The University of Nottingham

Avoiding Ambiguity in Requirements Specifications

Sri Fatimah Tjong, B.Sc.

Thesis submitted to the University of Nottingham

for the degree of Doctor of Philosophy

February 2008

Supervisors:

Dr. Michael Hartley

University of Nottingham Malaysia Campus, Faculty of Engineering & Computer Science, Malaysia

Prof. Daniel M. Berry

University of Waterloo, Waterloo, Ontario, Canada

Abstract

Writing a requirements specification (RS) in natural language (NL) requires dealing with the inherent ambiguity of the NL. Any RS with words all, any, and, or, and/or, but, unless, if, only, also, it, they, or plural nouns is potentially ambiguous. Ambiguity in an RS is a contributor to system difficulties because the RS fails to specify unique requirements for the system. Another cause of system difficulties is the requirements analyst's (RA's) failing to notice ambiguities in a RS and thinking that his or her first interpretation of the RS is the only one.

This thesis is focused in the field of requirements engineering (RE). It gives guiding rules, derived from a study of a corpus of nine industrial RSs, that help to write less ambiguous NL RSs. The guiding rules can serve also as an inspection checklist that help find ambiguities in RSs.

The applicability of the guiding rules was demonstrated by their use to find ambiguities in real-life RSs. An experimental tool, SREE, was designed to aid a RA in detecting instances of potential ambiguity in a NL RS. When SREE finds an instance of potential ambiguity, SREE reports the instance to the user, so that the user can decide if the instance is truly ambiguous and to disambiguate the instance if desired. SREE is a lexical analyser, searching for instances for only specific words in SREE's database. This way, SREE is designed to have 100% recall for potential ambiguities in SREE's scope. SREE was applied to two of the RSs used to construct the guiding rules. From these applications, SREE's less than 100% precision was calculated.

Acknowledgement

First and foremost, I thank my parents for the prayers they sing to bless me.

I express my thanks to University of Nottingham Malaysia Campus for providing a PhD scholarship which allows me to develop this research work.

I truthfully thank Prof. Daniel M. Berry for being my advisor, for his constant support, for sharing his ideas electronically, giving prompt countless electronic replies to my questions, for inviting me to University of Waterloo, Canada where I spent quality time to work and have fruitful discussions.

I express my gratitude to Dr. Michael Hartley, Dr. Nasreddine Hallam, and Dr. Mohammed Saleh who had provided scientific guidance and practical helps during my PhD supervision.

I thank also Jack for many discussions, ideas, and technical support during the development of SREE.

Vocabulary

AIC – Ambiguity Indicator Corpus

CIC – Customised Indicator Corpus

OIC – Original Indicator Corpus

FL – Formal Language

RE – Requirements Engineering

RM – Requirements Management

RS – Requirements Specification

RStat – Requirements Statement

NL – Natural Language

NL RS – Natural Language Requirements Specification

NLP – Natural Language Processing

Table of Contents

List of Algorithm

List of Figures

List of Tables

1. Introduction

"Unintended ambiguity is the Achilles' heel of SRSs …" – Daniel M. Berry, Erik Kamsties, and Michael M. Krieger.

Requirements engineering (RE), being the core of software development, is concerned with identifying the purpose of a software system and the contexts in which it will be used. RE also facilitates effective communication of the requirements among the stakeholders affected by a software system. However, poor communication of domain knowledge is a major source of requirements defects. A significant number of requirements defects can be characterised as failures to adequately take into account the context of the software to be built. The pervasive medium for this communication, natural language (NL), is widely accepted to be problematic for high-precision communication because of its characteristic ambiguity and informality. NL, regardless of the availability of various notations such as diagrams, formal languages (FLs), or even pseudo-code, is still the most frequently used representation in which to state requirements that are to be met by information technology (IT) products or services. Even if a FL is used for a formal requirements specification (RS), the initial conception of a system are almost always conveyed in NL.

Many an approach to removing ambiguity and increasing the precision of a RS written in NL advocates the practice of converting the RS's representation into a formal one, for several reasons, the most familiar being that a FL is inherently unambiguous and analysable. A FL uses precise mathematical notation to eliminate ambiguity. Examples of FLs are: Z and B, VDM, LOTOS, Petri Nets, etc. [Kazmierczak, 2003].

Others propose the use of formal modeling techniques and methods, as summarised in QUASAR [Denger et. al., 2001].

1.1 Main Contribution

This thesis describes the analysis of several RSs with the aim of producing guiding rules for avoiding writing ambiguous RSs. The analysed RSs [Bray, 2002; BPS, 2005; CLS, 1999; DCS, 2002; EVLA, 2003; LAT, 2000; PESA, 2001; Stevenson et al., 2005; MRD, 1997; CMS, 2006] form a corpus of NLRSs. The analysis identifies certain words such as all, any, and, or, and/or, but, unless, if, only, also, it, they, and others, to be potentially ambiguous. A guiding rule is an instruction describing an ambiguous language use pattern with a suggestion for replacing that ambiguous language use with a less ambiguous way to say what is intended. One may use the guiding rules also to drive inspections for ambiguities in a RS.

To validate the usefulness of the guiding rules, this thesis discusses and gives examples of ambiguous requirements statements (RStats) from the corpus and their rewritten forms. Moreover, this thesis describes also the development of a Systemised Requirements Engineering Environment (SREE) tool that searches for instances of potential ambiguity in a RS and shows them to its user without attempting to disambiguate them. Distinguishing a true ambiguity from a potential ambiguity and correcting an ambiguity require human intelligence and understanding. No machine is able to understand any NL perfectly, Hence, it is better to only report instances of potential ambiguity and to let the user decide if they are truly ambiguous (TA).

The objective of this thesis is to contribute the following results:

- collecting a corpus of NLRSs from different domains,
- analysing existing approaches for detecting and correcting ambiguous NLRSs,
- identification of ambiguities frequently found in RSs and exhibiting possible interpretations of these ambiguities,
- constructing a collection of guiding rules to help avoid ambiguities while authoring or inspecting RSs, and
- developing SREE, an experimental tool that automatically detects instances of potential ambiguity in a RS. For every potential ambiguity detected, SREE is to show to the user the potential ambiguity it found, and let the user decide whether the potential ambiguity is a true ambiguity.

