doi:10.2498/cit.1002278

A Semiotic Analysis of *If we are holding hands, whose hand am I holding?*

Martin Wheatman
Withinreap Barn, Moss Side Lane, Preston, United Kingdom

This paper presents a system which asserts understanding of complex utterances by semiotic analysis. The question in the title was chosen because: it enquires about a conceptual state of affairs, referring to in two ways; and reaches an answer by subtracting *I* from *we* to get the unspoken *you*. The system introduces *concepts*, that about which we can speak, defined by *repertoires* in natural language, including phrasal analysis patterns. The repertoire *conceptualization* grounds it on a state of affairs against which reasoning can be performed. On matching an utterance, associated intentions are internally vocalized; thus, concepts are informed by more specific concepts. The repertoire *conceptualism* allows hypothetical states of affairs to be constructed, such as in the title question.

Keywords: semiotics, utterance, linguistic information system, machine understanding

1. Introduction

The embodied approach to intelligent systems has received much interest in recent years (Goth 2011, Pfeifer *et al* 2012); hardly a month goes by without reference to driver-less cars, mechanical bipedalism or empathetic automata. Their advantage over disembodied systems is that they supposedly have a grounding in the real world, enabling them to interact more successfully with that environment (Steels 2008); however, disembodied problems, like understanding the title question, remain. In a world of mechanical devices, if such a device learns to hold hands and is aware of a hand holding situation, it may still fail to comprehend the title question since it is possible that we cannot hold hands: perhaps you are a double amputee, or we have only ever met over the Internet. The same problem, if at a different scale, is found with systems that attempt to model the neural structure of the brain: it builds a phenomenal truth, not a conceptual truth. Further, embodied approaches remain disparate systems – the notion of computer as universal machine requires the use of the universal medium: language.

The interest in embodied systems is fuelled by a failure of disembodied systems to make inroads into understanding. Systems such as ELIZA (Weizenbaum, 1966) and typical chatbot entrants to the Turing Test (Loebner, 2013), at best only have a shallow understanding and at worst are positively evasive. Since this project started, Apple have released SIRI on their mobile platform (Apple, 2013), a natural language understanding tool, but one which does not understand the title question; further, Google has announced Hummingbird (Google, 2013), a concept based approach to search. The web search approach (IBM, 2012) will presumably fail if no answer can be found on the Internet. No answer is defined in this paper; indeed, there is no definitive answer to this context-sensitive question, although one is given in the example. Even systems which are useful, such as booking system (VocalIQ, 2014), or banking Interactive Voice Response systems (PhoneBank, 2014), are dedicated to a particular task, so a more complex comprehension, as sought by this paper, cannot be ascertained. Perhaps the most remarkable system in this area, since the dawn of computing, is SHRDLU (Winograd 1972) which makes sense of a limited world of blocks. This project, however, deals with *signs*, metaphors addressing the real world, rather than blocks.

The simplest sign model is of a sign, or *signifier*, which forms a *signified* mental concept in the mind (Saussure, 1915). This paper fol-

The first and abridged version of this paper appeared originally in LISS 2013, Reading, UK, August 21–24, 2013. Published in partnership with ICIR of Beijing Jiaotong University.

lows the more comprehensive sign model of Charles Sanders Peirce, described in Section 2. Zemanek made the first connection between Information Systems and Semiotics (Zemanek 1966), an approach taken more fully by Eco (Eco, 1976). Semiotics was also used by Stamper in describing Business Information Systems (Stamper, 1973). Further, he modelled information as a triadic sign (Stamper 1985). Since then the various approaches to Signs in IS has been directed at the design of IS and their interaction with society, in what is more commonly known as Organizational Semiotics (Andersen, 1997; Stamper, 2001; Liu, 2000). A return to the semiotics of source code as a system of textual signs was presented in (Tanaka-Ishii, 2010).

This paper's novel approach applies semiotics to natural language understanding, by modelling in software the signs that represent utterances. It shows that complex utterances can be analyzed by Semiotics, using phrases which inform utterance interpretation, such as: *X means Y*; and, *X implies Y*. It models the cognitive processes as *saying*, *doing* and *thinking*. This paper presents: a summary of the semiotics used and introduced; the required semiotic analysis of the title; a mechanism for representing textual signs; the repertoires needed to analyze the title question; an example session; and finally a brief discussion of its implications.

2. Semiotic Terminology

This section highlights semiotics which has inspired the approach in this paper. Mostly, it describes Peirce's Semiotic; it is a sprint through his work, which evolved from 1867 through to 1910. It also mentions Austin's critique of traditional linguistic structure, and introduces *concept*.

This paper draws directly on Peirce's triadic Semiotics (Peirce, 1935-57, §2.227-306). In essence, the connection between the *representamen*, or that which acts as the sign, and its referent *object* (in this case the utterance and hand holding state-of-affairs) occurs only through the mediation of *interpretant* – the cognitive artifacts created by the act of interpretation, or *semiosis*, which cannot be shared other than by further representamen and interpretant. Interpretant serves as representamen in further, more developed, signs. This is sometimes referred to as: *denotation*, the first meaning; and, *connotation*, subsequent, or developed meanings (Hjelmslev, 1961). The sign triad has been illustrated in (Ogden and Richards, 1923). The relationship between the Saussurean and Peircean models is complex, and a mapping between the two is given in (Tanaka-Ishii, 2010).

