Lexpresso: A Controlled Natural Language

Adam Saulwick

Defence Science & Technology Organisation, Australia

Abstract. This paper presents an overview of 'Lexpresso', a Controlled Natural Language developed at the Defence Science & Technology Organisation as a bidirectional natural language interface to a high-level information fusion system. The paper describes Lexpresso's main features including lexical coverage, expressiveness and range of linguistic syntactic and semantic structures. It also touches on its tight integration with a formal semantic formalism and tentatively classifies it against the PENS system.

Keywords: Controlled Natural Language, Formal Semantics, Linguistic structures, Human–Computer Interaction, High-level Information Fusion.

1 Introduction

'Lexpresso' is a Controlled Natural Language (CNL) developed at the Defence Science & Technology Organisation as a bidirectional natural language interface to a high-level, agent-based, information fusion system called Consensus. This paper is the first published description of Lexpresso's broad features, including lexical coverage, and range of syntactic and semantic structures. It also describes the tight integration with DSTO's bespoke formal semantic formalism, Mephisto, initially conceived by Lambert & Nowak [1]. Lexpresso was first developed in 2008 and is under active development.[1]

The Consensus system performs high-level information fusion of heterogeneous data for Situation Awareness. In our current demonstration system, synthesised input data types include maritime and aviation tracks[2], natural English texts, emails and spoken English statements. In general terms the Consensus system is designed to demonstrate a working solution to problems of high-level information fusion by the 'semiautomation of [some of] the functionalities of

---

[1] Some previous work in natural language interfaces at DSTO focussed on automated speech-to-text recognition linked to template rules. From June 2007 to July 2008 a DSTO initiated collaborative research program into situation awareness was conducted between DSTO and NICTA, see [2]. Among other things, this research involved the development of a CNL which was based on, or inspired by, PENG [3, 4]. Subsequent to these activities Lexpresso was built from scratch by DSTO.

[2] Synthesised tracks are currently processed at circa 100 per second and contain fields for source, temporal offset, track ID, time, coordinates, direction, speed, class, type, allegiance and nationality.

sensation, perception, cognition, comprehension, and projection that [are] otherwise performed by people for situation awareness' [5]. Among other functions Consensus does this by automatically transforming diverse information sources into a canonical semantic machine-readable form called Mephisto which facilitates computational reasoning. A problem however is that Mephisto is only interpretable by machines or by a few human experts (and then slowly) and so, human interaction with Consensus would be very difficult, if not impossible, even for specialists, without a natural interface. Lexpresso is that interface. It bridges the natural-language/formal-language gulf and thus it permits relatively natural interaction with a formal semantic reasoning system via spoken and written controlled natural English.[3]

Lexpresso's bidirectionality means that it has both input and generation capabilities. Further, because Lexpresso is tightly coupled with Consensus's formal semantic knowledge representation and reasoning system whose primary function is automated inferencing over real-time track data and texts for enhanced situation awareness, it provides human users with the enhanced ability to query the nature of current and historical real-world and potentially far-flung events. Answers are given in the form of situation reports. These reports may concern the transit or spatiotemporal interaction of observed maritime, land &/or air-based platforms and even the social relationships between people inferred from certain text descriptions.

While these capability descriptions are accurate, they are not intended to obfuscate Lexpresso's limitations. Its breadth and degree of coupling with Mephisto are the subject of ongoing research and development. Consensus is a prototype system and, subject to space constraints, some limitations will be mentioned in Section 4.

The remainder of this paper is structured as follows. Section 2 describes the system architecture and the main CNL modules. Sections 3 and 4 exemplify the main syntactic and semantic structures respectively. Section 5 proposes a classification of Lexpresso based on the PENS system. Section 6 concludes with a summary.

2   System Architecture and Module Functions

Echoing the traditional transformational grammar distinction [6, 7], language processing in Lexpresso is conceived on a spatial metaphor of depth in which 'surface CNL' refers to observed spoken or written forms of language and 'deep CNL' refers to an underlying abstraction with certain linguistic features. Input processing takes surface language and transforms it into deep linguistic structures. After further processing to remove ambiguities, these are converted into our universal semantic representations called Mephisto structures. Reasoning and inferencing is primarily performed on Mephisto structures. Output processing takes Mephisto structures and uses the same core syntactic parser to validate and generate surface CNL for consumption by users.