Constructing a complete set of guiding rules, which capture all possible ambiguities and are applicable to all domains, demands research that is beyond the scope of this thesis and also is impossible. Of course, it is always possible to add new guiding rules.

1.2 Problem Investigation

The main underlying problems and practical results of this thesis are:

- **The Problematic Role of Context for Setting Guiding Rules**

  Context is fundamental in communicating substantial information with few words. The modelling of context and using context in understanding NL are complicated. Unlike a programming language in which one can define contextual influence in a limited and controlled way, context is all pervasive and powerful in NL communication. Different domains may use different contexts as the source of information to represent the intended idea. Understanding a domain's context requires human knowledge and insight. For example, the acronym OS can refer to Open Source and also to Operating System, and both could be used in a document describing an open source operating system.

- **The Source of Ambiguity: Imprecision, Indeterminacy, Uncertainty, and Vagueness**

  For decades, ambiguity has been a focus of NL processing research [Brill and Mooney, 1997]. Some of the potential ambigutity of the RStat Turning the switch down (up) turns the light on (off). comes from imprecision. Some of the potential ambiguity of the RStat The user shall be trusted or not trusted. comes from indeterminacy. Some of the potential ambiguity of the RStat The Science Analysis Software performs prompt processing of Level 0 data to produce Level 1 event data. comes from vagueness. This thesis collapses these phenomena into one term, "ambiguity", because each phenomenon contributes to the same problem, which is the inability to know what a RStat means.

- **Disambiguating an Ambiguous RS Needs Contextual Knowledge**

  In interpreting a RStat, the inappropriate placement of grammatical context is one contributor to syntactic ambiguity and semantic ambiguity. From the study of ambiguity in the corpus of RSs, it is clear that the ultimate way to disambiguate any ambiguity is to ask the author of the RS what he or she meant.

- **Automatic Disambiguation Is Not Possible Without Intelligence**

  A software tool can neither understand nor automatically disambiguate any potential ambiguity it detects, It cannot even decide if a potential ambiguity is indeed truly ambiguous. It may not even be able to find a potential ambiguity. Ultimately, these activities require human intelligence.

- **Practical Application**

  SREE was developed as an experimental tool to show how guiding rules can be practically implemented. Basically SREE recognises a RS and reports each potential ambiguity it finds. SREE detects instances of ambiguity indicators that are given in its ambiguity indicator corpus (AIC). The user of SREE is to decide whether any reported potential ambiguity is truly ambiguous.

1.3 Overview of Thesis Content

Chapter 2 of this thesis describes the nature of RE, key issues in RE, and the phases of the RE lifecyle. It discusses how RE relates to NLs, how NLs serve as interfaces in the RE domain, and the challenging issues from this relationship. Chapter 3 discusses ambiguities involving the coordinators and, or, inclusive or, and exclusive or. It covers also ambiguities involving unless, only, also, and if. Chapter 4 describes ongoing research and the state of practice in writing NLRSs. Chapter 5 outlines the research method adopted in this thesis. Chapter 6 gives the guiding rules discovered by the thesis author. Chapter 7 shows how the guiding rules were validated. Chapter 8 describes SREE and its implementation. Chapter 9 compares SREE with TIGER, another ambiguity finding tool, by comparing their analyses of two RSs. Chapter 10 summarises the thesis work, and Chapter 11 suggests future research.

1.4 Thesis Convention

In the rules and in examples, text from a RS is typeset in a sansserif typeface. A constant in such text is typeset in an upright sansserif typeface, and a variable in such text to be replaced by constant text of the variable's type is typeset in an oblique, a.k.a. slanted, sansserif typeface. Non-example, explanatory text of the thesis is typeset in a serif typeface. The reader should pay attention to the typeface of any punctuation at the end of any sansserif snippet to determine if the punctuation is part of the snippet or part of the surrounding explanatory text. When there are two adjacent punctuation symbols, usually, the first belongs to the snippet, which therefore ends with the first symbol, and the second belongs to the explanatory text surrounding the snippet.

2. RE and NL

"Requirements engineering is where the informal meets the formal." – Michael Jackson

Development of a software system can involve building its software from scratch, reusing existing components, extending and modifying existing systems, or integrating commercial-off-the-shelf packages. In any of these situations, it is important to guarantee consistency between various RSs generated during the software development process. In particular, it is important also to ensure consistency between the RS and the design specifications, to ensure the consistency between the design specification and the software system's code, and therefore, to ensure that the developed software system meets the RS.

RE, the writing of a RS is a critical task, since many software failures originate from inconsistent, incorrect, imprecise, or even ambiguous RSs. Detecting and correcting any of these problems is difficult, time consuming, and expensive, especially when the problem is detected in later phases of software development.

2.1 How RE relates to NL

A contributor to a defective RS is the inherent ambiguity of NL. A RS may be written in a NL or in a FL. A NL is inherently ambiguous, whereas a FL is inherently unambiguous. Ryan argues that a NL processing system can never be relied on to provide a complete understanding of a NLRS [Ryan, 1993]. However, some requirements are difficult and may even be impossible to express in any existing FL.

Hence, despite NL's inherent ambiguity and informality, NL is still necessary to provide the semantics not expressible in any FL. Fuchs and others [Fuchs and Schwitter, 1995; Schwertel, 2000] have developed a restricted NL called Attempto Controlled English (ACE), which uses a sublanguage of English simple enough to avoid NL ambiguity but expressible enough to be able to define requirements in it with the rigour possible with a FL. Section 2.3 discusses the key issues of NL use in RE.

2.2 Overview of RE

A RS consists of RStats intended to describe a software system to be built. Generally, the RS describes the users' expectations on performance, availability, usability, reliability, and other quality attributes of the software system to be built. Besides, it is necessary also to expose pertinent business rules, design and implementation constraints, and assumptions of the various stakeholders.

