Ambiguity in Natural Language Requirements Documents

Daniel M. Berry, CSCS & SE Program
University of Waterloo

Outline of Talk

- **Natural Language is Key in RE**
- **Avoiding or Detecting Ambiguity**
- **Overview of Dangerous Constructions**
- **Definitions and Examples of Ambiguity**
- **Conclusions**

Natural Language is Key in RE

Natural Language (NL) Requirements Specifications (RSs):

Overwhelming majority of RSs are written in NLs.

Virtually every initial conception for a system is written in NL.

Virtually every RFP is written in NL.

1996 Estimates

Osborne and MacNish reported John McDermid's estimates [Osborne1996] that

- **about 90% of SRSs are written in solely NL**
- **less than 10% of SRSs are written in NL plus formal language**
- **less than 1% of SRSs are written in solely formal language**

But...

We all know that...

NL is *so* ambiguous, and so inherently so!

No wonder RSs are such messes!

There is that old tradeoff: RSs written in

- **NLs or**
- **Mathematics-Based (MB) Formal Languages (FLs)**

NL:

– **inherently ambiguous**

\+ **always someone who can write it**

\+ **always more or less understood by all stakeholders,**
albeit somewhat differently by each

MB FL:

+ inherently unambiguous

- not always someone who can write it

- not understood by most stakeholders, although all that do understand it understand it the same

Stark Reality

But, the reality is that there is no escaping NL RSs.

Michael Jackson [Jackson1995] reminds us that

> **Requirements engineering is where the informal meets the formal.**

Stark Reality, Cont'd

Therefore, NLs are inevitable, even if it is only for the initial conception.

(Unless the client is some really weird math-type nerd who thinks in first-order predicate calculus, and how many of these are there?)

Stark Reality, Cont'd

Even if one moves immediately to FLs, the inherent ambiguity of the NL initial conception can strike as the transition is made.

What the formalizer understands of the conception may be different from what the conceiver meant.

The phenomenon of subconscious disambiguation strikes! [Gause2000]

Subconscious Disambiguation

In subconscious disambiguation (SD), the reader of an ambiguous phrase is not even aware that there is an interpretation other than the one that came first to his or her mind.

The reader understands an interpretation and thinks that it is the only one.

SD, Cont'd

In fact, *here* is where it's most important to catch ambiguity: right up front when the requirements analyst (RA) is getting raw information, be it goals, business rules, or requirements, from the clients and users.

The RA must find *each* ambiguity and ask the clients and users what they mean with it!

SD, Cont'd

Don Gause and Jerry Weinberg's original formulation of SD [Gause1989] was at a fairly high level:

Create a means for protecting a small group of humans from the hostile elements of their environment.

If the client was thinking of an igloo, but the RA immediately thought of a space station, then SD has struck again.

SD, Cont'd

We now realize that SD can strike at low-level details,

e.g., The spam filter only marks the e-mail it considers to be spam.

What does it *really* mean?

More on the problems with only **later!**

Subconscious Ambiguation

Subconscious ambiguation (SA) is the flip side of SD.

In SA, the writer of an ambiguous phrase is not even aware that he or she has written a phrase that has an interpretation other than what he or she thought to create the phrase.

Semi-Formal Languages?

What about semi-formal languages like UML notations?

Double whammy

Ambiguity can still strike when go from conception to model.

And

The model itself is not unambiguous.

Problem-Avoiding Approaches:

1. **Learn to write less ambiguously.**
2. **Learn to detect ambiguity.**
3. **Use a restricted NL which is inherently unambiguous.**

The first two reduce the disadvantage of existing NLs.

The third restricts the NLs to disallow the disadvantages.

Processing Restricted NL RSs

By restricting the NL (but then it is not so natural!), it becomes possible to ascribe precise semantics to the sentences.

Avoiding or Detecting Ambiguity and Other Problems in NL RSs

Some are focusing on the actual writing of the NL RS by the human writers, with an aim that they produce more complete, consistent, and less ambiguous NL RSs.

Some are focusing on improving inspections of NL RSs to more reliably detect incompleteness, inconsistencies and ambiguities.

Avoiding or Detecting Overview

Rupp & Rolf Götz [Rupp1997], Berry & Kamsties [Berry2000, Berry2005], and Berry, Kamsties, & Krieger [Berry2003] identify dangerous constructs to avoid.

Kovitz [Kovitz1998] and Dupré [Dupré1998] explain how to write requirements and technical material less ambiguously.

Kamsties [Kamsties2001a, Kamsties2001b] explains how to inspect RSs for ambiguities.

Avoiding or Detecting, Cont'd

Osborne & MacNish [Osborne1996], Wilson, Rosenberg, & Hyatt [Wilson1996, Wilson1997], Mich, Garigliano, *et al* [Mich2000, Mich2001, Kiyavitskaya2007], Bucchiarone & Gnesi [Bucchiarone2005, Berry2006], and Tjong [Tjong2006b, Tjong2006a, Tjong2007] have built or are building tools that find instances of potential ambiguities.

One Bit of Bad Advice

One bit of advice common to a lot of writing style guides and subscribed to by every writing teacher I had,

Vary your words; use synonyms to avoid boring repetition!

is *bad* advice for RSs writing.

Bad Advice, Cont'd

It creates what I call unnecessary synonymy (US) that leaves the reader wondering if there is an intended technical difference between two terms that generally mean the same thing, e.g.

data item **and** data element **and** datum**.**

I'd rather be bored by a RS than left wondering if these terms mean different things.