---

[3] Consensus also utilises other interfaces such as a 3-dimensional geospatial display and a virtual adviser avatar. These are not discussed here.

Lexpresso is designed as a modular system to facilitate integration of new features as required. Depending on how one counts them, it consists of around 17 modules, see Figure 1. Due to space constraints not all components are described.
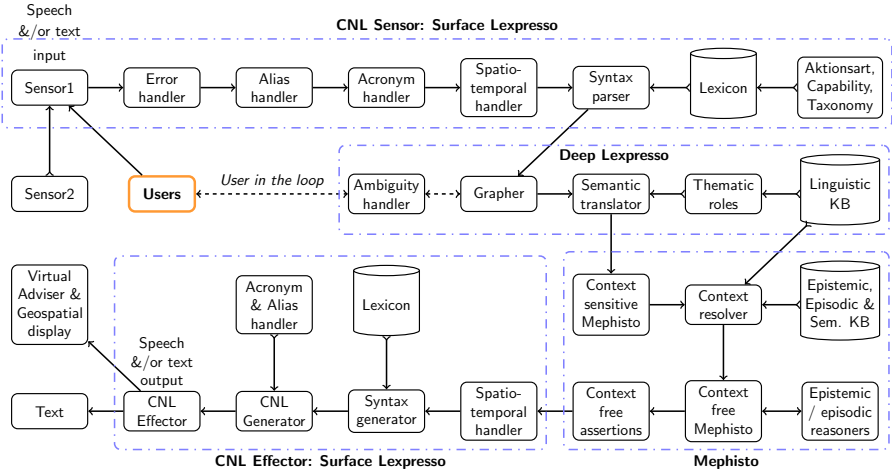

Fig. 1. Lexpresso system architecture

CNL Sensor. This is the users' primary input screen. It contains a text panel for typing controlled natural English. Spoken English also appears in this panel via the Automated Speech Recogniser. Sentences can be automatically or manually inserted. Manual insertion is an interactive process and thus permits error correction prior to further processing. Once inserted the input appears in the CNL Sensor log window accompanied by the name of the 'teller' and a status message. Each line is also timestamped, displayed directly above it, see Figure 2.[4]

Error handler. During manual insertion, a pre-parser checker notifies the user of unknown or undefined words or out-of-grammar expressions. This provides dynamic feedback on lexical coverage and grammaticality of surface input to inform the user of input status.

Alias handler. This module converts particular multi-word expressions into atomic terms for manipulation at the deep linguistic level, e.g. 'Becker', 'Bender', air, force, base becomes becker_bender_AFB. It is also used to handle contractions, e.g. 'can't' becomes 'cannot'. It is also used for mapping fixed idiomatic forms to a single lexical correspondence. Although the functionality of the aliasing module is currently used for simple surface level structures, it is also capable of handling metonymy.

---

[4] The 'sensor' & 'effector' terminology is adopted from the Attitude Too cognitive model [8].

Acronym handler. This module expands acronyms and titles into multi-word expressions. It also constrains their syntactic position (e.g. pre-/post-nominal) based on their part-of-speech.
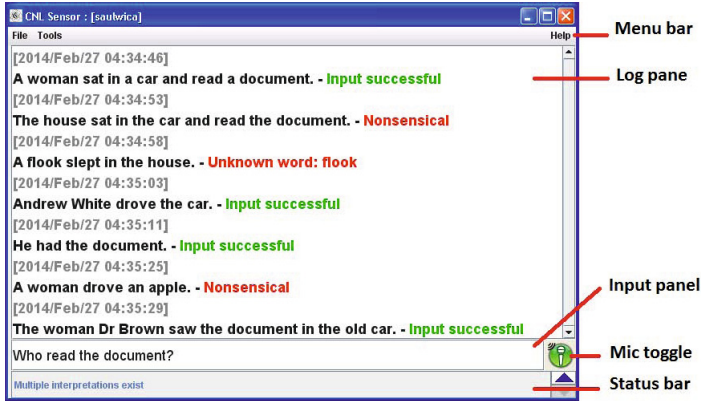
Fig. 2. CNL Sensor: showing sample text (with timestamps, proper names, person title & anaphoric resolution), input panel (with possible query), colour-coded feedback messages in log pane, microphone toggle button (on) for speech input & status message

Spatiotemporal handler. This module converts date and time information into universal standard timestamps, accepting a broad range of natural language expressions such as time-formatted numerals (e.g. 13:59:59 or Zulu time, and a variety of time points in natural language, e.g. 1 PM, one o'clock). Each time is calculated against Coordinated Universal Time (UTC). Our system is designed to allow the sensor, effector and cognition modules to be in different spatiotemporal locations, hence the generation of temporal phrases is calculated on an off-set to UTC.