RE is the process of discovering all requirements, needs, purposes, and constraints of the proposed system required by the stakeholders and then documenting them in a RS that is useful for system implementation. RE research has produced an extensive body of knowledge and a variety of methods, notations, or automated tools. The main objective of RE is for a RA to understand what the stakeholders are envisioning so that the RA can specify all requirements that will guide the development team's work.

2.2.1 Key Issues in RE

RE research has led to the development of techniques for eliciting and analysing stakeholders' goals, to the modeling of scenarios that characterise different contexts of system use, to the use of ethnographic techniques for studying organisations and work settings, and to the use of formal methods for analysing safety and security requirements. Despite these advances, RE remains one of the most challenging parts in software development [Lamsweerde, 2000a; Lamsweerde, 2000b; Nuseibeh and Easterbrook, 2001].

On the other hand, the histories of a variety of software projects have revealed that inadequate, inconsistent, incomplete, or ambiguous requirements are numerous and have a critical impact on the quality of the resulting software [Bell and Thayer, 1976]. Late correction of requirements errors was observed to be incredibly expensive. A survey over 8000 projects in 350 US companies showed that only 16% of the projects were considered to be successful; 33% of them were considered to be failures; while 51% were considered to be neither successful nor failures, providing only partial functionalities, with major cost overruns and late deliveries [Lamsweerde, 2000a]. An independent survey of 3800 European organisations in 17 countries showed that half of the managers surveyed mentioned requirement specification and requirements management as in the first positions on their lists of main software problems [EURO96].

According to the Standish Group's CHAOS report, five of the top eight reasons why software development projects fail are related to requirements [STANDISH, 1994]:

- **Lack of user involvement**

  The CHAOS report reveals that in many a software development project, lack of stakeholder involvement was the number one contributor to project failure. Hence, involving key stakeholders in RE will avoid problems that result from differing visions and different interpretations of what should be included or excluded in the software delivery.

- **Incomplete requirements**

  Incomplete requirements cause software failure because incomplete requirements do not accurately describe the intended software's features and functionalities. Due to incomplete requirements, the software is not prepared to deal with the expected situations, or deals with the expected situations in an unexpected way. It is not possible to write complete requirements for a system without a clear understanding of the system's scope, mission, and operational concepts.

- **Unrealistic customer expectations**

  Unrealistic customer expectations contribute to overrunning a software development project's cost and time budgets. As a result, a RA has to perform additional analysis in order to prevent *gold plating* [1] [pp 194-195 and pp314, Robertson and Robertson, 1999] and to determine the customer's real needs and expectations. Hence, only requirements that are necessary, attainable, and verifiable should appear in the specification.

- **Changing requirements and specifications**

  Changing requirements is as certain as death and taxes. Requirements evolve due to changing requirements of current software functionality or the need for new software functionality. For example, in the anticipation of possible problems at the beginning of the year 2000, many programs had to be changed. By creating a

[1] *Gold plating* refers to any requirement that contributes more to the cost of a software system than it does to the benefit of the software system, regardless of the origin of the requirement. A gold plated requirement may come from any stakeholder, including the client, a user, the requirements analyst, and a developer. In essence, all stakeholders must decide carefully if a gold plated requirement is really worth the cost to implement.

minimal, obtainable base level of requirements and developing those features, the effect of changes can be reduced. Furthermore, delivering minimal features allows user to quickly see software delivery.

Although the CHAOS Chronicles [STANDISH, 2004] and the Extreme CHAOS [STANDISH, 2001] Reports have shown major improvements since the first CHAOS Report [STANDISH, 1994], project successes have increased to only just over 34%, project failures have decreased to only 15%, and challenged projects account for the remaining 51% [Software Magazine, 2004]. The Extreme CHAOS Report concludes that the lack of user involvement, which traditionally was the number one reason for project failure, has become the number one reason for project success [Hartmann, 2006].

Also Macaulay [Macaulay, 1996] has identified some possible causes of development failures, as illustrated in Table 1.

Possible cause / Type of failure	Lack of a systematic process	Poor communication between people	Lack of appropriate knowledge or shared understanding	Inappropriate, incomplete or inaccurate documentation	Poor management of people or resources
Process	O	O			O
Interaction		O	O	O	
Expectation		O	O	O	

Table 1. Causes of Process Failure [Macaulay, 1996]

Table 1 shows that process failure is linked to three possible causes: (i) lack of a systematic RE process, (ii) poor communication between people, and (iii) poor management of people and resources. Process failure relates to an information system development process in which budget, time, or other resource allocations have overrun to the point where all benefits expected from the proposed system have been negated, or in which the allocated resources do not result in a viable system.

On the other hand, interaction failure and expectation failure are linked to three possible causes: (i) poor communication between people, (ii) lack of appropriate knowledge or shared understanding, and (iii) inappropriate, incomplete or inaccurate documentation. Interaction failure is the argument that a low level of use of the system can be interpreted as failure. Expectation failure is simply when the system has failed to meet the expectations of at least one stakeholder group.

RE is a critical phase in which over-idealisation of goals, requirements, and assumptions contribute to a software development that causes inconsistencies between the specification of the proposed system and the proposed system's actual behaviour. As a consequence, software developed from those requirements will inevitably result in poor performance, sometimes with hazardous consequences on the environment [Lamsweerde and Letier, 1998].

2.2.2 Activities involved in RE

RE was traditionally considered to be restricted to a particular phase of the software development life cycle, which would normally occur before design, implementation, testing and utilisation. However, this restrictive view of requirements has evolved a great deal in the last two decades. Some of the activities that were traditionally thought of as design, such as a feasibility study, have become recognised as crucial to RE. Moreover, it is generally accepted that the requirements phase is not confined to only the initial stage of software development because requirements are continually being refined throughout the life cycle.