This is sort of the opposite of ambiguity!

Contracts, Laws, and RSs

Legal contracts, laws, and software RSs are similar in that

- **both are written in NLs and**
- **both have to anticipate all possible contingencies, i.e., exceptions.**

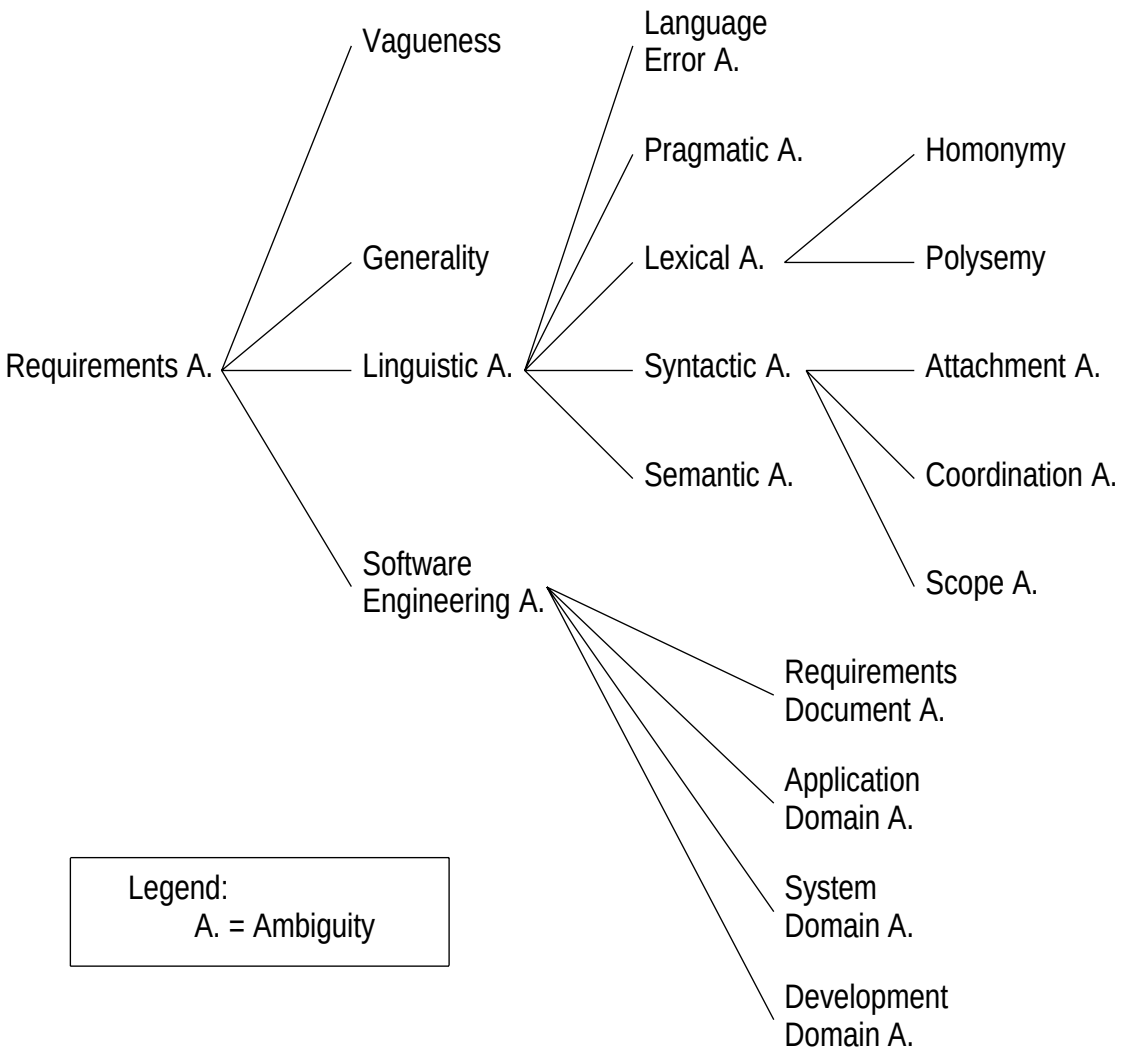
They share the same kind of ambiguity problems.

Definitions and Examples

Let's now focus on what *is* ambiguity in requirements specifications.

There are several sources of additional information [Berry2000, Berry2003, Berry2004, Berry2005].

First, a taxonomy of ambiguities in requirements specification


Requirements A.
Vagueness
Generality
Linguistic A.
Software Engineering A.
Language Error A.
Pragmatic A.
Lexical A.
Syntactic A.
Semantic A.
Homonymy
Polysemy
Attachment A.
Coordination A.
Scope A.
Requirements Document A.
Application Domain A.
System Domain A.
Development Domain A.
Legend:
A. = Ambiguity

Definitions of Ambiguity

- **Dictionary**
- **Linguistic**
- **Software Engineering**

Dictionary Definition

The Merriam-Webster English Dictionary [Merriam-Webster1998] defines "ambiguity" as

1a. the quality or state of being ambiguous especially in meaning,

1b. an ambiguous word or expression, or

2. uncertainty,

Dictionary Definition, Cont'd

where "ambiguous" means

1a. doubtful or uncertain especially from obscurity or indistinctness < eyes of an ambiguous color >,

1b. inexplicable, or

2. capable of being understood in two or more possible senses or ways.