Temporal intervals are also handled. Surface level forms include the template 'from TIME to TIME', and a range of temporal words, including 'today', days of the week, months of the year, decades, and centuries, etc. Where required by natural English, these phrases can be further modified with prepositions, such as 'from 12:00 to 13:00', 'in January', 'for a week', and with temporal grounding to specific times, such as 'last week', 'yesterday', 'in one month'. Inferencing with these temporal expressions as reference points is done in the cognition using Allen interval algebra [9], not discussed further here.

Lexpresso pays careful attention to the subtleties of the English tense system. For instance the relationship between the time of user interaction and the tense of the assertion or query is captured and stored as temporal information relevant to each entity. To achieve this, each input is internally labelled with utterance type and time. The latter becomes the reference time for every CNL interaction.

For instance, in (1),[5] via a `tells` predicate, the system registers that it has perceived a CNL Sensor interaction at a certain time and from a certain teller; here the author.

(1)   ```
         Mon Jun 02 10:33:48 CST 2014 [SENSOR : INTERACTION]
         perceive(cnl_sensor,tells(teller(@(skc1,invl
         (timestamp(2014,6,2,1,3,48),timestamp(2014,6,2,1,3,48)),s_5),
         Adam_Saulwick),...)),
         ```

Example (2) demonstrates how the system stores spatiotemporal information about entities.[6] Given the simple past tense of the surface input sentence in (2a), the time of 'standing' is encoded at the deep Mephisto cognitive level (2b) to have taken place *before* the time of the assertion registered in (1).

(2)    a.  I: The woman stood in the house.
       b.  C: ```
           animate(@(skc2,t_4,s_2)),female(@(skc2,t_4,s_2)),
           before(t_4,invl(timestamp(2014,6,2,1,3,48),
           timestamp(2014,6,2,1,3,48))),
           location_in([stands(@(skc2,t_4,s_2))],@(skc3,t_4,s_3)),
           woman(@(skc2,t_4,s_2),[animate,definite,singular,...]),
           house(@(skc3,t_4,s_3),[definite,singular,prep(in)]),
           stands[@(skc2,t_4,s_2)],[past,...])).
           ```
       c. O:The woman stood in the house before Monday the 2nd of June 2014 at 10:33:48 AM.

For explicitness by default the time before which the event is asserted to have occurred is rendered visible to the user in the CNL Effector window, as in (2c).[7]

The location of the 'standing' event in (2) is encoded via a `location_in` predicate which is formed on the fly from a combination of surface language and automatically identified semantic roles (see *Thematic roles* below). (2b) shows this binary predicate with manner and referenced arguments, each encoding

---

[5] Our notation uses a ternary @-predicate—adopted from the Mephisto conceptualisation with a perdurantist philosophy which 'holds that an identity is formed from different things at different times, that an identity is a process, an assembly of different temporal parts'—representing @(label, time, space). (Space constraints prohibit discussion of this, but see [1].) The first argument is the entity's identifier (here a Skolem constant), the second is a temporal point, interval or even a sum of temporal intervals, and the third is the space it occupies. See [10] for further details.

[6] Abbreviations to the examples indicate representations at Input (I), Cognition (C), and Output (O) levels. Skolem constant numbers have been simplified for expository purposes but numbers are automatically assigned at input. Further, information not germane has been omitted and replaced by ellipses. Space limits a full description of the contents of C here.

[7] A number of the other underlying linguistic features of the surface input are also sent to the cognition—namely animacy, gender, part-of-speech, number, and definiteness—as exemplified for the intransitive simple past phrase in (2). These are used for various internal purposes, such as anaphoric resolution, predicate argument typing and grammatical agreement.

their skolem identifiers, space and time. As with all Mephisto structures, this deep predicate can be utilised by Mephisto reasoners for logical inference. The formation of a variety of other spatial predicates follows this principle.

