the patient agrees, having the test (a plan toward the above goal) is put on the agenda; and so on.

Now let us shift to decision functions as such. Say the medical trainee suggests that the VP have the endoscopic procedure called pneumatic dilation, which the VP has never heard of. If the VP is very trusting it might agree to the procedure then ask what it is, or it might agree to it and wait to see if the doctor provides any information about it. If, by contrast, the VP is not very trusting, or if it is trusting but just very curious, it might ask questions before agreeing. The function that determines its behavior relies on the ontologically grounded features of the VP recorded in its profile, some of which are persistent (e.g. trust) and some of which are situation-dependent (e.g. anxiety). Several decision-making functions of VPs are detailed in [18].

8. KRL \rightarrow NL Translation for Language Generation

The input to the language generator in OntoAgent is content specification expressed in the KRL. Among the sources of content are decision-making, such as the decision

to ask a question before agreeing to a medical procedure, and other mental actions, such as searching memory for the answer to a question. The format of these KRL structures are identical to the TMRs discussed above. Currently, we use a template-style NL generator, as language generation has not been a recent focus of development.

9. KRL \rightarrow Simulation for Simulated Physical Action

The simulated physical action referred to in this section is agentic, non-verbal action — for example, an agent makes a doctor’s appointment, takes its medicine, drinks coffee, and so on. These actions are instantiated as goal-directed plans, and they are triggered by decision functions of the type described in Sec. 7. For example, when a VP’s symptoms persist for long enough, it puts an instance of the plan SEE-MD-1 on its agenda, and that plan stays there until its effects are posted — i.e. the action has been carried out. These actions are remembered in the same way as verbal input, interoception, and the results of agent reasoning, and they are subject to the same types of memory management (memory consolidation, forgetting, etc.) as memories from those sources.

10. Final Thoughts

The view of the nature and role of KRL promulgated and tested in *OntoAgent* can be summarized as follows. The KRL for the support of communicating intelligent agents is symbolic and serves the needs of describing world models (ontologies) as well as meanings of elements of communication (texts, dialog turns, etc.). The lexis of this language is drawn from an ontology and the syntax follows a set of format conventions that can, in fact, change between applications. The KRL also provides the formalism for encoding the outputs of the various modules of natural language analysis systems and inputs to a subset of modules of language generation systems. Text meaning representations consist of instances of ontological concepts and may be stored in the agent’s fact repository. The fact repository, together with the ontology and other knowledge resources, make up a model of the agent’s memory. New facts to be stored in the fact repository may be created through the operation of the perception modules available to the agent (in addition to text understanding, perception may cover visual, tactile, non-language auditory, olfactory and interoceptive signals). They may also be generated as a result of the agent’s mental actions — making and remembering new knowledge on the basis of the current content of its memory.

The salient point here is that facts in the fact repository are recorded using the same KRL. In general, the applicability of the same KRL for various perception, reasoning and action tasks is a worthy desideratum. Not only does such a KRL obviate the need for multiple intellectually boring format conversion tasks, it also significantly simplifies the learning curve for knowledge acquirers, system builders,

evaluators and end users. Though this latter consideration is not theoretical, it is crucially important for system builders.

References

- [1] Y. Bar Hillel, *Aspects of Language* (Magnes, Jerusalem, 1970).
- [2] S. Beale, S. Nirenburg and K. Mahesh, Semantic analysis in the Mikrokosmos machine translation project, in *Proceedings of Symposium on Natural Language Processing*, Bangkok, Thailand, 1995.
- [3] S. Beale, B. Lavoie, M. McShane, S. Nirenburg and T. Korelsky, Question answering using ontological semantics, in *Proceedings of ACL-2004 Workshop on Text Meaning and Interpretation*, Barcelona, Spain, 2004.
- [4] P. Clark, P. Harrison, T. Jenkins, J. Thompson and R. H. Wojcik, Acquiring and using world knowledge using a restricted subset of English, in *Proceedings of FLAIRS-2005*, 2005.
- [5] N. E. Fuchs, K. Kaljurand and G. Schneider, Attempto controlled English meets the challenges of knowledge representation, reasoning, interoperability and user interfaces, in *Proceedings of FLAIRS-2006*, 2006.
- [6] N. Ide and Y. Wilks, Making sense about sense, in Agirre and Edmonds (eds.), *Word Sense Disambiguation* (Kluwer, Dordrecht, 2005).
- [7] C. Manning, Language learning: Beyond Thunderdome, in *Proceedings of CoNLL-04*, 2004.
- [8] D. McAllester and R. Givan, Natural language syntax and first-order inference, *Artificial Intelligence* **56** (1992) 1–20.
- [9] M. McShane, G. Fantry, S. Beale, S. Nirenburg and B. Jarrell, Disease interaction in cognitive simulations for medical training, in *Proceedings of MODSIM World Conference*, Medical Track, 2007, Virginia Beach, Sept. 11–13, 2007.
- [10] M. McShane, S. Nirenburg, S. Beale, B. Jarrell and G. Fantry, Knowledge-based modeling and simulation of diseases with highly differentiated clinical manifestations, in *Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME '07)*, Amsterdam, The Netherlands, July 7–11, 2007.
- [11] M. McShane, B. Jarrell, G. Fantry, S. Nirenburg, S. Beale and B. Johnson, Revealing the conceptual substrate of biomedical cognitive models to the wider community, in *Proceedings of Medicine Meets Virtual Reality 16*, eds. J. D. Westwood, R. S. Haluck, H. M. Hoffman, G. T. Mogel, R. Phillips, R. A. Robb, K. G. Vosburgh, 2008, pp. 281–286.
- [12] M. McShane, S. Nirenburg and S. Beale, *Meaning-Centric Language Processing*, in preparation.
- [13] M. McShane, S. Nirenburg and S. Beale, An NLP lexicon as a largely language independent resource, *Machine Translation* **19**(2) (2005) 139–173.
- [14] M. McShane, S. Nirenburg and S. Beale, Resolving paraphrases to support modeling language perception in an intelligent agent, in *Proceedings of the Symposium on Semantics in Systems for Text Processing (STEP 2008)*, Venice, Italy, 2008.
- [15] G. Miller, Wordnet: A lexical database for English, *Communications of ACM* **38** (1995).
- [16] I. Niles, and A. Pease, Towards a standard upper ontology, in *Proceedings of Formal Ontology in Information Systems (FOIS 2001)*, Ogunquit, Maine, USA, 2001.
- [17] S. Nirenburg, The Maryland Virtual Patient as a task-oriented conversational companion, in Yorick Wilks (ed.), *Close Engagements with Artificial Companions* (John Benjamins, 2010).