Linguistic Definitions

Linguistic ambiguity is investigated in linguistics [Lyons1977] and in related fields, namely, computational linguistics [Hirst1987, Allen1995] and philosophy [Levinson1983, Walton1996].

I give examples of these shortly, but first, let's dispense of the third definition.

Software Engineering Definition

There is no single comprehensive definition of ambiguity in SE literature.

Each gives some key aspects and misses others.

Most are operational.

I give only two here, but there are others [Schneider1992, Gause1989].

IEEE Definition

The IEEE Recommended Practice for SRSs [IEEE1993] says that "An SRS is unambiguous if, and only if, every requirement stated therein has only one interpretation."

Presumably, an SRS is ambiguous if it is not unambiguous.

The problem with this definition is that there is no such thing as an unambiguous specification.

There are useful [Parnas2002] specifications!

Davis's Definition

Alan Davis [Davis1993] has suggested a test for ambiguity to serve as a definition:

"Imagine a sentence that is extracted from an SRS, given to ten people who are asked for an interpretation. If there is more than one interpretation, then that sentence is probably ambiguous."

Davis's Definition, Cont'd

The problem with this test is that, as in software testing, there is no guarantee that the eleventh person will not find another interpretation.

However, this test does capture the essence of a *useful* SRS, which is unambiguous for most practical purposes.

Linguistic Ambiguities

- **Lexical Ambiguity**
- **Syntactic Ambiguity**
- **Semantic Ambiguity**
- **Pragmatic Ambiguity**
- **Generality & Vagueness**
- **Language Error (NEW!)**

Lexical Ambiguity

***Lexical ambiguity* occurs when a word has several meanings.**

There are several kinds:

- **homonymy,**
- **polysemy, and**
- **systematic polysemy**

Lexical Ambiguity, Cont'd

***Homonymy* occurs when two different words have the same written and phonetic representation, but unrelated meanings and different etymologies, e.g.,**

bank

***Polysemy* occurs when a word has several related meanings but one etymology, e.g.,**

drunk, green

Lexical Ambiguity, Cont'd

***Systematic polysemy* occurs when the reason for the polysemy is confusion between classes, e.g., between unit and type, e.g.,**

I like this jacket.

and between process and product, e.g.,

I like writing.

Syntactic Ambiguity

***Syntactic ambiguity*, also called *structural ambiguity*, occurs when a given sequence of words can be given more than one grammatical structure, and each has a different meaning.**

There are several kinds, including:

- **attachment ambiguity and**
- **coordination ambiguity.**

Syntactic Ambiguity, Cont'd

***Attachment ambiguity* occurs when a particular syntactic constituent of a sentence, such as a prepositional phrase or a relative clause, can be legally attached to two parts of a sentence, e.g.,**

The police shot the rioters with guns.

Syntactic Ambiguity, Cont'd

***Coordination ambiguity* occurs**

- **when more than one conjunction,** and **or** or**, is used in a sentence, e.g.,**

  I saw Peter and Paul and Mary saw me.

- **or when one conjunction is used with a modifier,**

  young man and woman**.**

Semantic Ambiguity

Semantic ambiguity **occurs when a sentence has more than one way of reading it within its context even if it contains no lexical or structural ambiguity.**

These include

1. **coordination ambiguity (see above)**
2. **referential ambiguity (e.g, of pronouns, also is pragmatic ambiguity), and**
3. **scope ambiguity, e.g.,**

   All linguists prefer a theory.

Pragmatic Ambiguity

***Pragmatic ambiguity* occurs when a sentence has several meanings in the context in which it is uttered, e.g.,**

Every student thinks she is a genius.

Generality & Vagueness

Cousin **is general w.r.t. gender in English. So,**

Sue is visiting her cousin.

is general.

fast **has no clear boundary between fast and not fast. So,**

fast response time

is vague.

Language Error

Our experience has identified another category of ambiguity, *language error*.

As with all other categories, language error may not be mutually exclusive of other categories.

Language Error, Cont'd

A language error ambiguity occurs when a grammatical, punctuation, word choice, or other mistake in using the language of discourse leads to text that is interpreted by a receiver as having a meaning other than that intended by the sender.

Examples

The most common language errors are:

- only
- all **and plural**
- **pronouns**

They are certainly the most difficult to detect if you are not aware of the problem.

For a complete discussion of these errors and others, see our handbook [Berry2003]

Dangerously Misplaced “Only”

A very common mistake in English writing and speaking is the misplaced only**.**

To be correct, an only **should be immediately preceding the word or phrase that it limits.**

The typical native English speaker puts only **always before the main verb of its sentence, no matter what is limited by** only**.**

Misplaced “Only”, Cont’d

E.g., if it is desired to say that the only e-mail that a spam filter delivers to the user is e-mail that the user wants, one properly says:

The spam filter delivers only the e-mail that the user wants.

Misplaced “Only”, Cont’d

Many a native English speaker says instead:

The spam filter only delivers the e-mail that the user wants.

The meaning of this alternative sentence is that the only thing the spam filter does to the e-mail that the user wants is to deliver it; it does not forward, eat, modify, or anything else the e-mail that the user wants.

Misplaced “Only”, Cont’d

It does *not* promise to deliver only the e-mail that the user wants.