Peirce classified each triad component in three types, each given here in ascending complexity. A representamen can be: a *qualisign*, or individual quality of a sign; a *sinsign*, or group of qualisigns as a single element; or *legisign*, or rule governing the use of signs. The manner in which a representamen refers to its object can be: *iconic*, or referring to some physical quality (color, shape, taste, smell etc); *indexical*, referring to some pointer (a weather cock pointing to the direction of the wind, or a smooth stone pointing to the action of a river); or *symbolic*, a sign referring by convention. An interpretant can be: a *rheme*, a timeless sign; a *dicent*, a historical aspect; or an *argument*, a hypothetical sign. He also writes of the ten types of Sign, being selected from these three classifications values. Being written in the present tense, the title question could be seen as a *rheme-symbol-legisign*; however, with the first clause defining a hyperthetical situation, it is argued that the title question is the most complex sign: *argument-symbol-legisign*.

Peirce also talks of *ground*, what we would call the medium in which sign systems function. In this case, the sign system is grounded in natural language, or utterance.

Semiotics also includes the construction and use (and the consequences of repeated use) of sign: Pragmatics. Morris proposed a model of semiotics based on the linguistic principles of *syntax*, *semantics* and *pragmatics* (Morris, 1938) This ties in the traditional linguistic structuring with pragmatics, which has been a burden ever since. Meaning cannot be sought from structure alone – *the plane has landed* and *the Eagle has landed* have the same structure but vastly different meanings. The insertion of pragmatics – the use of signs – does not counteract the use of syntax, and its notion that individual words have means, and this approach is not followed. Peirce defined three divisions in the development of interpretant as: *immediate*, the initial, and not necessarily correct, reaction; *dynamic*, the correct reaction to, or functioning of, the sign; and, *final*, the habits formed by use of the sign; e.g. a phrased question, *what is your X?*, will have a defined reply framed as *my X is...*

This paper also draws inspiration from Austin's criticism of traditional linguistic methods of defining language, given in a series of lectures at Harvard (Austin 1962). Words are not labels for things, only utterances can make sense in their entirety. Austin's model of: *locution*, speech; *illocution*, action; and *perlocution*, reaction (Austin, 1962 pp. 94–108), is essentially a triadic model directed towards speech acts: a given phrase uttered by someone, under certain conditions, elicits some felicitous response.

Further, conversational interfaces draw on the Peircean notion of Quasi-mind, as modelled in (Wheatman, 2011), by a pair of signs arranged in opposition. The representamen of one being the object of the other which illustrates the essence of an information system as a balancing of interpretant in a process of mirrored denotation. This models information systems as diverse as the request/response conversation of the HTTP protocol, to the genotype/phenotype balance of evolutionary biology.

The construction of Signs, and thus the construction of Concepts, is an example of the notion of *autopoiesis*, (Maturana and Varela, 1980). This is the ability of a system to self-create, as compared with the creation of a material thing: *allopoiesis*. The traditional example is that organisms create organisms, whereas a car factory is needed to produce cars. Information Systems often have an autopoietic nature: a compiler is usually written in the language it compiles. There is the temptation to class autopoiesis in terms of a meta-language; however, this the language of allopoiesis, since it would be a different one to the one it describes. While the autopoiesis repertoire is built into the language engine, it is not directly used in the deconstruction of the title question and is not documented in detail in this paper.

Finally, this paper introduces Concept as a complete group of cooperating signs. Peirce merely notes that the construction of signs, and their use and the consequences of repeated use as Pragmaticism (Peirce, 1935-57 §5.402), are so named to distinguish his notion from that of other pragmatists (Thayer, 1982). The notion here is that different Representamen, which refer to the same Object through a different Interpretant, form a repertoire about that Object. Where that repertoire is deemed to be closed, or *complete*, they form a Concept. A simple, but complete, repertoire may be the concept of name: *my name is X., my name is not X., what is my name?* This is described below in Section 4.

3. Semiotic Analysis of *If we are holding hands...*

This section describes the semiotic analysis that is required by this paper. The difficulty with the title question is that there are different valid responses to this. Further, the answer required can be expressed in many ways; however, it is not the intention to define one here.

The title question is an analytic one: the answer can be found within the question, rather than in referring to known data, or a goal to seek. The first sign to interpret, therefore, is the question in its entirety. Being short and easily quotable, it can act as an example of the type of questions disembodied intelligence programs have difficulty in answering. As such, it is an atomic phrase which is merely in need of recognition that it has been uttered: *precisely*. However, the intention in this paper is to show how it can be deconstructed to arrive at a logical answer.

