17th ICCRTS
Operationalizing C2 Agility

Title of Paper:

Information Extraction Using Controlled English to Support Knowledge-Sharing and Decision-Making

Topics:

Data, Information and Knowledge
Collaboration, Shared Awareness, and Decision Making

Name of Authors:

Ping Xue; Stephen Poteet; Anne Kao
Research & Technology, The Boeing Company
P.O. Box 3707 MC 7L-43
Seattle, WA 98124-2207. USA
Email: ping.xue@boeing.com; stephen.r.poteet@boeing.com; anne.kao@boeing.com

David Mott; Dave Braines
Emerging Technology Services, IBM United Kingdom Ltd.
Hursley Park, Winchester, SO21 2JN, UK
Email: MOTTD@uk.ibm.com; dave_braines@uk.ibm.com

Cheryl Giammanco
US Army Research Laboratory
Human Research & Engineering Directorate
ATTN: RDRL-HRS-E
459 Mulberry Point Road
Aberdeen Proving Ground, MD 21005-5425
Email: cheryl.a.giammanco.civ@mail.mil

Tien Pham
US Army Research Laboratory
Sensors and Electron Devices Directorate
ATTN: RDRL-SES-A
2800 Powder Mill Road
Adelphi, MD 20783-1193
Email: tien.pham1.civ@mail.mi

Point of Contact:

Ping Xue
ping.xue@boeing.com
(425) 373-2861

Abstract

Current and future coalition operations involve multi-team and/or multi-nation collaborations. While large volumes of structured/unstructured data are often available, improvement of data access, information extraction, and knowledge sharing is critically important but remains a major challenge for effective and efficient C2 operations. In this paper, we propose an approach to information extraction using International Technology Alliance Controlled English (CE) to improve fact extraction and knowledge sharing, aiming to enhance situation awareness and support decision-making. CE is a subset of English with a restricted grammar to reduce complexity and avoid ambiguity. The current version of CE has a formal syntax and semantics and is consistent with First Order Predicate Logic. CE is used to model both the inputs and outputs of the information extraction process, and to support end-users in configuring information extraction tools. Thus, CE provides, among other things:

(i) A user-friendly language for queries and system-to-user report representation.
(ii) A common form of expression that supports extending and modifying domain models (ontologies), and enables mapping between models and terminology or language variants.

CE-based information extraction will greatly facilitate the processes in the cognitive and social domains that enable forces with diverse backgrounds to collaborate effectively and efficiently.

Keywords: Coalition operations, multi-nation collaborations, structured/unstructured data, data access, information extraction, knowledge sharing, controlled natural language, Controlled English , situation awareness, decision-making, ambiguity, formal syntax and semantics, First Order Predicate Logic.

1. Background and Needs

The U.S. and the UK have established a collaborative research alliance called *International Technology Alliance* (ITA)[1] to address coalition related problems in Network Enabled Operations. One of the major problems addressed by the ITA is the inability of coalition forces to share data and information at the “edge” of the network. During coalition operations such as partner capacity building (e.g., in Afghanistan), it is critically important for Soldiers from different nations to share and exploit information from data repositories gathered by organizations that may not share a common mission.

[1] In 2006, the US Army Research Laboratory (ARL) and the UK Ministry of Defence (MoD) established a collaborative research alliance with academia and industry partners called the *International Technology Alliance* (ITA) to address fundamental issues in *Network and Information Sciences* to enhance the abilities of the US and UK to conduct coalition operations. The ITA is a unique UK-US collaborative venture. It is a multi-disciplinary research program that focuses on coalition needs and seeks to develop a mutual understanding and strong US-UK partnerships among the government, academia and industry participants.

The information requirements for organizations are mission specific and the ability to analyze this information depends on the operational tempo. Soldiers have access to vast amounts of information in short periods of time due to advances in information processing, yet they lack the analytics to provide them with context relevant information for their domain model and workflow. The purpose of the research reported in this paper is to provide organizations and individuals with the capability to perform cross-domain queries using "human friendly" analytics to share and exploit information for collaboration and decision-making.

In this paper, we present some of coalition sharing capabilities based on a controlled natural language (CNL) developed within the ITA to support knowledge-sharing and decision-making. Section 2 discusses knowledge-sharing and information extraction. Section 3 discusses CNL for human-to-machine interactions. Section 4 proposes a system architecture for CNL implementation based on International Technology Alliance Controlled English (ITA CE, hereafter CE). Section 5 discusses knowledge sharing and decision-making from the perspective of non-technical domain-specialist users and knowledge engineers. Finally, section 6 summarizes the CE discussion and future work in extending CE in areas such as syntax, semantics and general expressivity in order to be able to capture and represent a diversity of concepts and to support a wide range of coalition applications.