This incorrect sentence is understood by most native English speakers as it is probably meant to be understood, because what the sentence really means does not make much sense.

While the error can be made in other natural languages, in practice, speakers of other languages make the mistake far less often.

Misplaced “Only”, Cont’d

However, there are sentences of this form in which what the sentence really means is as meaningful as what it probably means, and the careful reader is left wondering what the writer really means.

Now do the in-class exercises!!!!

Misplaced “Only”, Cont’d

E.g.,

It only illustrates the concepts.

To most native English speakers, the sentence means:

It illustrates only the concepts and not reasons for them.

Misplaced “Only”, Cont’d

But, it really means:

It only illustrates the concepts and does not define them.

The correct sentence for the first is:

It illustrates only the concepts.

Misplaced “Only”, Cont’d

Another example:

I only nap after lunch.

To most native English speakers, the sentence means:

The only time I nap is after lunch.

Misplaced “Only”, Cont’d

But, it really means:

The only thing I do after lunch is nap.

The correct sentence for the first meaning is:

I nap only after lunch.

Misplaced “Only”, Cont’d

Another example:

The spam filter only marks the e-mail it considers to be spam.

Each of (1) the correct meaning, (2) what most native English speakers think it means, and (3) yet another meaning is a reasonable utterance about spam filters:

Misplaced “Only”, Cont’d

1. The spam filter only marks, and does not delete, the e-mail it considers to be spam.

2. The spam filter marks only the e-mail it considers to be spam.

3. The spam filter only marks only the e-mail it considers to be spam.

A California Proposition

Only marriage between a man and a woman is valid in California.

instead of the intended:

The only marriage that is valid in California is that between a man and a woman.

California Proposition, Cont'd

Hmmm, so are all relationships between a man and a woman other than marriage, e.g. friendship, dating, engagement, sex, separation, divorce, parental, grandparental, sibling, etc., not valid in California?

Misplaced “Only”, Cont’d

There are other words that have the same problem as only**. Among these words are** almost**,** also**,** even**,** hardly**,** just**,** merely**,** mostly**,** nearly**, and** really**.**

In other words, it is common to misplace also also**, not only** only**. (Thanks go Jo Atlee for the pun effects.)**

Other Languages

These syntactic problems with only **are not restricted to English.**

Other Languages, Cont'd

Each of the above examples using only **can be duplicated with the same meanings**

- **in French with** seulement **or** ne … que**,**
- **in German with** nur**,**
- **in Hebrew with** רק**,**
- **in Italian with** soltanto**,**
- **in Portuguese with** somente**, and**
- **in Spanish with** solamente**,**

respectively.

Syntactically Dangerous “All”

Consider the sentence,

All the lights in any room have a single on-off switch.

The question to be asked is “How many switches does any room have, one or one per light?”

Dangerous “All”, Cont’d

The problem with this sentence is that it is not clear whether

1. **each light in any room has its own single on-off switch that isn’t shared with any other light, or**
2. **all lights in any room share a common single on-off switch.**

The sentence is ambiguous.

Even a Third Meaning

There is yet another, more obscure, meaning, that …

3. **each light in any room has its own single on-off switch that may be shared with another light.**

Dangerous “All”, Cont’d

If one writes

Each light in any room has a single on-off switch.

or

Each light in any room has its own on-off switch.

then the first meaning is clearly intended.

Dangerous “All”, Cont’d

If he writes

All lights in any room share a single on-off switch.

then the second meaning is clearly intended.

Dangerous “All”, Cont’d

The ambiguous sentence

All the lights in any room have a single on-off switch.

is a classic example of scope ambiguity; it is not clear which quantifier equivalent, all**, for “∀”, or** a**, for “∃!” (there exists a unique), takes precedence over the other.**

Dangerous “All”, Cont’d

Mathematics shows the problem clearly. The two meanings are:

1. $\forall y \in$ the lights in a room, $\exists! x$ such that x is the on-off switch of y

2. $\exists! x$ such that $\forall y \in$ the lights in a room, x is the on-off switch of y

Dangerous “All”, Cont’d

Many times the same ambiguity is hidden by domain knowledge.

E.g., consider ambiguous sentence (that is structurally similar to the “lights” sentence),

All persons have a unique national insurance number.

There is another, semantic danger in the sentence: it ain’t true,

Therefore, the software should not depend on it!

Dangerous “All”, Cont’d

Domain knowledge tells the reader that the intended meaning of the sentence is that

- **each person has his or her own unique national insurance number,**

and not the ridiculous idea that

- **all persons share a common unique national insurance number.**

Dangerous “All”, Cont’d

The second option is so ridiculous that most readers of the sentence would not even think that there *is* another option and that the sentence is ambiguous.

Syntactically Dangerous Plural

Closely related to the syntactically dangerous all**, is the syntactically dangerous plural.**

The use of plural to describe a property of elements of a set or of sets makes it difficult to determine whether the property is that of each element or of the whole set.

Dangerous Plural, Cont'd

Consider the two structurally identical sentences:

Students enroll in six courses per term.

Students enroll in hundreds of courses per term.

Domain knowledge tells us that the first sentence is talking about each student while the second is talking about the whole set of students.

Dangerous Plural, Cont'd

Without this domain knowledge, there is nothing in either sentence to indicate whether enrollment in the stated number of courses per term is a property of each student or of the set of all students.