In doing so, the first thing to notice is the word *if*. From a Computer Science perspective, the token **if**, in programming languages, is a division in the path of a process to execute either a **then** or an **else** clause, dependent on the value, zero or non-zero, of the following condition. In this case however, *If...* paints a picture of a hypothetical situation: it says, *Once upon a time...* The corollary in computing should not be the branches of a conditional statement, but of two branches in a change management system, where two realities co-exist, until release by the corollary *we are not holding hands*.

Such *conceptualism* allows the deconstruction of the question into the two utterances, *we are holding hands* and *whose hand am I holding?* This assertion and question will now be dealt with in turn, each accessing the conceptualization of the holding of hands through separate repertoires.

The first sentence uses the phrase *holding hands*, which here is atomic as it could be used in sentences such as *we are X* – where X could be laughing or crying or, as in this case, holding hands. In the example given below, it is deemed to be a special case – as an example of how concepts are informed by concepts. Indeed, it could be further deconstructed: we may be holding left and right hands while walking together, or holding both left/right and left/right in a two person "ring'o'roses" circle, or even left/left and right/right in a figure-of-eight – the configuration is not specified: we are, for the moment, merely performing the social construct of holding hands.

The second sentence is a general *doing-to* query, such as *whose M is P N?* where *M* and *N* could be car and driving, dog and walking, or in this case hand and holding. This is a general *entity's action to another entity's attribute* relationship – the meaning in each case is effectively the same, other than further implications provided by the particular entity or action. This is a class of sentence, or a pattern, which can be applied to several (many) instances.

The holding of hands can be conceptualized in a manner which can be used to represent both, illustrated below. The former atomic phrase can be represented by the latter's composite doing-to terms: if *I am holding your hand* and *you are holding my hand*, then *we are holding hands*. Consequently, if we are holding hands, then you are holding my hand *and* I am holding your hand – it is a consensual act. Conversely, if we are not holding hands, either I am not holding your hand and/or you are not holding my hand.

Finally, the answer will be one of many expressions representing a single concept. The exact one chosen by the respondent is of no consequence and one is not defined in this paper; but, it will be aligned to their motive and ability. It is also worth noting here that we are effectively playing the Imitation Game (Turing 1950), as one is now offering a hypothetical state of affairs, rather than operating the Turing Test, as typically framed (Loebner 2013), of merely generating conversation, which can come across as evasive.

4. Semiotic Mechanism

This section describes the utterance processing mechanism in Peircean terms. The software architecture consists of a language engine, which is configured by a set of text files, known as *repertoires*. These describe the various interactions with a concept. A simple, but complete, repertoire may be the concept of name: *my name is X., my name is not X., what is my name?* Repertoire is constructed through its own repertoire of *autopoiesis*, which is built into the engine.

4.1. Sign

A list of signs is built up inside the engine from their repertoire descriptions. The process of semiosis amounts to the searching of this list for an appropriate match and then interpreting the associated intentions until a felicitous outcome is achieved. The felicitous nature of the outcome is maintained throughout the processing of intentions, which is used to determine the way in which the interpretant is to be further used in processing that utterance, see 4.3. The matching of an utterance against these signs continues until one is found or all are exhausted: if none are matched or those found are inappropriate, the utterance is deemed not understood. The default reply, which is independent of the utterance, is *I do not understand*, defined in a configuration file rather than being hard-wired English. All other replies are provided in the appropriate interpretant.

4.2. Utterance as Representamen

A representamen is identified by matching the utterance against the list of signs. It is defined by a pattern, or *phrasal analysis string*, defining that which can be said e.g. *my name is X.*, whereby upper-case letters (and words) denote hot-spots, or variables, and lower-case letters denote boilerplate, or a constant text value. It is assumed that this form is commonly understood. The value for each variable is assigned from the given utterance.

Pattern matching searches all examples and returns the match with the least amount of variables, to return the most specialized sentence –

so the utterance *the Eagle has landed* matches *the eagle has landed* before *the X has landed*, where $X =$"eagle". Further disambiguation can be obtained by querying object space to determine in what context an utterance is given. Pattern matching here requires an exact match. Unlike modern search methods, which usually remove small words from search terms, contextual words provide useful boilerplate for pattern matching. Currently, one token in the pattern is matched with one token in the utterance, unless it is a hot-spot prefixed by PHRASE-.

4.3. Rules of Natural Language Interpretation as Interpretant

Interpretation of interpretant occurs once an utterance is matched with a pattern: the intentions of the matched pattern are applied in order. Interpretant is represented as an ordered bag (i.e. a non-unique list) of attributes, e.g. *name="value"*, or *intentions*. In that the cognitive process is being modelled, there is a broad intention classification of *thinking*, *doing* and *saying*. In the current prototype there are merely seven intentions: *think*, *perform*, and *reply*; their infelicitous corollaries *elseThink*, *elsePerform* and *elseReply*; and, *finally*. The felicitous outcome of each intention is maintained, and subsequent intentions are applied appropriately. The notion of an answer is also maintained from a previous intention, to be inserted into the appropriate reply, replacing an ellipsis. For example, a reply might have the utterance *"your name is..."* The seven intentions can be described in the following three groups.