2. Knowledge-sharing and Information Extraction

As science and technology continue to advance, the volume of available data and information has been rapidly growing in both structured and unstructured forms, generated from sensors, intelligence reports, web commentary and other information sources. These data, diverse in format and in word usage, often contain critical information that, if extracted and properly represented, can provide significant insights to improve knowledge-sharing and to support decision-making. However, it has long been acknowledged that accessing critical information and improving shared understanding among coalition partners remains a major challenge for military coalition operations, especially for supporting distributed collaboration with teams and team members across multiple domains[2]. The difficulty stems from several dimensions. Team members from different domains often have different domain concepts due to different perspectives. Similar situations may be conceptualized and organized in different ways. As a result, organizations (even related organizations) may have somewhat different underlying conceptual models of the world.

Language variations constitute another problem dimension. Even for English speaking teams, the English language used by team members from different nations (such as US vs. UK) and/or from different organizations may vary to some degree in vocabulary, sentence structure, language usage and style. Same terms, phrases or commands may

[2] International Technology Alliance (ITA), among other major coalition research programs, has recognized shared understanding as one of the hard problems for current and future coalition operations (Verma 2009)

have different semantics and allow different pragmatic interpretations. For structured data, metadata may also vary in meaning between domains. Identical metadata elements may be used to refer to similar but distinct concepts. In addition, most data for decision-making is unstructured in form and difficult to automatically extract meaning from, as unstructured data contains irregularities and ambiguities. Typical examples of unstructured information are free text descriptions of entities (such as people, places, organizations, etc.), events, and situations. Furthermore, even structured fields in a database can contain unstructured data (free text).

To identify critical information and make it useful for decision-making, the disparate information must be processed and transformed into knowledge. Information extraction, among other text analytics techniques, is a process to extract key information items (such as entities, locations, and events) from the unstructured data sources and create a structured and semantic view of the information present in the data (Moens 2010; Cowie and Lehnert 1996). The output of information extraction is a collection of information items usually in some form of structured data system such as a database. For our purpose, namely assisting the users with knowledge-sharing, we focus on extraction of entities and facts (including events, general states of affairs or situations). This will necessarily involve extraction of relations between entities. Our information extraction method employs natural language processing techniques to parse the language text, recognize the sentence structures, detect properties of the analyzed sentence units, identify and extract the targeted information items, such as entities, relations and facts. In our system the extracted information is represented in the CE format using domain specific terminology according to the underlying conceptual model, rather than a computer technical format.

In summary, given large volumes of structured and unstructured information, knowledge sharing across the coalitions will need common information structures and representations that are unambiguous yet flexible to support communication, information sharing, coordination and interoperability among teams and team members across domain boundaries. In addition, given the problematic aspects of the data as discussed above, two preconditions are critical for effective information extraction: (i) the need for normalization and organization of free-text descriptions, (ii) the need for domain expertise for specifying and extending the domain model (ontology), including the vocabulary and terminology and their relation to the domain concepts.

3. Controlled Natural Language for Humans and Machine

3.1 Types and Functions of Controlled Natural Language

A controlled natural language (CNL) is a subset of a natural language (NL) using a restricted set of grammar rules and a restricted vocabulary. A number of CNLs have been proposed and developed for a common goal: to reduce or eliminate the ambiguity and complexity of a natural language, and thus to improve readability and interpretation of the text for humans or machines. Well-known examples of CNLs include ACE (Attempto

Controlled English, Fuchs et al. 1998), CPL (Computer Processable English, Clark et al. 2005), PENG (Processable English, Schwitter 2010), Rabbit (Talking Rabbit, Engelbrecht et al. 2009), Caterpillar Fundamental English (Verbeke 1973), and STE (Simplified Technical English, 2010)[3]. While sharing the basic common goal, CNLs can be categorized into two major types: CNLs that are primarily for human readers and writers to simplify readability and encourage more precise writing, and CNLs that enable automatic computational analysis and processing. STE, developed as an aerospace language (grammar and style) standard for airplane maintenance manuals, is a representative of the first type. Technical procedures are precisely described with minimal ambiguity so that the technical procedures and the related concepts can be correctly interpreted and comprehended by the readers, especially those whose native language is not English. The other CNLs listed above were developed primarily to facilitate various kinds of automatic processing, such as machine translation (Caterpillar Fundamental English), automatic proof generation (ACE and PENG), querying of the semantic web (Rabbit), and technical knowledge entry and retrieval (CPL).

CNLs typically specify that words be unambiguous and often specify which meaning is allowed for all or a subset of the vocabulary. For example, the English word '*replace*' can mean either '*substitute*' or '*put back*'. STE defines the word "replace" as only meaning '*substitute*'. Phrase or sentence structure also contributes to ambiguity. A simple example is concerned with noun clusters. In English, one noun is commonly used to modify another noun. A noun phrase with several nouns is usually ambiguous as to how the nouns should be grouped. To avoid potential ambiguity, many CNLs do not allow the use of more than 3 nouns in a noun phrase.