Activities in RE are diverse in nature and approach. Each of these activities may present those involved in developing and managing requirements with different kinds of problems. In the RE literature, many different definitions have been given for the RE process as well as for the activities that take place during RE life cycle [Zowghi, 2007]. The activities in the RE life-cycle are illustrated in Figure 1 and are discussed briefly in the subsequent subsections.

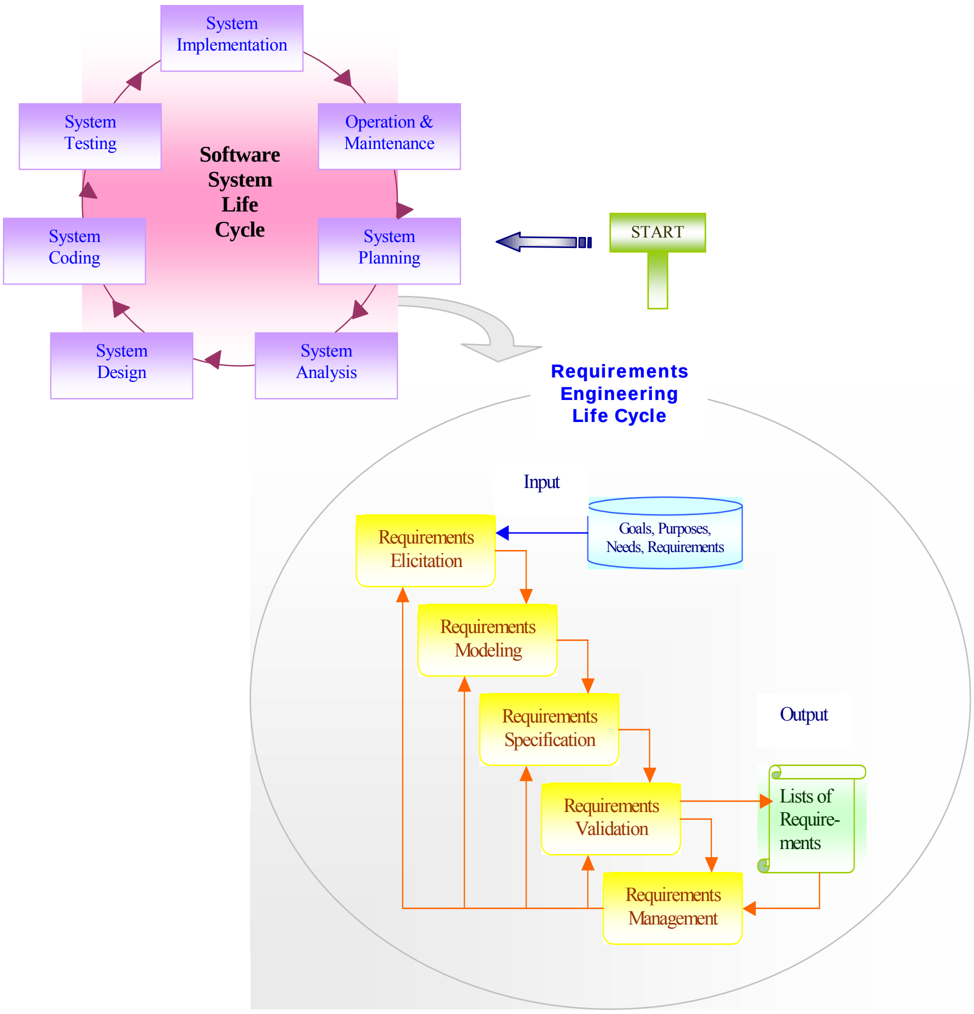

Figure 1, shown above, depicts the life cycle of a software system development and the activities of RE incorporated in it. Customarily, the RE for a system can be an independent process with its own plan, budget, schedule, and statement of work. The RE process studies the world to devise the best possible requirements to be used in the software system development. As an outcome of this process, a list of requirements, which is the complete description of the functionalities and behaviours of the proposed software system, is produced. However, once the system is built and is being used, inevitably new requirements will emerge, and the system will go through the RE life cycle again.

- **Requirements Elicitation**

Requirements elicitation is fundamentally a human activity and is when stakeholders are identified and relationships are established between the development team and the customer. It is the process of discovering and eliciting all of the proposed system's requirements from stakeholders. There are several techniques used in requirements elicitation: interview, questionnaire, apprenticing, observation, requirements workshop, etc. Each technique has its own strengths and weaknesses, and it is common to use several techniques together in requirements elicitation. To elicit requirements, the RA needs to have an understanding of the problem to be solved, the business process in the customer's organisation, the way the system will be used, and the application domain of the system.

- **Requirements Modeling**

Requirements modelling is the process of transforming requirements gathered during requirements elicitation into a model of the target system. This modelling process facilitates the transition from an understanding of what currently exists to a specification of what is wanted. In addition, the RA may elaborate alternative models for the target system and produce a conceptual model of the enterprise for the system's future user.

Because of the varieties of requirements modelling techniques ranging from formal methods, object-oriented analysis (OOA), use of Unified Modeling Language (UML), and the Structured Systems Analysis and Design Method (SSADM) [Macaulay, 1996], there is no single modelling technique that is applicable to all types of software systems. Therefore, a RA needs to understand many different types of modeling techniques so that he or she can choose the one that is best to design the application.

- **Requirements Specification**

Requirements specifiction is a crucial activity in system development, in which RA precisely describes the target system's components and behaviour. The resulting RS is to act as a contract between the customer and the developers. Writing a correct and precise RS is not easy, because it must [Lamsweerde, 2000c; Meyer, 1985]:

- adequately state the problem at hand,
- have a meaningful semantic interpretation that makes true all specified properties taken together,
- be unambiguous, so that it does not have multiple interpretations, and
- be minimal, not to state properties that are irrelevant to the problem and not to state properties that are relevant to only one or more solutions for that problem.

To guide the writing of a RS, the IEEE has defined the IEEE Std. 830-1998 [IEEE, 1998], which contains eight criteria for a Software Requirements Specification (SRS) to be considered acceptable are correctness, completeness, consistency, verifiability, prioritorisation, traceability, modifiability, and lack of ambiguity.