Syntax parser. This module defines and selects hand-crafted grammatically valid syntactic forms to ensure compliance with Lexpresso's controlled English syntax. Selected syntactic structures are described further in Section 3.

Lexicon. This module is used by the parser to instantiate leaf nodes. The lexicon covers core and domain specific terminology. It includes all major parts of speech containing over 20,000 unique word-forms comprised of a core of circa 12,130 high frequency English tokens plus non-high frequency general and domain specific terms. The class of nouns is comprised of circa 6,900 common nouns, circa 1,000 proper names and 62 forms of pronouns. Nouns are categorised according to certain features: mass/count, number, gender, alienability, and possible syntactic dependencies.

The class of verbs is comprised of main verbs, auxiliaries and modal auxiliaries. There are circa 8,380 main verbs classified according to a number of features, including semantic type, agreement, tense, aspect and mood inflection, syntactic and semantic frames, e.g. [11, 12, 13].

The class of adjectives is comprised of circa 2,644 forms and categorised into attributive, predicative, comparative and superlative types. Adjectives and other modifiers are further categorised according to the following primarily semantic types: age, amplifier, century, colour, compass, denominal, evaluative, girth, height, noun, objective, ordinal, participle, provenance, religion, shape, size, subjective, weak. This classification is used to stipulate the order adjectives in noun phrases.

Other classes of words consist of articles, cardinal and ordinal numerals, prepositions and other forms such as conjunctions, *wh*-words and directionals. Finally, there is a special sub-lexicon of domain specific expanded acronyms containing over 41,000 entries.

The Lexicon and Parser draw on syntactic and semantic knowledge (including *Aktionsart* [14], entity functional capability, and taxonomic relations) to constrain possible interpretations and reduce over-generation of deep CNL structures.

At this point in the system architecture linguistic content moves from surface to deep Lexpresso modules.

Grapher & Ambiguity handler. The Grapher transmutes parser outputs into graph structures so associations and semantic structures can be more easily reviewed than with parser output. In cases of multiple interpretations, e.g. (3b), a separate module identifies user defined interpretation preferences. This selects top ranked interpretations in a given context. Where multiple interpretations are equally ranked a separate graph structure is generated for each. The user can compare and select the desired interpretation to ensure each semantic form

passed to the reasoner is unambiguous. Evaluation of this potentially burdensome method is required.

Semantic translator & Thematic roles. The graph structure is then converted into our universal semantic constructs for use by automated inferencing modules, not discussed here. Thematic roles and other linguistic information are identified by combining lexical and constructional semantics from the Syntax parser, Lexicon and Aktionsart modules with the Linguistic Knowledge Base. The results of this process ensure deep structures contain the requisite richness of linguistic semantic information for both inferencing and language generation. The Thematic roles module associates possible semantic roles with generic entities at the highest possible level in the Taxonomy. Subsumed entities will inherit the role associated with their genus.

Context resolver. Lexical semantic features associated with nouns (such as gender, animacy, cognitive or other capability) are identified via a relatively shallow hierarchy and used to resolve anaphoric pronouns and *wh*-forms. Types of NP anaphors include personal, reflexive, reciprocal and indefinite pronouns, as well as demonstratives. The only current VP anaphor is forms of the generic verb 'do'. A number of rules (not discussed here) determine how anaphors are resolved.

Where possible the CNL Effector (see Figure 1) makes use of existing Lexpresso modules as already described (such as the lexicon and parser and associated semantic knowledge) to handle the generation of surface language from deep Mephisto semantic constructs. Space constraints prohibit further explanation here.

3   Syntactic Structures

This section exemplifies selected basic syntactic structures permitted by the parser. Space constraints prohibit a comprehensive exposition of all of Lexpresso's syntax. The expository emphasis is on giving a sense of Lexpresso's expressiveness.

Sentence Types. These include declaratives, interrogatives, directives and indirect speech acts.

Declaratives. These sentence types include basic intransitives (3a), transitives (3b) and ditransitives (3c) with and without adjuncts.

(3)   a. The boy slept on Monday.

      b. The woman in the car read the message on the sign.

   c. The woman gave the man the document.