For the purpose of the work presented here, we are largely concerned with the second type of CNL, namely computer processable controlled language (CPNL henceforth). Intuitively, we might assume that a controlled language that is easy for people to understand would also be easier for machines to process. However, it turns out that the constraints for easy human comprehension and those for easy machine representation and processing are different. The most important difference is that human readers tolerate a degree of uncertainty and are often able to resolve ambiguity to a large extent while it is very difficult to get computers to deal with ambiguity in a reasonable way. This difference leads to two different philosophies and approaches in designing CPNL[4]. One approach treats CPNL as a simplified form of NL, thus treating CPNL processing as a simplified form of NL processing. Allowing certain degrees of ambiguity in the language, this approach aspires to keep the CPNL natural and user-friendly as much as possible, while using standard NL processing techniques, lexical-semantic resources and the domain model to select an optimal interpretation among multiple possibilities.

[3] MacDonald, M.L., Simplified Technical English For All; A Customer-friendly Specification. AeroSpace and Defence Industries Association of Europe (ASD), 2008, http://www.x-pubs.com/resources/2008conf/downloads/4X-Pubs2008_Maria_McDonald_Simplified_Technical_English_For_All.pdf

[4] See Clark et al. for a detailed comparison of these two philosophies and a detailed discussion of the relevant issues.

In contrast, the other approach focuses on the computational aspect, treating a CPNL more as an English version of formal language where the CPNL interpretation is completely deterministic, following “one sense per word” principle and allowing absolutely no ambiguity. This deterministic property applies to both the lexicon and grammar. A significance of this property is that the interpretation is predictable and the computation is reliable and very efficient. While a CPNL of this kind is easier to use by humans than a regular formal language, it is by no means easy for users who have not had any training in this language in terms of both lexicon and grammar, because the restricted grammar and lexicon of this CPNL will constantly compete with his/her normal English intuition (i.e., the grammar and lexicon that the user has been exposed to since his/her birth). In short, a restricted version of English in this form is not always easier for users than ‘full’ English. In fact, there is often a tension between the user-friendliness and predictability. The closer the CPNL to the normal NL, the more natural and the easier to use by humans, but the less predictable and the more computationally complex it will be. The converse is also true. The more deterministic the CPNL is, the more predictable it is, but the more difficult it is for human to use. We will return to this issue below.

CE is designed to support both human usage (generation as well as readability) and machine processing, specifically providing:

(i) A user-friendly language in a form of English, instead of, for example, a standard formal query language (e.g., SPARQL or SQL), which enables the user to construct queries to information systems in an intuitive way
(ii) A precise language that enables clear, unambiguous representation of extracted information to serve as a semantic representation of the free text data that is amenable to rule-based inferencing
(iii) A common form of expression used to build, extend and refine domain models by adding or modifying entity, relation, or event types, and specifying mapping relations between data models and terminology or language variants
(iv) An intuitive means of configuring system processing (such as specifying entity types, rules, and lexical patterns)

As CE is designed for both human and machine, a good balance between the naturalness and predictability of the language is fundamentally important, which will need to take into account both theoretical considerations and results and feedback arising from empirical experimentation.

3.2 ITA CE: a Brief Introduction

CE is consistent with First Order Predicate Logic and provides an unambiguous representation of information for machine processing, while aspiring to provide a human-friendly representation format that is directly targeted a non-technical domain-specialist users (such as military planners, intelligence analysts or business managers) to encourage a richer integration between human and machine reasoning capabilities (Mott 2010, 2009; Mott et al. 2010; Mott and Hendler 2009). CE builds upon earlier work on Controlled

Natural Languages, such as Common Logic Controlled English (Sowa 2007) and aims to provide a single standard language for representation of all aspects of the information representation and reasoning space. In addition to more traditional areas such as knowledge or domain model representation and corresponding information, CE also encompasses the representation of logical inference rules, rationale (reasoning steps), assumptions, statements of truth (and certainty) and has been used in other areas such as provenance and argumentation.

The CE syntax is readily compatible with existing ontology modeling languages such as OWL (Web Ontology Language), and capabilities to convert information to/from OWL ontologies and process associated RDF data have been implemented. Consideration has also been given to the CE query syntax as against relevant Semantic Web query capabilities such as SPARQL and SWRL (Semantic Web Rules Language) as well as technologies such as RIF (Rule Interchange Format) (Mott 2009). In order to briefly introduce the CE syntax some simple examples are given below.

First of all the creation of the domain model (or a general model across domains) using CE is accomplished by the definition of (domain) concepts, relationships and properties. These are all achieved through the "conceptualise"[5] statement:

```
conceptualise a ~ person ~ P.
```

After a conceptualise statement had been made the concept in question has been created within the CE domain model and statements relating to that concept can be made:

```
there is a person named Fred.
```

A slightly more advanced example would be:

```
conceptualise a ~ person ~ P that is an agent.
conceptualise the person P
    ~ is married to ~ the person P2 and
    has the value A as ~ age ~.
```

Thereby creating "person" as a subconcept of "agent" and indicating that it can have the property of "age" and enter into a "married" relationship with someone, allowing:

```
the person Fred is married to the person Jane and has 54 as age.
```

The examples given so far have included two simplistic lexical styles of asserting relationship information: Verb Singular (using the "is married to" example), and Functional Noun (using the "has as age" example). The simple addition of this one literary device to allow the stating of information in these two simple forms has enabled the creation of more human-friendly CE statements, and the coupling of this with the

[5] The spelling of "conceptualise" is due to the origin of CE at IBM, UK.

statement of the concept name within the sentence and the ability to use space delimited “normal” names (rather than typical CamelCase names prevalent in other ontology languages) aids the general readability of the CE sentences. The CE language supports multiple inheritance (implemented via the use of the “is a” syntax within the conceptualise sentences) and also allows any instance to be asserted as any number of concurrent concepts, for example “the person Fred is a mechanic and is a lottery winner and is an academic example.” – valid CE, and note the “an” which is used whenever required.