- **Requirements Validation**

Requirements validation aims to check the accuracy, correctness, and completeness of a RS. A RS must represent the entire software system to be built. Each requirement should be technically feasible and should fit within budget, schedule, and other constraints.

Figure 2 illustrates the activities involved in requirements validation. Each unvalidated requirement will undergo requirements management. One worthnoting point on requirements inspection is that many consider inspection to be a form of testing, particularly if the requirements specification is not executable, in a sense the inspection process consists of a mental execution of the specification [Gilb and Graham, 1993; Ebenau and Strauss, 1994]. Eventually, each of the validated requirements will be used in the later system design process.

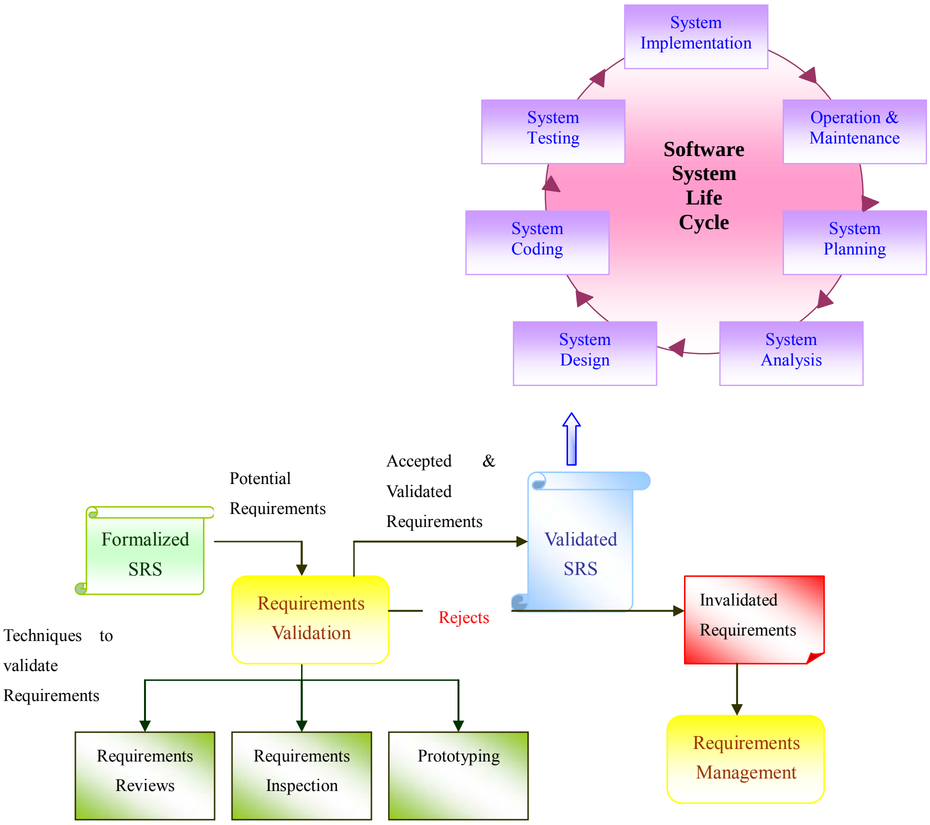

Figure 2. Requirements Validation process

There are several ways to conduct requirements validation, which namely by:

(i) Prototyping to avoid the wastage of resources caused by trying to satisfy erroneous requirements.

(ii) Model validation to perform a static analysis to verify that communication paths exist between, for example the object models, in the stakeholders domain, exchange data.

(iii) Acceptance test via test cases developed from the requirements specifications, to describe an observable feature of a system which is of interest to the stakeholders.

◆ **Requirements Management**

Requirements management (RM) is the process of handling and managing changes of software requirements in a RE process. Often, changes occur due to new technological advances or to a change in the business operation of the stakeholders. Any change may require redoing the overall RE process.

It is believed that the activities involved in RM are the activities in the RE life cycle. As shown in Figure 3, whenever there are changes to the requirements, the RE life cycle has to be repeated as new requirements are elicited, modeled, specified, and validated.

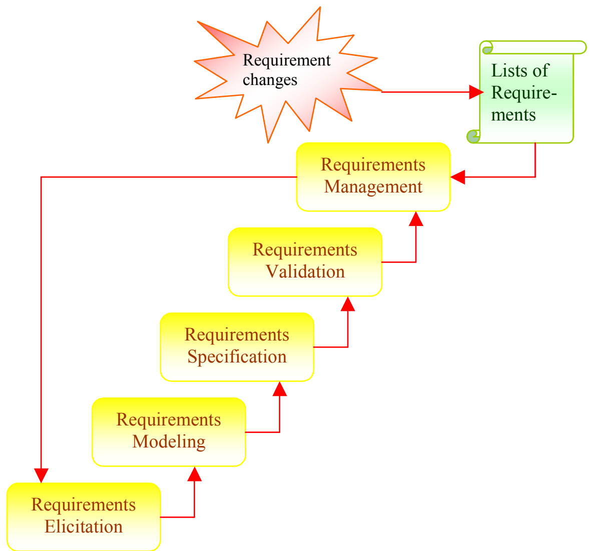

Figure 3. The life-cycle of Requirements Management

2.3 Requirements represented in NL

The overwhelming majority of RSs are written in NL, although often amplified by information in other notations, such as formulae and diagrams. NL has many serious shortcomings as a medium for description. Because of NL's inherent ambiguity, it is very hard to describe complex concepts precisely in a NL.

Interest in resolving ambiguity in RSs started as early as 1980s when it was realized that ambiguity relates also to eliciting a project's real requirements [Gause and Weinberg, 1989]. Gause and Weinberg discuss all aspects of the requirements process: how to cover all requirements and how to detect and remove ambiguity in requirements. Gause was among the first to mention the phenomenon of subconscious disambiguation. Many an ambiguity is not noticed, because a reader subconsciously disambiguates an undetected ambiguity to his or her first interpretation. The reader is not aware that there is an interpretation other than the one that came first to his or her mind.