Interrogatives. These can query for a range of syntactic elements: the subject (4a), object (4b) or predicate, such as generic 'do', (4c) of the basic sentences, as well as temporal information (4d) and locational adjuncts (4e). Indefinite pronouns can be used to query for any argument; (4f) demonstrates its use in a yes/no query.

(4)   a. Who gave the document to the boy?
      b. What did the woman read?
      c. What did the boy do?
      d. When did she read it?
      e. What region is she in?
      f. Did anyone see the woman?

Directives. These are currently limited to commands to the system to generate situation reports on specified tracks monitored by a track sensor module (not discussed here), e.g. 'Show merchant ship situation report on MR41_PAN-EAV' and 'Show commercial aircraft situation report on NAT57_FL310'. The range of useful commands to the system will to a certain extent dictate development of other directives. Directives are queries expressed in the imperative mood.
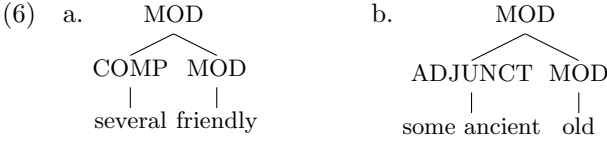
Indirect speech. These cover statements with embedded speech act verbs, such as 'say' and 'tell', e.g. 'Michael said that the woman read the document.' and 'Michael told Kerry that the woman read the document.' Subclauses introduced by 'that' are also permitted in other sentence types, not exemplified here.

Noun Phrases. These can be highly complex with multiple layers of embedding and recursion. The tree structures in (5) cover two primary basic types of noun phrase: (5a) specifies nouns with pre and post modifiers, whereas (5b) specifies conjoined noun phrases. Note its recursion.
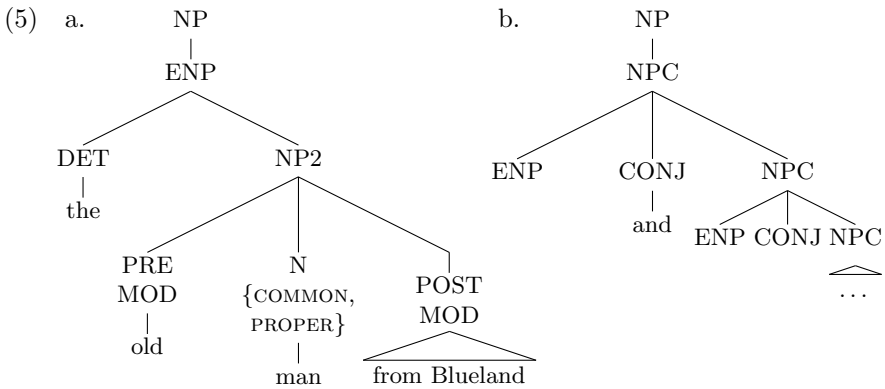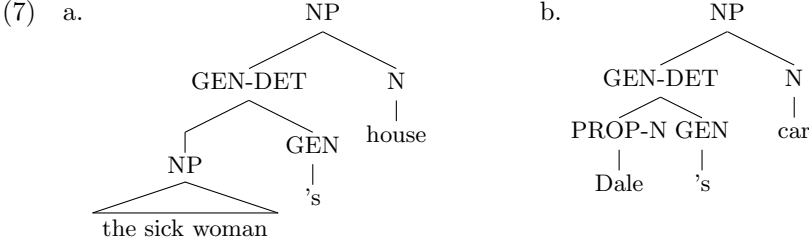

The modifier node is itself internally complex and permits modification by complements and adjuncts, as in (6). Justifications are not given here.

(6) a.
```
        MOD
       /\
  COMP   MOD
    |     |
several friendly
```

b.
```
          MOD
         /\
  ADJUNCT   MOD
     |       |
some ancient  old
```

Finally, the trees in (7) exemplify permitted noun phrases with genetive-*s*. Again note the recursion in (7a).

(7) a.
```
                 NP
                /\
         GEN-DET    N
         /\         |
       NP   GEN   house
      /\      |
the sick woman  's
```

b.
```
                  NP
                 /\
          GEN-DET    N
          /\         |
    PROP-N  GEN     car
      |      |
    Dale    's
```

4   Semantic Structures

Kuhn [15] identified some five expressiveness features of Controlled Natural Languages (see a–e in *Fairly high expressiveness* in Section 5 below). I briefly exemplify these with I and C forms for each semantic type.