Clearly the examples given above are very simplistic, but CE has been used in a number of example applications with a reasonable number of concepts, relationships, queries and rules used to model and interact with complex real-world environments with a high level of coverage and practical expressivity being achieved.

3.3 Language Facts, Linguistic Description and Conceptual Matters

From a processing of view, CE is used for two purposes in the context of an information extraction system: (i) the target of the linguistic processing, where the CE is acting as the unambiguous semantic representation language; (ii) a means by which language data are analyzed and modeled. For the first purpose it is necessary to have a conceptual model of the domain and to know the mapping between the words in a sentence and the concepts in the domain conceptual model, and the mapping will involve disambiguation which will be discussed in more details in Section 4.3; for the second it is necessary to have a model or grammar that describes language facts, including linguistic categories and relations. More precisely:

> (i) The General Linguistic Model, contains our theory of language in general, including such concepts as ‘lexicon’, ‘word’, ‘phrase’, ‘noun phrase’ (all subconcepts of ‘symbol’), syntactic relations such as ‘head’ and ‘dependent’, structures such as ‘linguistic frame’ which holds relationships between CE statements about syntax and semantics, and semantic hierarchies such as ‘WordNet synsets’ (as described below). This model also defines the relations and boundaries between CE and normal English.
>
> (ii) The Domain Conceptual Model, containing specific concepts of entities and relations (for example this might include ‘IED’ or ‘village’ or ‘ambush’). More generally, a domain model can be considered a specialization of a Common Conceptual Model, which contains the concepts of entities and relations that are commonly used across the relevant domains, such as ‘artifact’ or ‘place’ or ‘act/perform an action’.

As described in more detail below, the parser agent turns a syntactic parse tree into a set of CE sentences that is easier to process via linguistic rules. These sentences use the concepts defined in the general linguistic model. Given the sentence “the patrol in East Dulwich discovers the factory”, this might be initially turned into sentences including:

> the noun phrase np1 has the noun |patrol| as head and has the prepositional phrase pp1 as dependent and stands for the thing [001].
> the prepositional phrase pp1 has the word |in| as head and has the noun phrase np2 as object.
> the noun phrase np2 has the proper noun |East Dulwich| as head and stands for the thing [002].

Here the syntax tree is represented in attributes such as ‘dependent’ and ‘head’, and the (minimal) semantics as ‘stands for’ (based on the idea that each noun phrase stands for some object in the domain).

3.4 Mapping between Language Facts and Domain Concepts

In order to map between the syntax of the sentence and the semantics of the domain, we are currently employing an open source parser (specifically the Stanford Parser, Klein and Manning 2003) to provide a basic syntactic parse tree, allowing users to focus on the mapping of this parse tree into the meaning of the sentence, i.e. the specific entities, events, and situations represented in the analyst’s domain conceptual model.

The construction of the semantics may be considered at two levels: mapping to general semantics (that which is independent of a specific domain) and mapping to specific semantics (that which is defined in the domain model). We undertake this mapping in an incremental fashion, matching general patterns inferring the general semantics followed by rules that match more specific domain-based patterns adding inferences about the more specific semantics. Since the domain model is itself based on general concepts, this incremental mapping allows the more specific information to be consistent with the general information, but adding more detailed constraints. More specifically we undertake the mapping using the following functions (which may not necessarily follow this sequence):

- Words in the parse tree are matched to concepts in the domain conceptual model
- General structures in the parse tree are matched to generic semantic concepts
- Specific structures in the parse tree are matched to specific concepts
- Further inferences are made about the specific entities using domain specific rules

Matching words to concepts is undertaken via CE sentences such as:

> the noun |patrol| expresses the entity concept ‘patrol unit’.

based on the semantic generalization that nouns typically represent concepts which are realized (or instantiated) by entities in the domain. Such linking sentences must be derived from the analyst’s understanding of the meaning of the concepts (s)he defined, and a tool called the Analyst’s Helper is being developed for this purpose (see below). Analogous sentences will be used to map verbs to events and adjectives to properties, in the prototypical cases.

The following is an example of how a mapping between syntax and semantics may be represented as logical rules in the linguistic model. The concept of ‘container’ captures the idea that if something is “in” something else (for example expressed as a prepositional phrase headed by “in”), then the second in some sense “contains” the first. The current rule to infer this is:

```
if ( the noun phrase NP1 stands for the thing T1 and
        has the prepositional phrase PP as dependent ) and
          ( the prepositional phrase PP has the word '|in|' as head and
        has the noun phrase NP2 as object ) and
          ( the noun phrase NP2 stands for the thing T2)
then
  ( the thing T1 is contained in the container T2 ).
```

Here the rule preconditions will match on earlier parse tree CE sentences to infer:

> the thing [001] is contained in the container [002].