The content of the *think* intention, and its corollary *elseThink* is, privately, and silently, re-uttered as in the notion of internal thought (Murdoch 1970). Thus thinking is modeled as a linguistic process, and the interpretant can thus act as representamen in further sign triads. Thinking propagates the felicity of outcome.

The *perform* intention and its corollary *elsePerform*, interacts with persistent object space; thus, the format of its content echoes the format of a script, where hot-spots are represented as dollar prefixed variables. The output from a *perform* supplies the answer to be replied, and sets the felicity of outcome. A *finally* intention, which is a *perform/elsePerform* intention, is applied regardless of the felicitous outcome.

The *reply* intention, and its corollary *elseReply*, formats a reply with the given answer and its content, a format composed of text and various object references. When a reply is applied, it is seen as a *got it* moment, and all other intentions, other than the *finally* intention, are skipped. Replying propagates the felicity of outcome, unless it is a (defined) positive reply, *I know* to a negative outcome.

4.4. A State of Affairs as Object

A *state-of-affairs* is composed of a set of persistent non-linear name/value pairs, or attributes. These are name/value pairs and links between a name and another name/value pair. Because these are values and not signs, they can be reasoned with, or processed, in much the same way as can database values. An attribute represents the basic Saussurean sign pair of Representamen/Immediate Object. The name is structured around an object model.

Inspiration for this has been taken from Wittgenstein's Tractatus (Wittgenstein 1922), and is represented by a three-tiered overlaid entity space, entities; and attributes of values and links between entities. Overlays represent some level of reality, and in much the same way as an overhead projector slide can be overlaid and modified, while leaving the underlying slide inviolate; or, as a configuration management system, e.g. Clearcase (IBM 2013), represents the various states of a multi-file development. Each overlay contains environment variables and entities which in turn contains sub-entities, text values, or entity valued attributes (links to other entities or sub-entities). In that underlying overlays remain inviolate, there is the need to provide additive-deletions, a placeholder, in the top overlay to delete entities, which is analogous to scribbling over an entity in the top overlay only. A similar mechanism is required for entity renaming – a placeholder is provided in the top overlay to present a new name, and to cover up the old one. Some mechanism for reconciling layers is possible, and desirable, but not a yet implemented nor a priority.

The *conceptualization* repertoire is used to interaction with this state-of-affairs allowing access via a command line interface. Return values are: zero, representing false or error; non-zero, representing true; or, a pointer to a string. Conceptualization scripts are specified in the *perform* and *finally* intentions. Object references here are represented by dollar-preceded lower case forms, emphasizing its scripting heritage. It is from this conceptualization that answers are obtained, which are then formatted by a following *reply* intention. In order to insulate the user from having to write such scripts, there is an interface to this object component written using the think intention, and hence in natural language. This is encapsulated within the concept of *conceptualization* (concepts are introduced below).

The desire for a simple conceptualization model over an enriched one means the underlying entity model does not directly support the Object-Oriented model of classes, instances, inheritance and such like. The model contains entities, attributes (values or links) and a mechanism for transitive links, from which an object model is created by the concepts of *has* and *isa*. These repertoires are detailed further in Section 5, although they are not directly used in the deconstruction of the title question.

4.5. Concept – a Group of Cooperating Signs

This project uses the notion of concept as a complete group of cooperating signs, or *repertoires*. A concept may effectively be merely a sinsign; however, working at a conventional level, it would have to be a legisign. A portion of a state-of-affairs is managed through a repertoire, much in the same way a programming language class is a set of members managed through an interface. The concept *relative-to* is managed by being able to say, e.g., *X is REL than Y*; or, *is X REL than Y?* where REL is larger, taller, heavier, or whatever other relative value introduced by further repertoire. The concepts required to perform the analysis of the title question are described in Section 5.

5. Repertoires

The question has been analyzed, satisfactorily, into an assertion and a question within a conceptual framework; the repertoires to exploit this analysis are now described. For brevity, the given examples are the phrasal analysis patterns extracted from the full repertoires, apart from the last which is the full repertoire. It is assumed, also for brevity, that the *conceptualization* concept is defined elsewhere, constructing the repertoire of object space, e.g. *X exists! delete X X exists?* etc. to encapsulate (most) *perform* intentions. There follow descriptions of the general concepts of: *isa* and *has* constructing the repertoire of the object model – both using this conceptualization; and then *action*, which is used to implement concept of *holding hands*. Both of these are then used within a framework provided by the concept of *conceptualism*. Finally, there is an example session presented showing the resolution of the title question.

5.1. The Isa and Has Repertoires

A simplistic object model is represented by the *isa* and *has* repertoires, both making use of the *conceptualization* repertoire. This proves useful in modeling real-world objects, as it abstracts away the need to know of implementation details. The semiotic approach can be described as being itself in some way polymorphic, since the actions associated with an utterance are connotative: they are connected, or linked, through a third connection, rather than being denoted, or directly linked. The isa repertoire is as follows:

CLASS is a class.

is ENTITY a class?

what is a CLASS?

what is a class?

a DERIVED is a BASE.

is a DERIVED a BASE?

OBJECT is a CLASS.

is OBJECT a CLASS?

what is OBJECT?