Dangerous Plural, Cont'd

The first sentence is talking about each student; it should be written in singular form:

Each student enrolls in six courses per term.

Dangerous Plural, Cont'd

Using a singular formulation for talking about properties of each or any student reserves the plural formulation:

Students enroll in hundreds of courses per term.

for talking about properties of the collection of students.

Dangerous Plural, Cont'd

Alternatively, you could insist on singular even for sets, by introducing some set equivalent to hold the elements of the set:

The student body enrolls in hundreds of courses per term.

Dangerous Plural, Cont'd

The same syntactic problem exists with other, non-universal quantifier equivalents, e.g., some, many**, which are all plural.**

Even in Math or Tech Writing

Plural ambiguity is particularly problematic in mathematical or technical writing, although there are occasionally nearby formulae that disambiguate.

E.g., Systems contain subsystems.

Is containment one–one, one–many, many–one, or many–many?

Math or Tech Writing, Cont'd

Domain knowledge tells us what makes sense in this case, …

but if you are trying to learn new mathematics with a sentence like this, then what?

Guilt and My Writing

I began to feel guilty about using plural in my own writing.

Could I in good conscience write any sentence with an ambiguous use of plural?

These sentences amount to almost every sentence with plural.

Would I be able to sleep nights?

Guilt, Cont'd

I finally decided to banish sentences with the plural ambiguity from my writing, using plural *only* when I am talking about a property of the set of objects denoted by a plural construct, or rarely, when ambiguity is totally innocuous [Chantree2005].

Banishing Plural

Implementing this decision meant finding singular equivalents for many**,** some**,** few**, etc. …**

e.g. in Many people like to eat their lunch with a cold drink.

or Many people like to eat their lunches with cold drinks.

Singular Substitutes

Like each **is a singular substitute for** all**.**

For many**, I use** many a **or** the typical**.**

For some**, I use** the occasional**.**

For few**, I use** the rare**.**

e.g. Many a person likes to eat his or her lunch with a cold drink.

and *definitely not* Many a person likes to eat their lunch with a cold drink.

My Recent Writing

For the past 3 years I have been careful in my writing to avoid ambiguous plural and other problems mentioned in these slides, especially misplaced only**s and** also**s.**

My Recent Writing, Cont'd

Sometimes the wording is strange.

My coauthors notice it and complain, but eventually accept it.

An occasional reviewer complains that we have English errors all over or that we need to have an English editor clean up the paper.

The typical reviewer nevertheless says that the paper is well-written and clear, even when occasionally rejecting it!

My Recent Writing, Cont'd

I have had trouble with the typical copy editor who rewrites sentences into plural and moves the only**s and** also**s to their normal sounding places, *even in an article about the plural problem or about ambiguities, including that of misplaced* only*s and* also*s.***

Other Languages

These syntactic problems with plural universal quantifier equivalents and with plural sentences are not restricted to English.

Other Languages, Cont'd

Each of the above examples using all **or** each **can be duplicated with the same meanings**

- **in French with** tous **or** chaque**,**
- **in German with** alles **or** jeder**,**
- **in Hebrew with** כל **or** כל אחד**,**
- **in Italian with** tutti **or** ogni**, and**
- **in Portuguese and Spanish with** todo **or** cada**,**

respectively.

Dangerous Plural, Cont'd

Mathematics has adopted a convention that makes intent very clear.

In mathematics, the universal quantifier $\forall$, read as for all **is singular as in,**

$\forall\ x \in Int,\ x < x+1$

For all Integers x, x is less than $x+1$

Dangerous Pronouns

With each pronoun, to which noun it refers is problematic, e.g.,

Every student thinks she is a genius.

One must be careful in writing to make sure that the referent of a pronoun is what is intended.

When one is writing text, she has no difficulty understanding a pronoun's referent.

Dangerous Pronouns, Cont'd

However, the poor reader must often guess.

The grammatical rules say that the referent of a pronoun must be the previous noun or the non-pronoun sentence subject.

This rule alone is ambiguous.

Moreover, sometimes the writer does not follow this rule in her thinking.

Dangerous Pronouns, Cont'd

The best defense is to use nouns instead of pronouns, but that can sound funny.

Another good defense is to introduce formal names:

Consider the switch *s1*. … *s1* is turned off.

Dangerous “This”

The most insidious problem is that of This**. (I capitalize it because it usually comes at the beginning of a sentence.)**

The writer says, e.g.:

This prevents security breaches.

Dangerous “This”, Cont’d

To what does This **refer?**

to the previous noun?
to the previous sentence subject?
to the idea of the previous sentence?
to the idea of the previous *n* sentences?
to the idea of the current paragraph?
to the idea of the previous paragraph?

Dangerous “This”, Cont’d

I have seen all these possibilities, and …

I have seen situations in which more than one of these makes sense.

Dangerous “This”, Cont’d

The defense:

Always follow this **by a noun that restricts the referent**

e.g,

This encoding scheme prevents security breaches.

Other Languages

These problems with This **are not restricted to English.**

Certainly, each language other than English has pronouns, which can have uncertain referents.

Other Languages, Cont'd

Moreover, each of the above examples using This **can be duplicated with the same meanings**

- **in French with** Ceci**,**
- **in German with** Dieses**,**
- **in Hebrew with** זהו**,**
- **in Italian with** Ciò**,**
- **in Portuguese with** Isto**, and**
- **in Spanish with** Esto**,**

respectively.