Additional inferences based on other rules can infer more specific facts, such as that the thing [001] is located in the location [002] if, in fact, [002] is of conceptual type “location”.

3.5 Final Meaning Representation

After identifiers for the entities, events, and situations are created and assigned their appropriate concept in the domain model and all necessary properties expressed by adjectives and other modifiers are represented, the meaning of the input sentence is represented as one or more CE sentences. So, for example, the sentence:

> BCT patrol in South Baghdad discovers a bomb-making facility on Hilla Road.

is represented as the CE sentences[6]:

> the patrol unit '|BCT patrol|' finds the facility '|p6|' and is located in the place '|South Baghdad|' and is a NATO military unit.
>
> the facility |p6| makes the device bomb and is located on the road ‘|Hilla Road|’.

The sentences, or their underlying component sentences, can then be returned in response to queries about where bombs might be made or significant NATO activity in East Rashid. For example, a possibly query might look something like this:

[6] Note that the second sentence is not actually produced by the current system, but is in the spirit of CE and something like it is expected to be produced in the near-term.

for which F1 is it true that the facility F1 makes the device type bomb and is located in '|South Baghdad|'.

The sentences could also be used in logical inference processes to compute additional information in the CE format.

4. System and Architectural Description

The aim of CE is to provide a common form of information representation that can be used by all parties, with different (but overlapping) domain models supporting each specialization in support of the whole endeavor. In addition to this there are some tooling capabilities, such as the "CE Store" that can be used to directly support some of the requirements of specialist users.

CE is designed to be most useful in situations that have the following characteristics:

(i) A high degree of human interaction, usually involving specialist users with complex needs in non-trivial environments.
(ii) A likelihood of rapidly evolving or uncertain tasks, queries or other knowledge-based activities.
(iii) The need for collaboration, either between different people or teams, and/or across different disciplines.

CE is of little value if there is no human-involvement, little complexity, or very firm and stable requirements, and in such circumstances traditional application development processes are a much more straightforward and low risk solution. In cases where there is a high degree of customization, development, uncertain requirements or short lead times, especially in areas where human-led planning, thinking or decision-making are required then CE (or similar human-friendly information processing environments) could be a very useful capability.

The system has the following major components with an architecture as illustrated in *Figure 1* below:

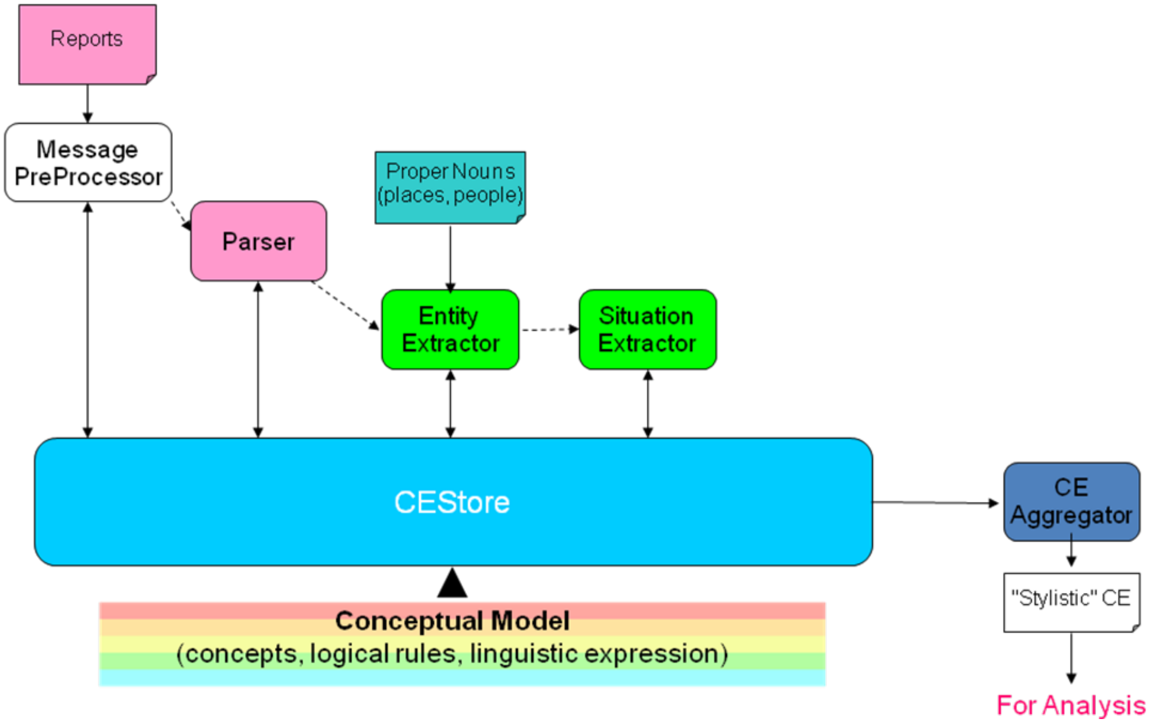

Figure 1: Processing Architecture

4.1 The Analyst's Helper Module

Our approach to linguistic processing relies upon the linking of words to concepts, specifically via the "expresses" sentences. Whereas the meaning of natural language words is generally understood by the community of speakers, the authoritative meaning of the concepts is only known to the analyst who developed the conceptual model. Only the analyst can determine the linking of words to the concepts, although (s)he may be assisted by tooling to perform this task. To this end we are developing an "Analyst's Helper" (AH) to assist the analyst in constructing the linguistic mappings between words and each concept in the conceptual model, that is the "expresses" sentences. To reduce the burden on the analyst, the Analyst's Helper uses WordNet (Miller 1995, Fellbaum, 1998) to suggest possible words for each concept. Each concept in the domain model is matched to all possible WordNet synsets (via a simple analysis of the words in the word senses) and the analyst is invited to choose the best matching synset from those found, or create new synsets if needed. When the choice is made, the Analyst's Helper constructs suitable CE sentences describing the match between the synset and concept, and constructs 'expresses' CE sentences linking the words in the synset and the concept.

Rationale[7] for these sentences is also specified, to allow future explanation of the NL processing steps.

An extended Analyst's Helper will further aid the matching process and allow more complex matching of verbs and adjectives to offer more "remotely" matching synsets and to feedback the sets of unrecognized words from the parser for consideration by the analyst. It may also be possible to build a set of predefined concepts and word/concept mappings which may be used as the basis for the building of a conceptual model by the analyst.

4.2 CE Store

The CE Store provides a basic CE processing environment that includes the following high-level capabilities:

(i) Basic CE sentence parsing
(ii) Define/extend any concept model
(iii) Assert any CE sentence conforming to the appropriate conceptual model(s)
(iv) Define and execute any CE query including an example "visual query composition" element
(v) Define and execute any logical inference rule, in the form of a "query with conclusion clauses" that can be used to assert new CE information
(vi) Define and execute any "CE agent" in the form of Java code which conforms to a simple "CE Store" interface
(vii) Operate entirely in memory, or persist information to a relational database format
(viii) Example web-based client to allow rapid development and browsing of CE-based information
(ix) Example agents to carry out basic information processing tasks
(x) Some capability to convert to/from OWL and RDF formats

The purpose of the CE Store is to demonstrate a "pure" CE-based implementation of an information-processing environment within which human and machine agents can contribute and interact with complex information based on common conceptual models of a domain. Actual applications can decide to include other non-CE based elements (e.g. other visual interfaces) in their human computer interface per application requirements.

4.3 Information Extraction Module

The Information Extraction Module is a central component of this system, performing the processing to extract information from the sentences and to convert it into CE sentences,

[7] Rationale within CE is the formal explanation of the inference steps that were taken to reach a conclusion. The rationale information is also expressed in CE and may contain information about assumptions and true/false support pathways. It is not discussed in detail in this paper.

using the formats defined in previous sections. This is based upon a sequence of agents running within the CE Store. Each agent reads the relevant CE sentences from the CE store, performs some processing and places the resulting CE sentences back into the CE store. In this scenario the following agents are executed:

(i) The reports are converted into sentences via the Message Preprocessor agent.
(ii) The parser agent is called on a sentence. This calls the parser[8] Java API code to produce a raw parse tree, and then turns the raw parse tree into a CE representation (defining phrases with heads and dependents, as described previously). Parsing sentences involves linguistic processing, to which we will return with more details below.
(iii) The entity extractor agent analyses the CE head/dependent representation and uses entity extraction rules to generate information about the 'things' that the noun phrases stand for, adjective phrases and prepositional phrases as outlined above. The result is a set of entities, their characterizations as domain concepts, and relations between them, as a set of CE sentences. As part of this processing, reference information is used, including:

   a. the 'expresses' links between words and entity concepts
   b. fact bases of proper nouns and their categorizations (e.g. place names, organizations), and their domain-level attributes (e.g. the coordinates of places)

(iv) The situation extractor agent further analyses the CE head/dependent representation of the parse tree together with information about the entities extracted in the previous step. This uses additional rules to determine the roles played by different participants in the situation (e.g. agent, patient, instrument) based on the verb. The result is a set of CE sentences about the situation.
(v) A "naming" agent is run to provide more readable names for the entities.
(vi) As a result of the previous steps, there are a number of CE sentences describing the entities and situations. Due to the incremental nature of the architecture, these sentences are small and atomic in form, and are best presented to the user in a more natural aggregated form. Thus a final "CE aggregation" processor is run to turn the atomic CE into a more "stylistically felicitous" CE, using techniques such as: aggregating all information about an entity into a single sentence; not duplicating information; not displaying supertypes that are obvious; and not displaying relationships that are easily inferable from other relationships.