Universal quantification over individuals. Instances of universally quantified entities without an article (8a) are rendered with an `all` predicate (8b) referencing its universally quantified Skolem constant together with the list of relevant linguistic features. Numerals are converted into set operations and can quantify all argument positions and predicates, e.g. 'Three men read four documents twice'.

(8) a. I: Women stand.
     b. C: `all([skc2],woman(@(skc2,t_3,s_2),[female,plural,...])`
            `=> stands(@(skc2,t_3,s_2),[general_habitual,...])).`

Binary or higher relations. In principle Lexpresso does not place a restriction on the arity of relations (`reads(x,y)` in (9b) exemplifies a binary predicate). However, our ability to reason with higher arity relations is determined by the reasoners used, not discussed here. Our reasoner does not restrict arity either and indeed allows atomic propositions to occur as relation arguments.

General rule structures. Multiple universal quantification can target all argument positions of relations, as in (9).

(9) a. I: All women always read all documents.
     b. C: `all([skc81,skc82,t_81],((woman(@(skc81,t_81,s_81),[...])&`
            `document(@(skc82,t_81,s_82),[...]))`
            `=> reads(@(skc81,t_81,s_81),@(skc82,t_81,s_82),[...]))).`

If–then conditionals are also expressible, currently with the form in (10). Note that all argument positions of these conditionals can be universally quantified, as in (10) with '`all`'.

(10)   a. I: If all women did not see the car then all women did not see the driver.

   b. C: `all([skc81],((woman(@(skc81,t_81,s_81),[...]) &`
      `car(@(skc82,t_81,s_82),[...]))) =>`
      `¬ sees(@(skc81,t_81,s_81),@(skc82,t_81,s_82)))) =>`
      `all([skc81], ((woman(@(skc81,t_81,s_81),[...]) &`
      `driver(@(skc84,t_81,s_84),[...]))) =>`
      `¬ sees(@(skc81,t_81,s_81),@(skc84,t_81,s_84),[...]))).`

Negation. Weak negation is expressed with a negation operator appended to the front of the negated predicate, as in (11). This can be applied to any proposition. Strong negation is expressed via lexical negators such as 'dislike', 'distrust', etc.

(11)   a. I: The woman did not read the document.

   b. C: `woman(@(skc81,t_22,s_81),[definite,...]),`
      `document(@(skc07,t_22,s_07),[definite,...]),`
      `¬reads(@(skc81,t_22,s_81),@(skc07,t_22,s_07),[past,...]).`

Second-order universal quantification. This was exemplified in (9) in which the predicate 'read' is universally quantified by 'always' and rendered as an operator over the time of the predicate.

Other features articulated by Kuhn as determinants of expressiveness were existential quantification, as in (2), equality, as in (12), and types of speech acts (not exemplified due to space constraints but mentioned in Section 3).

(12)   a. I: Andrew White is the Prime Minister.

   b. C: `Andrew_White(@(skc6,t_10,s_6),[...]),`
      `prime_minister(@(skc7,t_10,s_7),[...]),`
      `identical[@(skc6,t_10,s_6),@(skc7,t_10,s_7)].`

Discourse structures. The paragraph is taken as the basic unit of discourse. For the purpose of anaphoric resolution, a new paragraph signifies a new discourse context. A single sentence can constitute a paragraph. Anaphora occurs within a discourse unit.

5   Classification of Lexpresso

Kuhn [15] presented a classification scheme for CNLs labelled with the acronym PENS. This classifies a CNL according to a five-tier ranking (with 5 for maximal) for each of four orthogonal categories of Precision, Expressiveness, Naturalness and Simplicity. Based on my assessment of Lexpresso against Kuhn's 'PENS' scheme, I tentatively classify it as P^{3-4} E^4 N^{4-5} S^3 as evidenced by the following paragraphs.

Precision—reliably & semi-deterministically interpretable. P^{3-4} Although it is not currently possible for any natural English language text to be deterministically transformed by Lexpresso into a formal logic representation, the syntax is heavily restricted enough to make automatic interpretation reliable. In cases where the natural language syntax is ambiguous and not automatically disambiguated, multiple interpretations are presented. The user is then consulted to select the desired interpretation. Once done, controlled natural language is deterministically translated into formal structures. There is also a well established, conceptually broad, underlying formalism.