FLs such as Z, CSP, etc. avoid the ambiguity problem because of their syntax and semantics are formally defined. However, many a FL is not expressive enough to adequately describe every system aspect. In contrast, a NL is extraordinarily rich and is able to describe, although imperfectly, many concepts or system properties [Sawyer and Kotonya, 2001].

Another issue to consider when writing a FL RS from a NLRS, whether written or spoken, is that an incorrectly understood NLRS leads to an incorrect FL RS. Furthermore, difficulties caused by lack of understanding of the real world situation are not eliminated by the use of a formal method. Instead, the misunderstanding gets formalised into specifications, and may even be harder to recognise simply because formal definitions are harder to read by the clients. Hence, a FL RS is not inherently a solution to an ambiguous NLRS.

Sommerville [Sommerville, 2001] discusses why a NLRS may not be a particularly good basis for either a design or a contract between customer and system developer. There are several reasons for this:

1. Understanding a NLRS relies on the requirements readers' and writers' using the same terms for the same concept. However, often, the use of a term leads to more than one plausible interpretation because of the inherent ambiguity of NL words.
2. A NLRS is often over-flexible because one can say the same thing in completely different ways. Interpreting the author's intent requires the reader to fully understand the context of NLRS. Thus, deciding when requirements are the same and when the requirements are distinct can be an error-prone process.
3. Requirements are generally not partitioned effectively by a NL. Hence, to discover the consequence of a change in a RS, one may have to analyse every RStat rather than just a group of related RStats.

The problems that Sommerville discusses are the issues to consider when writing a a RS in NL. The greatest challenge is still the inherent ambiguity of NL, which is sometimes not noticed until the design or implementation phases of the software process, when it is very expensive to resolve.

3 Ambiguity in NLRS

"There is no sense of being exact about something if you don't even know what you're talking about." – John von Neumann

According to Oxford English Dictionary, the 500 words used most in English language each of them has an average of 23 different meanings. For example, round has 70 distinctly different meanings. The wide variation in word meanings in NL has always posed problems for anyone who attempts to construct an unambiguous and precise statement. It is often the case that different individuals interpret a written statement in different ways.

3.1 Definition of Ambiguity

Ambiguity in NL is a major problem in scientific disciplines. People who use NL can usually discern the intended meanings of otherwise ambiguous words and phrases by using various sources of knowledge. This ability allows human to be efficient in the use of language. Ambiguity is noticed when a statement has more than one distinct meaning. By contrast, a vague statement has only one meaning, but the distinction between the circumstances under which it is true and the circumstances under which it is false is not clear-cut [Nissanke, 1999]. This thesis concentrates on the discussion to deal with language ambiguities, particularly ambiguities found in NLRSs. An ambiguous requirement can have negative consequences for its software development project as a whole, in which a wrongly implemented requirement causes high costs for rework and delayed product releases.

According to Anderman and Rogers [2005], ambiguity arises from:

- a linguistic or syntactic problem such as polysemy, incorrect word order, logical connectives, incorrect collocations, use of quantifiers, articles, number, synonyms, use of tenses, and speech act conventions,
- a conceptual problem such as conceptual vagueness or a lack of correspondence, and
- an invariant metaphorical interpretation that requires the understanding of notions of events, properties, and relations from the source domain to the target domain

On the other hand, computational linguists have classified several types of ambiguity occurring in NL utterances [Nilsson, 1998],

a) *lexical ambiguity* of a word, in which the word can have several different meanings: Lexical Ambiguity can be divided into homonymy and polysemy. Homonymy of a word occurs when the word has unrelated meanings and different etymologies. An example of a homonymy is bank. According to WordNet 3.0 [Miller et al., 2006], bank may mean a depository financial institution; sloping land; a flight maneuver; or many others.

   Polysemy of a word occurs when the word has several related meanings but one etymology. For example, green suffers from polysemy. According to WordNet, among others, green may mean the colour green or unripe.

b) *Syntactic ambiguity* or *structural ambiguity* of a sentence arises when the sentence can be parsed in more than one way, resulting in more than one grammatical structure, each with a different meaning. An example of a syntactic ambiguity that is known also as an attachment ambiguity is:

   E1: I saw the girl with the telescope.

   where the prepositional phrase (PP) with the telescope can modify either the noun phrase the girl or the verb phrase saw the girl. The first interpretation means that the girl had a telescope with her. The second interpretation means that the equipment used to see the girl was a telescope.

c) *Semantic ambiguity* or *scope ambiguity* of a sentence arises when there is more than one interpretation to the sentence within its context, although the sentence contains no lexical, syntactic, or structural ambiguity. For example,

   E2: All lights share a switch.

   The use of universal quantifier all leads to a scope ambiguity. When the scope of a includes the scope of all, the interpretation is that each light has its own switch. When the scope of all includes the scope of a, the interpretation is that all lights share one switch.

d) *Referential Ambiguity* is the ambiguity caused from the use of pronoun or anaphora in which the pronoun or anaphora can have more than one antecedent. For example,

   E3: The trucks shall treat the roads before they freeze.

   The antecedent of the anaphor they can be either trucks or roads. An anaphor can refer also to a set of objects, a compound object, or a verb.

e) *Pragmatic Ambiguity* concerns the relationship between the meaning of a sentence and the context in which the sentence occurs. Sometimes pragmatic ambiguity results when human common sense knowledge and knowledge about context is uncertain. The difference between pragmatics and semantics is that pragmatics is concerned with context-dependent meaning whereas semantics is concerned with context-invariant meaning [Levinson, 1983]. Regardless of where an ambiguity arises from, it is often compensated by:

- speakers' conventions that disambiguate using contextual, inter-textual, and language systemic information and
- readers' assumptions based on known facts about communicated events; alhough a reader's assumptions may be subjectively accurate according to his or her knowledge about the communicated events, the prior knowledge may be misguided or erroneous, leading to wrong assumptions.