The final output, the set of CE sentences representing the entities and relations is now available for further processing and analysis, via machine or human.

The brief description above has omitted details of certain steps, especially (ii). Parsing sentences in free text form and converting them into CE sentences involve a series of

[8] Current implementation uses the Stanford Parser. Future implementations may differ.

steps of linguistics processing such as word disambiguation and reference resolution (e.g. what previous noun phrase a pronoun refers to).

Information extraction aims to extract certain types of information, depending on the interest and information requirements of the domain. Information that is useful to one domain may not be interesting to another domain. In this work, we are taking an ontology-based information extraction approach. The rationale for this comes from the fact that coalition operations are usually task oriented with specific targets and objectives. The domain conceptual model that provides explicit specifications of concepts within the domain plays a crucial role in our information extraction process. Entities and events are primary types of information to be extracted. Based on the domain model, our system correlates the conceptual representations and lexical/grammatical representations by means of linguistic frames, which encode predicate entity relations representing events. As information requirements differ from one domain to another, the domain conceptual model provides the guidance for identifying the types of predicate-entity relations, which are used to infer events of interest.

5. Support Knowledge-sharing and Decision-making

As mentioned earlier, the purpose of the CE language is to provide a more human – friendly information representation language to lower the technical barrier between such users and the capabilities of the information processing system. While each coalition operation has its specific objective and requirement, information access for knowledge-sharing is a common and pervasive need across coalition domains and objectives. As mentioned in Section 4, this system is designed to support use cases where the users may play different roles with different information needs and thus interact differently with the system. We envision two major types of users:

(i) Non-technical domain-specialist users (such as analyst, military personnel), who use CE to query the system and add new concepts to an existing domain model.

(ii) Knowledge engineers (including system developers and domain model owners), who use CE to extend, refine, and configure the knowledge base (general or base domain model and the general linguistic model).

The non-technical users are the majority who use the CE to query the system for information extraction purpose. They are primarily looking for facts and information from their own domain and other relevant domains that would be helpful for situation awareness and decision-making. They have the knowledge of their own domain but may not be familiar with the concepts and language expressions used in other domains. In order to be useful for this group of users, the information extraction system needs to meet the following requirements:

(i) A user-friendly language for querying

(ii) A user-friendly report representation

(iii) A common form of representation that maps between domain models
(iv) A common form of expression for terminology or language variants

Currently CE query syntax supports the ability to ask for a result set ("for which V1 …") or a count ("how many V1…") with the preamble part of the query being the only difference in syntax. Depending on the operational environment the CE query will return either a simple result set (in a manner similar to the results of a SQL or SPARQL query), or will return a set of CE statements that correspond to the matches within the CE corpus for the given query term. For example:

> the person Fred is married to the person Jane.
> the person Mary is married to the person Charles.

As CE is a form of English, it will be easier to understand by users who don't have formal training than any standard formal query language (such as SQL) would be. While it is straightforward to read CE sentences, it should be noted that some training is necessary for the users to write or construct CE queries that accurately represent the meaning intended. The reason is that each CE representation has a specific interpretation. Accurate interpretation of a CE expression requires knowledge of CE grammar, vocabulary and the domain model(s) which restrict the interpretation of the CE expression. For example, the word "against" has several meanings in English, including "in opposition to", "in contact with", etc. Suppose that "against" is a word in CE and has a single meaning. The user will need to know exactly what it is supposed to mean in CE in order to construct query with the precisely intended meaning.

User interactive capabilities of the system can alleviate these problems, assisting the non-technical users with CE when the system provides sufficient feedback (such as questions and/or examples) to the user so that there is no confusion about what the system's interpretation of the query is. For knowledge engineers and technical users, who are knowledgeable of the CE and the associated domain conceptual model, this is usually not a problem. With the knowledge of CE, the technical user knows the specific concepts (s)he wishes to express, what words can be used and how they can be combined to refer to these concepts. CE also has an annotation syntax which allows unstructured human comments and descriptions to be added to the CE sentences whenever needed.

Knowledge sharing across domains is challenging. Different but related domains overlap but also differ to some extent in concepts and terminology. A common model is necessary, which is an aggregation of all the concepts and terminology of the related domain models as well as the mapping relations between those that are related but different. CE plays a crucial role in defining the mapping between domain models. As we mentioned above, it is easy enough for the user to use and precise enough for the machine to process and interpret.

6. Conclusion and Future Work

Multi-team and multi-nation collaborations in current and future coalition operations involve conceptual as well as terminological and other linguistic variations across domain models, which pose major challenges for information sharing among teams for efficient C2 operations. Information extraction using CE can provide a powerful and practical means for improvement of information access and knowledge sharing across domains. CE is a simplified and common form of expression in English, which is not only user-friendly in nature but is also restricted in vocabulary and grammar for clear, unambiguous representation and interpretation. More precisely, CE meets the requirements for information extraction to improve cross-domain knowledge sharing and support decision-making. CE provides:

(i) a user-friendly language for querying
(ii) a user-friendly system-to-user report representation
(iii) a common form of representation that maps between domain models
(iv) a common form of expression for terminology or language variants.