Fairly high expressiveness. E^4 The range of propositions that Lexpresso can express includes all those articulated by Kuhn: (a) universal quantification over individuals; (b) relations of arity greater than 1; (c) general rule structures (if-then conditionals with multiple universal quantification that can target all argument positions of relations); (d) negation (strong negation or negation as failure); (e) general second-order universal quantification over concepts and relations; (f) existential quantification; (g) equality; and (h) types of speech acts including declarative, interrogative, directive and indirect. See Section 3 for examples.

Fair degree of naturalness. N^{4-5} While large scale texts have not been written in the language, small fairly natural texts and spoken interactions with internal interdependencies are parsable.

Simplicity. S^3 Lexpresso can be exactly, comprehensively defined with accepted grammatical and logical notations but it is likely to require more than ten pages to describe all its syntactic and semantic properties.[8]

6   Conclusion

This brief introduction to the Controlled Natural Language—Lexpresso—has presented the system architecture, and exemplified its main syntactic and semantic features. These features have been compared to Kuhn's [15] PENS classification system. Against this comparison I have tentatively classified Lexpresso as a P^{3-4} E^4 N^{4-5} S^3 CNL. According to this classification Lexpresso is a reliably or perhaps deterministically interpretable language, with high expressiveness, considerable naturalness and would require a lengthy treatment to cover its syntax and semantics. Given the page limit, this paper has not in any detail discussed Lexpresso's limitations nor the tight integration with the knowledge representation and reasoning capabilities which constitutes a significant component of our high-level information fusion system for which Lexpresso functions as a natural language interface.

---

[8] The description of Lexpresso's features presented here is not considered comprehensive and therefore does not qualify as an indicator of its simplicity score.

References

1. Lambert, D., Nowak, C.: The mephisto conceptual framework. Technical Report DSTO-TR-2162, Defence Science and Technology Organisation (2008)
2. Baader, F., et al.: A novel architecture for situation awareness systems. In: Giese, M., Waaler, A. (eds.) TABLEAUX 2009. LNCS, vol. 5607, pp. 77–92. Springer, Heidelberg (2009)
3. Schwitter, R., Tilbrook, M.: Dynamic semantics at work. In: Proceedings of the International Workshop on Logic and Engineering of Natural Language Semantics (in conjunction with the 18th Annual Conference of the Japanese Society for Artificial Intelligence), Kanazawa, Japan, pp. 49–60 (2004)
4. Schwitter, R., Tilbrook, M.: Processable english (2007)
5. Lambert, D., Blasch, E., Bossé., E.: Introduction. In: Blasch, E., Bossé, E., Lambert, D. (eds.) High-Level Information Fusion Management and Systems Design, pp. 173–190. Artech House (2012)
6. Chomsky, N.: Syntactic Structures. Mouton & Co., 's-Gravenhage (1957)
7. Chomsky, N.: Aspects of the theory of syntax. MIT Press (1965)
8. Lambert, D., Lambert, A.: The legal agreement protocol. In: Blasch, E., Bossé, E., Lambert, D. (eds.) High-Level Information Fusion Management and Systems Design, pp. 173–190. Artech House (2012)
9. Allen, J.: Maintaining knowledge about temporal intervals. Commun. ACM 26, 832–843 (1983)
10. Saulwick, A.: Spatiotemporal reasoning over natural language. In: CoSLI-3 3rd Workshop on Computational Models of Spatial Language Interpretation and Generation (2013)
11. Fillmore, C.: Frame semantics and the nature of language. In: Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech, vol. 280, pp. 20–32 (1976)
12. Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C., Scheffczyk, J.: FrameNet II: Extended Theory and Practice. International Computer Science Institute, Berkeley, California (2006) (Distributed with the FrameNet data)
13. Fillmore, C., Lee-Goldman, R., Rhomieux, R.: The framenet constructicon. In: Boas, H., Sag, I. (eds.) Sign-Based Construction Grammar. CSLI Lecture Notes (2012)
14. Vendler, Z.: Verbs and times. In: Linguistics in Philosophy. Cornell University Press, Ithaca (1967)
15. Kuhn, T.: A survey and classification of controlled natural languages. Computational Linguistics, 121–170 (2013)