The *isa* repertoire is reasonably straightforward, given that the presence, or absence, of the boilerplate *a* prior to a hot-spot points to it being a concrete object or an abstract object, or class, respectively. The first six utterances manage the class network by simple accumulation.

The *what is a class?* utterance is complete boilerplate giving the ability to provide an explanation of the meta-class *class*. Class is not itself a class, it is the meta-class, the class of classes. As such, the utterance *a person* specifies any of the entities which inherit from the class person, whereas *a class* does not specify all entities which inherit from class. Consider the utterance *a person has a name* meaning that the class person is attributed with a name value, whereas *a class has a name* does not have the same meaning – which would mean that all entities have a name. The last three utterances manage instances of classes.

The *has* repertoire is as follows:

a CLASS has a MEMBER.

can a CLASS have a MEMBER?

Does OBJECT have a MEMBER?

is ENTITY's MEMBER VALUE?

ENTITY's MEMBER is VALUE.

ENTITY's MEMBER is QUOTED-VALUE.

what is ENTITY's MEMBER?

ENTITY's MEMBER's ATTRIBUTE is VALUE.

what MEMBER is ENTITY's OBJECT?

The first three utterances deal with the attribution of members to classes. The remainder deals with assigning values to entity's members in various forms. This may be incomplete given the many ways values can be assigned; the last two, for example, deal with the dereferencing entities, such as in the utterance *Freddie's car's color is red*. While more work is due here, the repertoire is sufficient for the title question as it stands.

5.2. The Action Repertoire

The first utterance in the *action* repertoire is an utterance required for the concept, which is matched and conceptualized by the *isa* concept as described above:

Action is a class.

Then, for the purposes of this paper, there is a small set of multi-intention patterns created here at load time by autopoietic utterances. Further deconstruction may show that this concept can be split into a further base/derived pairs; however, for the purpose of this paper, this repertoire is complete.

X is not DOING anything.

X is DOING Y's OBJECT.

X is not DOING Y's OBJECT.

Whose OBJECT is X DOING?

What is X DOING?

Is X DOING Y's OBJECT?

Looking at the first of these in detail, as an example, it is constructed by the three autopoietic utterances:

On "X is not DOING anything.", perform "link destroy $x $doing".

Then on "X is not DOING anything.", if not, reply "I cannot destroy a link between X and DOING".

Then on "X is not DOING anything.", reply "ok".

"X is not DOING anything." implies "is DOING an action?"; if not, reply "DOING is not an action".

"X is not DOING anything." implies "is X a person?"; if not, reply "X is not a person".

This shows one advantage of such an approach; while complex, these utterances read as natural English – much in the same way as COBOL is created as a subset of English: not everyone will be able to create these patterns, but most should be able to read them.

Once these concept construction utterances have been issued, on being presented with *Fred is not holding anything*, the analysis proceeds as follows. First, the interpreter checks that *Fred* is a known person – the *implies* intentions are added to the front of the list. If this is felicitous, then this is followed by a check to show that the hotspot *holding* is an action. Both of these are relying on a concept of *isa* and ultimately on the contextual state-of-affairs.

Finally, if these implications are both felicitous, the conceptual state of affairs is affected to represent the message within *Fred is not holding anything*. In this case (as it is implemented in the current prototype) this is the command to remove the *holding* link from the entity *fred*.

5.3. The Holding Hands Repertoire

In specifying the concept of *holding hands*, the first two utterances, interpreted at load time, define some necessary context for the concept of holding hands:

Holding is an action.

A person has a hand.

Note that, for the purpose of this paper, we are not concerned with the number of hands a person has. These are followed by five concept specific patterns which are utterances supported by the action concept. In brief:

X is holding hands with Y.

X is not holding hands with Y.

Is X holding hands with Y ?

We are holding hands.

We are not holding hands.

Finally, there is an equivalence intention, which strictly is not required for the given example, but completes the repertoire, being the inquisitive corollary to *we are holding hands* and *we are not holding hands*. In full, it is:

“Are we holding hands?” means “Are you holding hands with me?”.

5.4. The Conceptualism Repertoire

Finally, to fully address the title question, there is the concept of *conceptualism*. To resolve the title question, this repertoire combines the two sentences resolved from the initial semiotic analysis given in Section 3. It consists of a single utterance containing two phrases: *if X, Y?* Since it does not depend on these phrases, it can also be said to be polymorphic.

In a more comprehensive system, the value of X can be obtained to resolve ambiguity with utterances such as *if someone is greater than your eye-level, they are tall*, which serves as a rule to be applied to further entities.

The full repertoire is thus:

On “if PHRASE-X , PHRASE-Y?”, perform “overlay create”.

Then on “if PHRASE-X , PHRASE-Y?”, think “X”.

Then on “if PHRASE-X , PHRASE-Y?”, if not, reply “I do not understand: ’X’ ”.

Then on “if PHRASE-X , PHRASE-Y?”, think “Y?”.

Finally on “if PHRASE-X , PHRASE-Y?”, perform “overlay destroy”.