Conclusions

NL is unavoidable in RSs, even if only at the very beginning when you are talking with the client.

SA strikes in writing.

SD strikes in reading.

Conclusions, Cont'd

Ambiguity abounds in places you never even thought of, e.g., in only**, in** all**, and in plural.**

Any tool must have 100% recall and good summarization at the expense of some imprecision.

Most Important Lesson

In reality, we are never going to prevent ambiguity.

So we must learn to spot it, not only in polished RSs, …

but also, and especially, *in goals, business rules, and INITIAL RSs, in whose reading SD first strikes!*

AND ASK THE CLIENT WHAT HE OR SHE MEANS!

[Abbott1983] R.J. Abbott, “Program Design by Informal English Descriptions”, *CACM* **26**(11) (November 1983).

[Allen1995] J. Allen, *Natural Language Understanding*, Second Edition, Addison-Wesley, Reading, MA (1995).

[Ambriola2000] V. Ambriola and V. Gervasi, “Supporting Multiple Views on Requirements”, pp. 321–330 in *Proceedings of the Sixth Maghrebian Conference on Computer Sciences*, Fes, Morocco (November 2000).

[Basili1997] V.R. Basili, “Evolving and Packaging Reading Technologies”, *Journal of Systems and Software* **38**(1), pp. 3–12 (1997).

[Berry2000] D.M. Berry and E. Kamsties, “The Dangerous ‘All’ in Specifications”, pp. 191–194 in *Proceedings of 10th International Workshop on Software Specification & Design, IWSSD-10*, IEEE CS Press (2000).

[Berry2003] D.M. Berry, E. Kamsties, and M.M. Krieger, “From Contract Drafting to Software Specification: Linguistic Sources of Ambiguity”, Technical Report, University of Waterloo, Waterloo, ON, Canada (2003), http://se.uwaterloo.ca/~dberry/handbook/ambiguityHandbook.pdf.

[Berry2004] D.M. Berry and E. Kamsties, “Ambiguity in Requirements Specification”, pp. 7–44 in *Perspectives on Requirements Engineering*, ed. J.C.S.P. Leite and J. Doorn, Kluwer, Boston, MA (2004).

[Berry2005] D.M. Berry and E. Kamsties, “The Syntactically Dangerous *All* and Plural in Specifications”, *IEEE Software* **22**(1), pp. 55–57 (January+ February 2005).

[Berry2006a] D.M. Berry and E. Kamsties, “Clarifying Ambiguity”, *IEEE Software* **23**(4), pp. 8–9 (July/August 2006).

[Berry2006b] D.M. Berry, A. Bucchiarone, S. Gnesi, G. Lami, and G. Trentanni, “A New Quality Model for Natural Language Requirements Specifications”, in *Proceedings of the Twelfth International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ)*, Luxembourg (5–6 June 2006).

[Booch1991] G. Booch, *Object Oriented Design*, Benjamin/Cummings, Redwood City, CA (1991).

[Bucchiarone2005] A. Bucchiarone and S. Gnesi, “Quality Analysis of NL Requirements: An Industrial Case Study”, pp. 390–394 in *Proceedings of the Thirteenth IEEE International Conference on Requirements Engineering (RE’05)*, Paris, France (29 August–2 September 2005).

[Chantree2005] F. Chantree, B. Nuseibeh, A. De Roeck, and A. Willis, “Nocuous Ambiguities in Requirements Specifications”, Technical Report No. 2005/03, Department of Computing, The Open University, Milton Keynes, UK (3 March 2005).

[Comer1983] J.R. Comer, “An Experimental Natural-Language Processor for Generating Data Type Specifications”, *SIGPLAN Notices* **18**(12), pp. 25–33 (December 1983).

[Davis1993] A. Davis, *Software Requirements: Objects, Functions, and States*, Prentice-Hall, Englewood Cliffs, NJ (1993).

[Denger2001] C. Denger, J. Dörr, and E. Kamsties, "A Survey on Approaches for Writing Precise Natural Language Requirements", IESE-Report, Fraunhofer IESE (2001).

[Dupré1998] L. Dupré, *Bugs in Writing: A Guide to Debugging Your Prose*, Second Edition, Addison-Wesley, Reading, MA (1998).

[Enomoto1984a] H. Enomoto, N. Yonezaki, M. Saeki, and H. Armata, "Formal Specification and Verification for Concurrent Systems by TELL", pp. 732–745 in *Advances in Artificial Intelligence, ECAI'84*, ed. T. O'Shea, Elsevier North-Holland (1984).

[Enomoto1984b] H. Enomoto, N. Yonezaki, M. Saeki, K. Chiba, T. Takizuka, and T. Yokoi, "Natural Language Based Software Development System TELL", pp. 721–731 in *Advances in Artificial Intelligence, ECAI'84*, ed. T. O'Shea, Elsevier North-Holland (1984).

[Frakes1992] W.B. Frakes and R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall, Englewood Cliffs, NJ (1992).

[Frost2006] R.A. Frost, "Realization of Natural Language Interfaces Using Lazy Functional Programming", *ACM Computing Surveys* **38**(4), pp. 1 (December 2006).

[Fuchs1999] N.E. Fuchs, U. Schwertel, and R. Schwitter, "Attempto Controlled English (ACE) Language Manual Version 3.0", Technical Report Nr. 99.03, Institut für Informatik der Universität Zürich, Zürich, Switzerland (1999).

