

Bidirectional mapping between OWL DL and Attempto Controlled English

Kaarel Kaljurand, Norbert E. Fuchs

[Attempto project](#)

University of Zurich

Introduction

- OWL DL is a complex and expressive language
- Current OWL DL front-ends assume knowledge of Description Logics (DL)
- Attempto Controlled English (ACE) is an expressive and well-studied controlled natural language
- ACE has a parser that converts ACE texts into Discourse Representation Structures (DRS)
- We propose to use ACE as a front-end to OWL DL, concretely to use DRS as interlingua between ACE and OWL DL

Introduction to ACE

- Subset of English
- Controls ambiguity
 - A man is a human. (ambiguous)
 - Every man is a human.
 - There is a man who is a human.
- Controls synonymy
 - Every man is a human.
 - If there is a man then he is a human.
- Easy to learn

Text-based OWL editor

- Vision: A primarily text-based system would convert ACE structures into OWL DL and translate the modifications done by external human and machine reasoners back to ACE
- Such a front-end would hide the math behind OWL DL and will thus be natural to use by everybody

What needs to be done

- Show how all the meanings which can be expressed in OWL DL can also be expressed in ACE
- Show how this mapping can be easily explained to average users
- Show how all OWL DL ontologies can be translated to ACE
- Extend ACE
- Extend the mapping to target more expressive languages (e.g. OWL 1.1, SWRL?)

Attempto DRS

[A, B, C, D, E, F]

```
object(C, atomic, man, person, cardinality, count_unit, eq, 1)-1
object(D, atomic, named_entity, person, cardinality, count_unit, eq, 1)-2
object(A, atomic, named_entity, person, cardinality, count_unit, eq, 1)-1
named(A, Bill)-1
named(D, William)-2
predicate(B, unspecified, like, A, A)-1
predicate(E, state, be, A, C)-1
predicate(F, state, be, A, D)-2
```

[G, H, I]

```
object(H, atomic,
businessman, person,
cardinality, count_unit, eq,
1)-3
object(G, group, thing,
object, cardinality,
count_unit, geq, 3)-3
predicate(I, unspecified,
own, H, G)-3
```

⇒

[]

[J, K]

```
object(J, atomic, man,
person, cardinality,
count_unit, eq, 1)-3
property(J, self_made)-3
predicate(K, state, be, H,
J)-3
```

∨

[L, M, N]

```
object(L, atomic,
programmer, person,
cardinality, count_unit, eq,
1)-3
predicate(M, unspecified,
know, L, A)-3
predicate(N, unspecified,
employ, H, L)-3
```

Bill who is a man likes himself. Bill is William. Every businessman who owns at least 3 things is a self-made man or employs a programmer who knows Bill.

Mapping from ACE to OWL

- Express in ACE the OWL distinction of Individual, Class and Property
 - Individuals: object/8 (i.e. nouns and propernames) in the toplevel DRS
 - Classes: object/8 in the if-then box
 - Properties: predicate/5 (transitive verbs/adjectives)
 - Class hierarchy: if-then box
 - Property descriptions: if-then box

Mapping from ACE to OWL

- Special meaning of copula 'be', depending on the context
 - John is Bill. (*sameAs*)
 - Every man is a human. (*subclassOf*)
- *owl:Thing*: indefinite pronouns (*something, everything, nothing*)
- Cardinality: generalized quantifiers (*at most, less than, ...*) with the plural of *thing*
- *intersectionOf*, *unionOf*, *complementOf*: *and, or, not* (and other ACE constructs which create the same meaning)
- Superproperties, *inverseOf*, *transitivity*: generally require if-then constructs

Our DRS in OWL DL

Bill who is a man likes himself. Bill is William.

Every businessman who owns at least 3 things
is a self-made man or employs a programmer who knows Bill.

$\text{bill} \in \top$

$\text{m1} \in \text{Man}$

$\text{william} \in \top$

$\text{bill} = \text{m1}$

$\text{bill} = \text{william}$

$\text{likes}(\text{bill}, \text{bill})$

$(\text{Businessman} \sqcap \geq 3 \text{ owns}) \sqsubseteq$

$(\text{SelfmadeMan} \sqcup (\exists \text{ employs } (\text{Programmer} \sqcap$
 $(\exists \text{ knows } \{\text{bill}\}))))$

Properties

- Properties are (mostly) described by rule-like constructs using explicit variables
- In ACE, we avoid keywords like *transitive*, *inverse functional*, etc
- subPropertyOf
 - Everybody who loves somebody likes him/her.
- transitivity
 - If something A is taller than something B and B is taller than something C then A is taller than C.
- inverseOf
 - If something A is taller than something B then B is shorter than A. If something A is shorter than something B then B is taller than A.

allValuesFrom

- ACE could be extended to better express *allValuesFrom*.
- E.g. Carnivore $\sqsubseteq \forall$ eats Meat
- Using *only* or *nothing but* is not yet supported
 - * Every carnivore eats only a meat.
- Current solution is not so natural...
 - No carnivore eats something that is not a meat.
 - Everything a carnivore eats is a meat.

Currently missing features

- No support for enumerations (*oneOf*). Do we need noun phrase disjunction which is currently lacking in ACE?
 - Every land is either England or Montenegro or ...
- No support for datatype properties
 - John → age = xsd:integer 30
- Things that make OWL DL a Semantic Web language, such as URI, owl:import, annotation properties and versioning, don't fit well in ACE

Explaining OWL ACE

- OWL ACE is a subset of ACE which can be mapped to OWL DL.
- Is this subset easier or harder to explain to the users?
- Some restrictions are easy...
 - No queries
 - No plurals apart from a restricted use of 'things'
 - No intransitive and ditransitive verbs, no modifiers such as adverbs, prepositional phrases, and adjectives

Explaining OWL ACE

- ... some are trickier, e.g. use only every-sentences without any anaphoric references, unless they point to a propername or a previously declared individual
 - Every man sees John.
 - * Every man sees himself.
 - * Every man sees a dog and sees a cat that sees the dog.
- Some of the restrictions should go away once we target a more expressive language

From OWL DL to ACE

- OWL-to-ACE must handle all OWL constructs, some of which the ACE-to-OWL does not produce
- Problem: Naming conventions used for OWL classes and properties
 - class names: SpicyPizza, MotherWith3Children
 - property names: accountName, brotherOf, isWrittenBy
- OWL ACE prefers classes to be named by singular nouns and properties by transitive verbs/adjectives
- We have a general verbalizer which converts DRSs into Core ACE but OWL ACE could need something more specific, which e.g. would use every-sentences as much as possible

Last slide

- Implementation
 - <http://www.ifi.unizh.ch/attempto/tools>
- Thanks for listening
- Questions?