The initial *perform* creates an overlay within which to imagine the following conceptualizations. The pattern matching algorithm uses the phrase attribute, forced by the PHRASE- prefix, to look for a multi-word target, where they are terminated by the first token in the outer pattern’s prefix tokens, in this case ‘,’ and ‘?’ The full stop (period) in the *think* utterance is added by default; question and exclamation marks must be added in the second think explicitly. The *finally* intention destroys the overlay, leaving the original state undisturbed. This is illustrated in the following session.

5.5. A Resultant Session

This paper documents a second working prototype, originally (Wheatman, 2013), with improved Interpretant representation. The name, *enguage* is derived from lan*guage eng*ine, and has been developed and tested in Java, under Ubuntu 12.04. The output from a session follows. By arriving at one of many possible answers, it illustrates an understanding of the title question.

In this example, the user’s typing is highlighted in italic. The -v is the switch to enable verbose mode; English refers to a folder containing the concept repertoires: we may equally

```
martin@magenta64:~/src$ java -jar enguage.jar -v English
Enguage (c) Martin Wheatman, 2001-4, 2011-13
There are currently 6 concepts (two built-in and one blank)
Enguage main(): overlay is: [/home/martin/sofa, default(1)]
Enguage> whose hand am i holding?
You're not holding anyone's hand.
Enguage> if we are holding hands, whose hand am i holding?
You're holding my hand.
Enguage> whose hand am i holding?
You're not holding anyone's hand.
```

have Spanish, *On "¿Dónde está el OBJETO", think...* The first and third utterances are to show that the initial and final states of affairs remain unchanged due to the *finally* intention tearing down the overlay. The second utterance, the title question, builds this overlay in which to assert, or imagine, the hand-holding situation, and to interrogate this situation.

6. Discussion

This discussion attempts to draw out some of the implications of this work in relation to natural language processing and the practical application of this, and further, work.

Firstly, the goal of this project is to produce a passive linguistic information system – capable of deconstructing the title – rather than claiming to feel emotions or attempting to act intelligently. The analogy here being the loyal butler, Mr. Stevens (Ishiguro 1989), it follows a radical semiotic approach to language in that it is not the written words that contain meaning, but utterances in their entirety. This is radical in that it follows Peirce's general triadic model of signs, and does not use the traditional linguistic parlance of *syntax* and *semantics*. In this sense, an utterance is a sign in much the same way as any other sensory experience, if more complex and directed. Syntax, and by extension semantics, are language and therefore culturally dependent, whereas the mechanism developed here is independent of a given language. Further, meaning is arbitrary and based on subjective interpretant: two similar phrases can have widely different meanings, such as in *the plane has landed*, a mundane and everyday occurrence, and *the Eagle has landed*, an iconic moment of the 20th century. This approach appears to be grasped by other works (Ogden and Richard, 1923), and also in the dyadic model (Saussure, 1915) in that words are not merely labels for things. It is worth noting that the terms Syntax and Semantics were introduced to semiotics (Morris, 1938), by aligning their use with Pragmatics as a third level – the practical use of Syntax and Semantics. This is not to denigrate Morris's work entirely, people who have extended Morris's work certainly have grasped this notion, especially (Stamper, 1985): Morris uses these words in a semiotic context. A further critique of traditional approaches to language (Austin, 1962), which refutes the use of the parlance of word types, such as *nouns* and *verbs*, arrives at a similar triadic model of language based on language as action. A final note on interpretant, the repertoire, as used in this paper, can be defined as *final* interpretant as it represents habit – the normal way to do things. Moving back into *dynamic* and *immediate* interpretant suggests working with the development of new language and new habit. This is left to further research.

There is a marked difference in this approach to that taken by ELIZA (Weizenbaum, 1966), and thus many natural language processing programs produced for (Loebner, 2013), whereby output is generated from input, although there is an attempt to determine output without the generation of a formal syntax tree. While it does not formally define a sentence structure, ELIZA effectively uses the structure of the utterance in turning it around for output. Thus, *I am feeling X* becomes *why are you feeling X*. The language engine, through its interpretant, has access to a rudimentary object database, and so an ability to understand, in some respect, and to reason with an internal real-world model. Thus it also has the ability to specify further questions, based on input, which can eventually be answered by more rudimentary understanding and therefore an ability to deconstruct. While it is entirely possible that commercial IVR systems could be programmed to deal with such complexity, none are. It is presumed that since they interact with databases of real-world data, they will have some of the features described in this paper, but will not be described in semiotic terms.

While it is not described in detail in this paper, this system is autopoietic since repertoires are constructed from an in-built repertoire of thirteen utterances. Because of its autopoietic nature, the language can be enhanced and extended, the loading and unloading of the appropriate repertoires gives room for development. Further, such additions are not saved to persistent store – so on re-running the software any additional repertoire extentions will need to be reloaded. There is not a great amount of work involved in persisting added language, since the shape of a repertoire is similar to the nature of states of affairs – objects and their attributes.