[Gause1989] D.C. Gause and G.M. Weinberg, *Exploring Requirements: Quality Before Design*, Dorset House, New York, NY (1989).

[Gause2000] D.C. Gause, "User DRIVEN Design—The Luxury that has Become a Necessity, A Workshop in Full Life-Cycle Requirements Management", ICRE 2000 Tutorial T7, Schaumberg, IL (23 June 2000).

[Gervasi2000a] V. Gervasi, "Environment Support for Requirements Writing and Analysis", Ph.D. Dissertation, TD-3/00, Dipartamento di Informatica, Università di Pisa, Pisa, Italy (2000).

[Gervasi2000b] V. Gervasi and B. Nuseibeh, "Lightweight Validation of Natural Language Requirements", pp. 140–148 in *Proceedings of Fourth IEEE International Conference on Requirements Engineering (ICRE'2000)*, IEEE Computer Society Press, Schaumburg, IL (19–23 June 2000).

[Henninger1978] K.L. Henninger, J.W. Kallander, J.E. Shore, and D.L. Parnas, "Software Requirements for the A-7E Aircraft", NRL Memorandum Report 3876, Naval Research Laboratory, Washington, DC (1978).

[Hirst1987] G. Hirst, *Semantic Interpretation and the Resolution of Ambiguity*, Studies in Natural Language Processing, Cambridge University Press, Cambridge, UK (1987).

[IEEE1993] IEEE, *IEEE Recommended Practice for Software Requirements Specifications*, ANSI/IEEE Standard 830-1993, Institute of Electrical and Electronics Engineering, New York, NY (1993).

[Ishihara1993] Y. Ishihara, H. Seki, and T. Kasami, "A Translation Method from Natural Language Specifications into Formal Specifications Using Contextual Dependencies", pp. 232–239 in *Proceedings of the IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, San Diego, CA (January 1993).

[Jackson1995] M.A. Jackson, "Problems and Requirements", pp. 2–8 in *Proceedings of the Second IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, York, UK (March 1995).

[Kamsties2001a] E. Kamsties, D.M. Berry, and B. Paech, "Living with Ambiguity in Industrial Requirements Specifications", Technical Report, Computer Science, University of Waterloo (2001).

[Kamsties2001b] E. Kamsties, "Surfacing Ambiguity in Natural Language Requirements", Ph.D. Dissertation, Fachbereich Informatik, Universität Kaiserslautern, Germany, also Volume 5 of Ph.D. Theses in Experimental Software Engineering, Fraunhofer IRB Verlag, Stuttgart, Germany (2001).

[Kiyavitskaya2007] N. Kiyavitskaya, N. Zeni, L. Mich, and D.M. Berry, "Requirements for Tools for Ambiguity Identification and Measurement in Natural Language Requirements Specifications", in *Proceedings of Workshop in Requirements Engineering (WER)*, Toronto, ON, Canada (May 2007), http://wer.inf.puc-rio.br/index.html.

[Kovitz1998] B.L. Kovitz, *Practical Software Requirements: A Manual of Content and Style*, Manning, Greenwich, CT (1998).

[Leite1993] J.C.S.P. Leite and A.P.M. Franco, "A Strategy for Conceptual Model Acquisition", pp. 243–246 in *Proceedings of the IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, San Diego, CA (January 1993).

[Levinson1983] S. Levinson, *Pragmatics*, Cambridge University Press, Cambridge, UK (1983).

[Lyons1977] J. Lyons, *Semantics I and II*, Cambridge University Press, Cambridge, UK (1977).

[Merriam-Webster1998] Merriam-Webster, "Merriam-Webster's Collegiate Dictionary", Tenth Edition, Merriam-Webster, Springfield, MA (1998), http://www.m-w.com/dictionary.htm.

[Mich2000] L. Mich and R. Garigliano, "Ambiguity Measures in Requirements Engineering", in *Proceedings of the International Conference on Software Theory and Practice (ICS2000)*, Sixteenth IFIP World Computer Conference, Beijing, China (21–24 August 2000).

[Mich2001] L. Mich, "On the use of Ambiguity Measures in Requirements Analysis", in *Applications of Natural Language to Information Systems, Sixth International Workshop NLDB'01*, ed. A.M. Moreno and R.P. van de Riet, Madrid, Spain (28–29 June 2001).

[Mich2003] L. Mich, M. Franch, and P. Novi Inverardi, "Requirements Analysis using Linguistic Tools: Results of an On-line Survey", to appear in *Requirements Engineering Journal* 2003 or 2004, Technical Report 66, Department of Computer and Management Sciences, Unversità di Trento, Trento, Italy (2003), http://eprints.biblio.unitn.it/view/department/informaticas.html.

[Mich2004] L. Mich, M. Franch, and P. Novi Inverardi, "Market Research for Requirements Analysis Using Linguistic Tools", *Requirements Engineering Journal* **9**(1 & 2), pp. 40–56 (in No. 1) 151 (in No. 2) (2004),
No. 1 has full article with inverted names, No. 2 has correction of names and reference to full article in Issue 1.

[Neumann1984] P.G. Neumann, "Only His Only Grammarian Can Only Say Only What Only he Only Means", *ACM SIGSOFT Software Engineering Notes* **9**(1), pp. 6 (January 1984).