In understanding ambiguous sentences, a human often overlooks the ambiguity and instinctively immediately subconsciously disambiguates. When the human processes a sentence, there is an interaction between the semantics of each word and the evolving interpretation of the current sentence. Each word primes the interpretation according to existing knowledge with which the word has been associated to the current context in the past. Hence in final interpretation, all the past contexts with their constituent words are combined. So when the past contexts do not signify ambiguity in their constituent words, subconscious disambiguation results, because all meanings of an ambiguous word are activated in the sense that they are an inherent part of the representation of the word. Therefore, the human has to carefully analyse each occurrence of ambiguity and to involve the writer to determine exactly what the writer means to avoid subconscious disambiguation.

3.2 The nature of NL Ambiguity in Quantifier, Coordinator, and Others

This section contains discussion on the kind of potential ambiguities resulting from quantifiers, conjunctive and, disjunctive or, only, also, but, unless, and if. Each grammatical context in which a conjunctive and is placed carries a different interpretation. Likewise, for a disjunctive or, some contexts favour the inclusive or and some contexts favour the exclusive or interpretation. The following discussion covers also the temporal involvement in the interpretation of unless and if.

3.2.1 Quantifier

Quantifiers, coordinators, and negations, to name a few, are examples of logical operators. Some ambiguities involve quantifier scoping. Each and every share a single syntactic structure but each's semantic structure can be ambiguous. For example, Every boy loves a dog. carries more than one interpretation, in which there is a one particular dog that all boys love or every boy loves one different dog.

A quantifier varies also with respect to vagueness. All may refer to the whole collection of elements or may refer to each element of the collection. Any can be interpreted as an existential quantifier instead of as a universal quantifier. Other quantifiers such as many, some, and few are vague because it is difficult to determine the size of the identified set.

3.2.2 Coordinator

English connectives or coordinators such as and, or, but, unless, if then, and if and only if, have always been problematic because they are not always truth functional. Using one of these coordinators as the main connective in a compound sentence, the truth value of the resulting sentence does not depend in all cases solely on the truth values of the component sentences [Cohen, 2004]. Thus, one way to see whether a coordinator is a truth-functional or not is by constructing a truth table for the compound sentence where the coordinator acts as the main connective. The truth table projects the truth value of both component sentences being compatible with either the truth or the falsity of the entire compound sentence.

Haspelmath [2004] defines coordination as: "Coordinating constructions can be identified on the basis of their symmetry: A construction {*A* *B*} is considered coordinate if the two parts *A* and *B* have the same status (in some sense that needs to be specified further), whereas it is not coordinate if it is asymmetrical and one of the parts is clearly more salient or important, while the other part is in some sense subordinate.". Following the first part of his definition shown above, Haspelmath continues: "In practice, we typically suspect that a construction will be coordinate if it is systematically used to render English constructions with the coordinating particles and, or, and but."

Coordination ambiguity is an ambiguity that derives multiple interpretations result from the different ways in which a sequence of words containing a coordinator can be grammatically structured [Chantree, 2005]. Literally, coordination ambiguity results from the use of coordinator to connect words, phrases, or clauses that have the same grammatical structure within a sentence. This thesis focuses majorly on the coordination ambiguity results from the use of coordinators such as conjunctive and and disjunctive or, minorly on only, also, unless, and conditional if statement. Coordinators such as conjunctive and and disjunctive or are the most common cause of coordination ambiguity and account for approximately 3% of the words in the British National Corpus (BNC) [Chantree et al., 2006].

3.2.2.1 Conjunctive and

Conjunctive and frequently corresponds to the logical operator & and is often used to coordinate words and phrases with external modifier being a word or phrase appearing either before or after the coordinator as in regulatory and utility technical requirements where the external modifier technical requirements may apply either to both regulatory and utility or to just utility. Applying technical requirements to both regulatory and utility derives a coordination-first reading. Applying technical requirements to only utility derives a coordination-last reading [Chantree et al., 2006]. Interestingly, conjunctive and involves also temporal sequencing as in The computer closes all programs and shuts down., in which the action of closing the opened programs in the computer precedes the shutting down action.

Conjunctive and is very ambiguous because when and is used to combine words or phrases, and can mean more than one interpretation, including:

- to suggest that one idea chronologically follows another,
- to suggest that one idea results from another,
- to suggest that one idea contrasts with another, in which case, and is frequently replaced by but,

- to suggest an element of surprise, in which case, and is frequently replaced by yet,
- to suggest that one clause is dependent on the preceding clause, and
- to suggest a commentary on the preceding clause.

Shortcomings in the analysis of a conjunctive and in the reading of a RS are largely due to the requirements analyst's having paid insufficient attention to grammatical context. The involved domain, where in the RS the and coordination occurs, what part of speech occurs on either side of the coordination, and other such factors, are relevant in determining the meaning associated with conjunctive and. Determining how these various elements interact is challenging. Therefore, a requirement analyst should focus on writing a less ambiguous RS, avoiding the inherent ambiguity caused by the use of a conjunctive and.

A conjunctive and paired with nouns can convey that all the elements of a set are to be considered together, but and can convey also that each element of the set is to be considered separately. The truth is that a conjunctive and can potentially cause problems, regardless of where the and is positioned in a sentence. The following section discusses the combination of roles that conjunctive and may serve in a RStat along with resulting potential ambiguity for each role.

- Subject Ambiguity

  When nouns linked by and constitute the subject of an RStat, it can be unclear whether the person or entity constituting the subject is to be considered individually as in E4.1, or collectively as in E4.2.

E4: The manager and the database administrator shall monitor every access to the database.