One possible downside of the closed approach is that the language is seen as limited – limited to the language defined within the repertoires: it is not a work of arbitrary language interpretation. However, this is not a great problem, it is simply a sign that this is a constructive approach: as language develops in a child, or a non-native speaker, it develops through being useful – there is no need for a full understanding of a given language before portions of it work. Indeed, taken to its extreme, everyone's language can be regarded as limited by their own experience, and the language specifications begun here are the first tentative steps. For example, one system knowing of location need not know how to converse on relative height. One mark of the utility of this work is if a single concept repertoire works, which could be seen as the basic autopoiesis repertoire upon which all other repertoires are defined. Further, this closed nature of repertoires fit the nature of work situations, of non-social verbal interactions – of information exchange. This can be seen in the manner in which Mr. Stevens (Ishiguro, 1989) is of use through verbal interaction, but does not respond to arbitrary language – especially the scene where the butler is asked to comment on international politics. Indeed, as a freelance Software Engineer the author's career has largely depended on the learning of the closed repertoires of different clients' industries, and being able to conceptualize this in source code. While this claim of a closed nature is intended to cover information systems and not real-world work interactions, such as the garage workshop example (Andersen, 1997), this could even be seen as possible, given an interface to some armed *mechanic* and the repertoire of, for example, moving and item through 3D-space: *to you*, *to me, up a bit*. Much in the same way as an information system could be taught to record phenomenal experience, without access to that experience; what is needed here, however, is trust. Another area for further research is the ability of this system to cope with longer, more intricate, utterances: through the ability to deconstruct utterances into simpler forms. One application of this is the use of descriptive words to identify an entity, such as the *tall* boy, rather than an entity identifier (note the lexicon of *noun*, *adjectives* and *noun-phrase* is not used here). The description is as dependent on the entity, as the use of an entity identifier within an utterance. Can a repertoire be developed to deconstruct sentences?

Analytic questions are of little use; certainly, they are trivial for the human. This paper has shown, however, the solving of analytic questions, certainly of the form *if... then...?*, is also straightforward in software. So now a small closed repertoire example of this language engine is being sought as proof of its wider application. The one proposed is that of a shopping list – with a simple repertoire of *I need X*, *I have X* and *what do I need.* It is envisaged that an individual will nominate things required by *I need X*: thus a list is composed. On shopping, the individual can ask *what do I need?* and can obtain these items. On acquiring these items, they can be removed from the list by uttering *we have X*: thus the list is managed and the functionality of data cumulation of a notepad application is surpassed. Work on implementing this is currently underway.

Finally, a definition of the word *understanding*, in its linguistic context (with one caveat) can be asserted as the ability to conceptualize – to retain (remember and recall) and to reason through a given utterance. For example, we may be holding hands, and as such, it may be understood by this concept that I am holding your hand and you are holding my hand. However, if the fact is given that I am holding your hand but you are not holding my hand, we may not be holding hands: you may be holding a gun with which you are attempting to shoot me! Such subtlety in state-of-affairs is possible, and it is important to be able to understand, along with the ability to determine if something is not understood.

The caveat is that this is *adequate* understanding: Peirce notes (Peirce 1935-57) that the rapport between speaker and listener defines a Quasi-mind, and that understanding is judged by the extent to which this Quasi-mind works.

7. Conclusion

This paper demonstrates a mechanism which effectively comprehends a complex question, allowing a user, via a command line interface, to set up and interrogate a given state of affairs. Working on an analytic question may seem of little value, since they are trivial to comprehend;

it serves as a good depth test for this natural language understanding mechanism. One key feature, though not described in detail, is that language here is seen as self-describing. Further work, as described in the last section, involves utilizing the text-to-speech and speech-to-text facilities provided by mobile devices. The goals include: in the long term, an open ended interpretation of natural language; and the shorter term goals are creating a full vocal interface to a common application – a shopping list, and developing the management of repertoires. This work is applicable to human-computer interaction, especially with groups difficult to reach, such as the visually impaired, dyslexic, illiterate and the elderly.

References

[1] P. B. ANDERSEN, *A Theory of Computer Semiotics (Human-Computer Interaction, Vol. 3)*. Cambridge University Press, Cambridge, UK, 1997.

[2] APPLE, see `http://www.apple.com/uk/ios/siri/`, retrieved 2013.

[3] J. L. AUSTIN, *How to do Things with Words*, (Compiled by Urmson, J. O. and Marina Sbisà). Oxford Paperbacks, Oxford, 1962.

[4] U. ECO, *A Theory of Semiotics*. Indiana University Press, Bloomington, Indiana, 1976.

[5] GOOGLE, `http://www.scientificcomputing.com/news/2013/09/googles-hummingbird- hatches-new-search-formula#.UkmDSxBpeZQwww.ap.org`, retrieved 2013.

[6] G. GOTH, I, Domestic Robot. *Communications of the ACM*, **54**(5) (2011), pp. 16–17.

[7] IBM Introduction to *This is Watson*. *IBM Journal of Research and Development*, **56**(3.4) 2012.

[8] L. HJELMSLEV, *Prolegomena to a Theory of Language*, (F. Whitfield, transl.). University of Wisconsin Press.

[9] IBM, see: `www.ibm.com/software/products/us/en/clearcase`, retrieved 2013.