We employ natural language processing techniques to process the language text, recognize the sentence structure, detect properties of the analyzed sentence units, identify and extract the targeted information items, such as entities, relations and facts. While our overall process of information extraction does not differ from most systems, our unique use of CE plays an important role in this process, with CE providing a form of representation for further analysis, information interpretation and representation of the final information report. In the meantime, CE also serves as a user-friendly and flexible way of allowing users to improve the information extraction process. Users, for example, can use CE to augment the domain model and the vocabulary used by the information extraction module and augment interpretation of sentence structure to identify key information items.

While this is a viable approach to enable information extraction and representation, the current CE implementation is relatively basic and needs extension in the areas of syntax, semantics and its general expressivity in order to be able to capture and represent a diversity of concepts and to support a wide range of use cases. For example, the current CE doesn’t allow use of prepositional phrases although the prepositional phrase is a widely used English phrase structure with various grammatical functions. Allowing use of prepositional phrases will no doubt make CE closer to normal English and more natural for the users. However, this also unfortunately adds to the potential for ambiguity, such as the notorious problem of prepositional phrase attachment (does a final prepositional phrase modify the last noun or the whole sentence?). How to allow it in a restricted way while maintaining the unambiguousness of current CE would be a major challenge. We believe that modification and/or extension of CE will need to be based not only on theoretical considerations but also on empirical evidence from usability studies and experimentation of real use case scenarios.

In addition to ordinary end-users, CE can also play an important role in assisting domain modellers and linguists. Entity recognition is the basis of information extraction. But the relations between the entities are essential for the entities to be part of the information and for the information extracted to be useful. The CE-based information extraction system is a highly user interactive system with a set of user-friendly tools to leverage the knowledge that the user already has. For example, knowledge engineers who are familiar with domain concepts and with CE can easily extend and refine the CE representation of the concepts as well as the mapping relations among the related concepts within a domain and/or across domains, while linguists can help with the analysis of linguistic expressions and representation and the linking of the lexical and domain models. As we continue to improve conceptual matters on the one hand and a further enhanced CE representation of the conceptual models on the other hand, we believe that CE-based information extraction will truly facilitate the processes in the cognitive and social domains that enable working together effectively and efficiently.

ACKNOWLEDGMENT

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

Clark, P., Harrison, P., Jenkins, T., Thompson, J., and Wojcik, R. (2005). Acquiring and Using World Knowledge using a Restricted Subset of English. In Proceedings of . FLAIRS'05.

Cowie, J. and Lehnert, W. (1996). Information Extraction, Communication of ACM Volumn 39 Issue 1.

Engelbrecht, P., Hart G., and Dolbear, C. (2009). Talking Rabbit: a User Evaluation of Sentence Production. Ordnance Survey. Workshop on Controlled Natural Language (CNL 2009). 8-10 June 2009. Marettimo Island, Italy. Appears in Controlled Natural Language, Volume 5972, of Springer’s LNCS/LNAI series.

Fellbaum, C. (1998, ed.). WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.

Fuchs, N. E., Schwertel, U., and Schwitter, R. (1998). Attempto Controlled English. Proceedings of LOPSTR'98.

Klein D. and Manning, C. D. (2003). Accurate Unlexicalized Parsing. Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430.

Miller, G. A. (1995). WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.

Moens, M.-F. (2010). Information Extraction: Algorithms and Prospects in a Retrieval context. Springer.

Mott, D. (2009). The representation of logic within semantic web languages, ITACS, url: https://www.usukita.org/papers/5242/details.html

Mott, D. (2010). Summary of Controlled English, ITACS, https://www.usukita.org/papers/5658/details.html.

Mott, D., Giammanco, C., Braines, D., Dorneich, M., and Patel, D., (2010). Hybrid Rationale and Controlled Natural Language for Shared Understanding. In Proceedings of the Fourth Annual Conference of the International Technology Alliance, London, UK, September 2010.

Mott, D and Hendler, J. (2009). Layered Controlled Natural Languages, In Proceedings of the Third Annual Conference of the International Technology Alliance, Maryland, USA.

Ogden, C. K. and Richards, I. A. (1923). The Meaning of Meaning.

Simplified Technical English. See http://en.wikipedia.org/wiki/Simplified_English Retrieved 8/18/2010.

Schwitter, R. (2010). Processable English. See http://web.science.mq.edu.au/~rolfs/peng/ Retrieved august, 18, 2010.

Sowa, J. (2007). Common Logic Controlled English, http://www.jfsowa.com/clce/clce07.htm.

The Stanford Parser, A statistical parser, http://nlp.stanford.edu/software/lex-parser.shtml

Verbeke, C. A. (1973). Caterpillar Fundamental English, Training and Development Journal, 27, 2, 36-40, Feb 73

Verma, D. (Ed.) (2009). International Technology Alliance in Network and Information Sciences, Biennial Program Plan 2009, Applicable Period: May 12th 2009 - May 11th 2011).