- [18] S. Nirenburg, M. McShane and S. Beale, A simulated physiological/cognitive “double agent”, in *Proceedings of the AAAI 2008 Fall Symposium on Biologically Inspired Cognitive Architectures*, 2008.
- [19] S. Nirenburg, M. McShane and S. Beale, A unified ontological-semantic substrate for physiological simulation and cognitive modeling, in *Proceedings of the International Conference on Biomedical Ontology*, University at Buffalo, NY, July 24–26, 2009.
- [20] S. Nirenburg, M. McShane and S. Beale, Aspects of metacognitive self-awareness in Maryland Virtual Patient, in *Proceedings of the AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems*, Arlington, VA, Nov. 11–13, 2010.
- [21] S. Nirenburg and V. Raskin, *Ontological Semantics* (MIT Press, Cambridge, MA, 2004).
- [22] S. Nirenburg and Y. Wilks, What’s in a symbol: Ontology, representation and language, *Journal of Experimental and Theoretical Artificial Intelligence* **13**(1) (2001) 9–23.
- [23] S. Nirenburg, M. McShane and S. Beale, The rationale for building resources expressly for NLP, in *Proceedings of LREC-2004*, 2004.
- [24] E. Nyberg and T. Mitamura, Controlled language and knowledge-based machine translation: Principles and practice, in *Proceedings of the First International Workshop on Controlled Language Applications (CLAW '96)*, 1996.
- [25] C. K. Ogden, *Basic English: A General Introduction with Rules and Grammar*, 1930.
- [26] A. Pease and W. Murray, An English to logic translator for ontology-based knowledge representation languages, in *Proceedings of the 2003 IEEE International Conference on Natural Language Processing and Knowledge Engineering*, Beijing, China, 2003.
- [27] PENG. PENG website: <http://www.ics.mq.edu.au/~rolfs/peng/>, 2006.
- [28] S. Pulman, Controlled language for knowledge representation, in *Proceedings of the First International Workshop on Controlled Language Applications*, 1996.
- [29] R. C. Schank and R. P. Abelson, *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures* (Lawrence Erlbaum, Hillsdale, NJ, 1977).
- [30] J. F. Sowa, Controlled English, <http://www.jfsowa.com/logic/ace.htm>, 2004.
- [31] J. F. Sowa, Common Logic Controlled English, Technical report, draft, 24 February 2004.
- [32] J. Z. Sukkarieh, Mind your language! Controlled language for inference purposes, in *Proceedings of the Joint Conference Combining the 8th International Workshop of the European Association for Machine Translation and the 4th Controlled Language Applications Workshop*, Dublin, Ireland, May 2003.
- [33] Y. Wilks, Reply to McDermott, *SIGART Bulletin*, No. 53, 1976.
- [34] Y. Wilks, Artificial companions, *International Scientific Reviews*, 2004.
- [35] Y. A. Wilks, Form and content in semantics, in *Computational Linguistics and Formal Semantics*, M. Rosner and R. Johnson (eds.). (Cambridge University Press, 1992), pp. 257–281.
- [36] L. Wittgenstein, *Philosophical Investigations* (Blackwells, Oxford, 1953).

Copyright of International Journal of Semantic Computing is the property of World Scientific Publishing Company and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.