[10] K. ISHIGURO, *The Remains of the Day*. Faber and Faber, London, 1989.

[11] K. LIU, *Semiotics in Information Systems Engineering*. Cambridge University Press, Cambridge, UK, 2000.

[12] H. LOEBNER, `www.loebner.net/Prizef/loebner-prize.html?`, retrieved 2013.

[13] H. R. MATURANA, F. VARELA, Autopoiesis and Cognition: The Realization of the Living, (1980) pp. 12–123. Springer, Dordrecht, Holland.

[14] C. W. MORRIS, *Foundations of the Theory of Signs*. Encyclopaedia of Unified Science, 1(2), University of Chicago, Chicago, 1938.

[15] J. I. MURDOCH, *The Idea of Perfection*. In: The Sovereignty of Good, Routledge, London, (1970) pp. 1–44.

[16] C. K. OGDEN, I. A. RICHARDS, *The Meaning of Meaning*. Routledge & Kegan Paul, 1923.

[17] C. S. PEIRCE, Collected Papers of C. S. Peirce, Ed. Hartshorne, C. Weiss, P., Vol. 2, Harvard University Press, Cambridge, MA, 2.227-2.306, 1935-57.

[18] R. PFEIFER, M. LUNGARELLA, F. IIDA, The challenges ahead for bio-inspired 'soft' robotics. *Communications of the ACM*, **55**(11) (2012), pp. 76–87.

[19] PHONEBANK, see `http://spotlight.ccir.ed.ac.uk/public_documents/demonstrators/DEMO-F-1-E%2520(LTSB)/PhonebankExpress_Demo_Instructions.doc` (retrieved 2014).

[20] F. DE SAUSSURE, *Course in General Linguistics*, Ed. 3 (C. Bally, A. Sechehaye, A. Riedlinger) (R. Harris, transl. 1983). Duckworth London, 1915.

[21] R. K. STAMPER, *Information in Business and Administrative Systems*. BT Batsford, London, 1973.

[22] R. K. STAMPER, Towards a Theory of Information: Information: Mystical Fluid or a Subject for Scientific Discourse? *The Computer Journal, British Computer Society*, **28**(3) (1985), pp. 195–199.

[23] R. K. STAMPER, Organisational Semiotics: Informatics without the Computer? In *Information, Organisation and Technology: Studies in Organisational Semiotics (Information and Organisation Design)* (K. Liu, R. J. Clarke, P. B. Andersen, R. K. Stamper, Eds.), (2001) pp. 115–172. Springer.

[24] L. STEELS, The Symbol Grounding Problem has been solved. So what's next? In *Symbols and Embodiment: Debates on Meaning and Cognition* (M. de Vega, A. M. Glenberg, A. C. Graesser, Eds.), (2008) pp. 223–244. Oxford University Press, Oxford.

[25] K. TANAKA-ISHII, *Semiotics of Programming*. Cambridge University Press, Cambridge, UK, 2010.

[26] H. S. THAYER, *Pragmatism: The Classic Works*. Hackett Publishing Company, Inc., Indianapolis, Indiana, 1982.

[27] A. M. TURING, Computing Machinery and Intelligence. *Mind*, **236** (October 1950), 433–460.

[28] VOCALIQ, see `http://youtu.be/tgYpiHjjo8Q` and `http://youtu.be/jzaDyAAWOvI`

[29] J. WEIZENENBAUM, ELIZA – A Computer Program for the Study of Natural Language Communication between Man and Machine. *Communications of the ACM*, **9**(1) (1966), 36–45.

[30] M. J. WHEATMAN, A Semiotic Model of Information Systems. Presented in the *Proceedings of the International Conference on Informatics and Semiotics in Organisations*, (2011) Leeuwarden, The Netherlands.

[31] M. J. WHEATMAN, If we are holding hands whose hand am I holding? An Autopoietic Conceptual Analysis System. *Proceedings of the 3rd International Conference on Logistics Informatics and Service Science*, (2013) Reading, UK.

[32] T. WINOGRAD, *Understanding Natural Language*. Edinburgh University Press, 1972.

[33] L. WITTGENSTEIN, *Tractatus Logico-Philosophicus*, (D. F. Pears, B. F. McGuinness, (transl.). Routledge, London and New York, 1922.

[34] H. ZEMANEK, Semiotics and Programming Languages. *Communications of the ACM*, **9**(3) (1966), 139–143.

Received: November, 2013
Revised: April, 2014
Accepted: July, 2014

Contact address:
Martin Wheatman
Withinreap Barn
Moss Side Lane
Preston, PR3 2ND
United Kingdom
e-mail: martin@wheatman.net

MARTIN WHEATMAN has worked as a software engineer for almost 30 years. He gained his Ph.D. from Reading University, UK, researching Organisational Semiotics. From this, he has prototyped a cognitive model with the ability to interface natural utterances to persistent data, using a set of natural utterances: programming in natural language. He continues this through the Yagadi enguage(TM) language engine on various mobile platforms, providing a pervasive aural information system which benefits many disenfranchised groups.