[Osborne1996] M. Osborne and C.K. MacNish, "Processing Natural Language Software Requirement Specifications", pp. 229–236 in *Proceedings of the International Conference on Requirements Engineering (ICRE)*, Colorado, Springs, CO, USA (1996).

[Parnas2002] D.L. Parnas, *personal communication via electronic mail*, November 2002.

[Popescu2006] D. Popescu, "Quality Improvement of Requirements Specifications via Automatically Created Object-Oriented Models", M.Sc. Thesis, SCS Technical Report; GIT-CSS-06-07, School of Computer Science, Georgia Institute of Technology, Altanta, GA, USA (2006),
http://smartech.gatech.edu/dspace/bitstream/1853/14345/1/GT-CSS-06-07.pdf.

[Popescu2007] D. Popescu, S. Rugaber, N. Medvidovic, and D.M. Berry, "Improving the Quality of Requirements Specifications via Automatically Created Object-Oriented Models", in *Proceedings of the Fourteenth Monterey Workshop, Wor kshop on Innovations for Requirements Analysis: From Stakeholders Needs to Formal Designs*, Monterey, CA, USA (10–12 September 2007).

[Rolland1992] C. Rolland and C. Proix, "A Natural Language Approach for Requirements Engineering", pp. 257–277 in *Proceedings of Conference on Advanced Information Systems Engineering, CAiSE 1992*, Manchester, UK (12–15 May 1992).

[Rupp1997] C. Rupp and R. Götz, "Sprachliche Methoden des Requirements Engineering (NLP)", in *Proceedings of CONQUEST-1, First Conference on Quality Engineering in Software Technology*, Nürnberg, Germany (25–26 September 1997).

[Ryan1993] K. Ryan, "The Role of Natural Language in Requirements Engineering", pp. 240–242 in *Proceedings of the IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, San Diego, CA (January 1993).

[Saeki1987] M. Saeki, H. Horai, K. Toyama, N. Uematsu, and H. Enomoto, "Specification Framework Based on Natural Language", pp. 87–94 in *Proceedings of the Fourth International Workshop on Software Specification and Design*, Monterey, CA (April 1987).

[Salton1989] G. Salton, *Automatic Text Processing: The Translation, Analysis, and Retrieval of Information by Computer*, Addison Wesley, Reading, MA (1989).

[Schneider1992] G.M. Schneider, J. Martin, and W.T. Tsai, "An Experimental Study of Fault Detection in User Requirements Documents", *ACM Transactions on Software Engineering and Methodology* **1**(2), pp. 188–204 (April 1992).

[Schwertel2000] U. Schwertel, "Controlling Plural Ambiguities in Attempto Controlled English", pp. 120–133 in *Proceedings Third International Workshop on Controlled Language Applications*, Seattle, WA, USA (29–30 April 2000), http://unizh.ch/attempto/publications/papers/claw2000.pdf.

[Schwertel2003] U. Schwertel, "Plural Semantics for Natural Language Understanding—A Computational Proof-Theoretic Approach", Ph.D. Thesis, Faculty of Arts, University of Zurich (2003).

[Somé1996] S. Somé, R. Dssouli, and J. Vaucher, "Toward an Automation of Requirements Engineering Using Scenarios", *Journal of Computing and Information* **2**(1), pp. 1110–1132 (1996).

[Srinivasan1992] R. Srinivasan, "Thesaurus Construction", pp. 161–218 in *Information Retrieval: Data Structures and Algorithms*, ed. W.B. Frakes and R. Baeza-Yates, Prentice-Hall, Englewood Cliffs, NJ (1992).

[Tjong2006a] S.F. Tjong, "Elaborated Natural Language Patterns for Requirements Specifications", Faculty of Engineering and Computer Sciences, University of Nottingham, Malaysia Campus, Selangor Darul Ehsan, Malaysia (August 2006), http://sepang.nottingham.edu.my/~kcx4sfj/.

[Tjong2006b] S.F. Tjong, "Improving the Quality of Natural Language Requirements Specifications through Natural Language Requirements Patterns", Faculty of Engineering and Computer Sciences, University of Nottingham, Malaysia Campus, Selangor Darul Ehsan, Malaysia (March 2006), http://sepang.nottingham.edu.my/~kcx4sfj/.

[Tjong2007] S.F. Tjong, M. Hartley, and D.M. Berry, "Extended Disambiguation Rules for Requirements Specifications", in *Proceedings of Workshop in Requirements Engineering (WER)*, Toronto, ON, Canada (May 2007), http://wer.inf.puc-rio.br/index.html.

[Walton1996] D. Walton, *Fallacies Arising from Ambiguity*, Applied Logic Series, Kluwer Academic, Dordrecht, NL (1996).

[Wilson1996] W.M. Wilson, L.H. Rosenberg, and L.E. Hyatt, "Automated Quality Analysis of Natural Language Requirements Specifications", NASA Software Assurance Technology Center, The Software Assurance Technology Center (SATC), NASA Goddard Space Flight Center (GSFC), Greenbelt, MD (1996), http://satc.gsfc.nasa.gov/support/PNSQC_OCT96/pnq.html.

[Wilson1997] W.M. Wilson, L.H. Rosenberg, and L.E. Hyatt, “Automated Analysis of Requirements Specifications”, in *Proceedings of the International Conference on Software Engineering (ICSE)*, Boston, MA (May 1997).