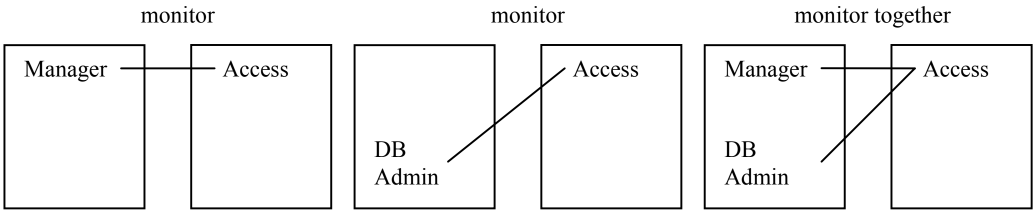

Figure 4. E4.1 and E4.2 Illustrations

E4.1: Each of the manager and the database administrator shall monitor every access to the database.

E4.2: The Manager and The Database Administrator shall together monitor every access to the database.

- Direct Object Ambiguity

A similar range of potential meanings arises when nouns linked by and are other than the subject of the sentence. E5 shows example where nouns linked by and serve as direct objects and Table 2.shows the derivation of decision table for E5.

Data / Time (t)	LVL0	LVL1	LVL0 and LVL1	Either LVL0 or LVL1
tn	True	True	True	False
tn	True	False	False	True
tn+1	False	True	False	True
tn+1	False	False	False	False

Table 2. Decision table for E5.2 and E5.3

E5: The system shall receive LVL0 and LVL1 data.

E5.1: The system shall receive LVL0 data and the system shall receive LVL1 data.

E5.2: The system shall receive both LVL0 data and LVL1 data, presumably one after another at different times.

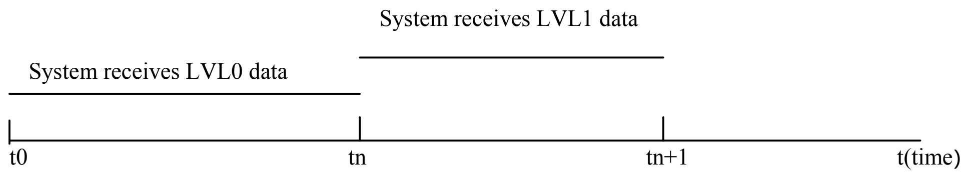

Figure 5. E5.2 Illustration

E5.3: The system shall receive both LVL0 data and LVL1 data together at the same time.

Figure 6. E5.3 Illustration

The following examples E6 and E7 show potential ambiguity caused by negation. E6.1 and E7.1 convey the intended interpretations of E6 and E7. E6.1 describes that the system shall not receive both LVL0 data and LVL1 data regardless of whether or not LVL0 data arrives at the same time as LVL1 data. E6.2 describes the system shall receive each data at one time but not both data concurrently at the same time.

E6: The system shall not receive LVL0 and LVL1 data.

E6.1: The system shall not receive LVL0 data and the system shall not receive LVL1 data.

E6.2: The system shall not receive LVL0 data and LVL1 data together, but shall receive LVL0 data or LVL1 data at different time.

E7: The system shall not log LVL0 data and LVL1 data.

E7.1: The system shall not log LVL0 data and the system shall not log LVL1 data.

E7.2 describes that the system shall not record both LVL0 and LVL1 data at the same time because the data may collapse each other, and instead to record the data one by one at one time.

E7.2: The system shall not log LVL0 data and LVL1 data together, but shall log LVL0 data or LVL1 data at different time.

- Subject-and-Direct-Object Ambiguity

E8 demonstrates a mixture of Subject ambiguity and Direct-Object potential ambiguity. It is unclear whether and coordination of the subjects is distributive as in E8.1 or collective as in E8.2. In other words, it is unclear whether each of the subjects is to notify one or both of the objects.

E8: The manager and the database administrator shall monitor every access and intrusion to the database.

E8.1: Each of the manager and the database administrator shall monitor every access to the database and every intrusion to the database.

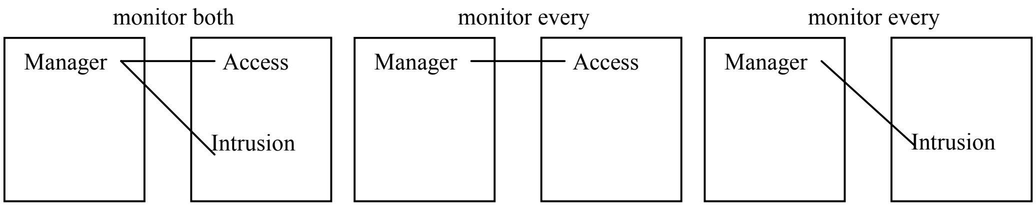

Figure 7. E8.1.1 and E8.1.2 Illustrations

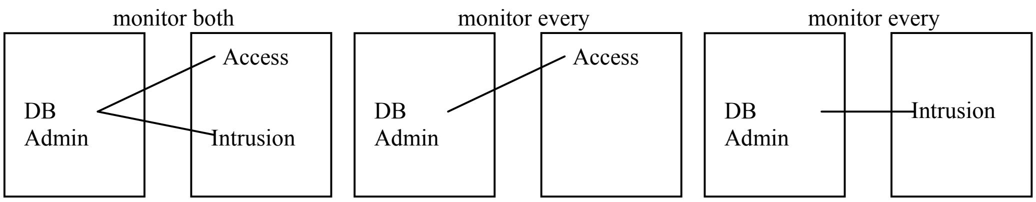

Figure 8. E8.1.3 and E8.1.4 Illustrations

Distributively, E8.1 can be further derived into several possible interpretations which are:

E8.1.1: The manager shall monitor every access to the database.

E8.1.2: The manager shall monitor every intrusion to the database.

E8.1.3: The database administrator shall monitor every access to the database.

E8.1.4: The database administrator shall monitor every intrusion to the database.

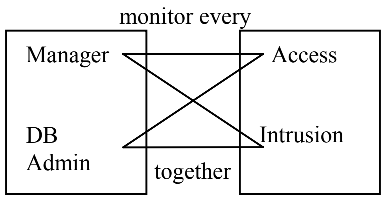

Figure 9. E8.2 Illustration

E8.2: The manager and the database administrator shall together monitor every access to the database and every intrusion to the database.

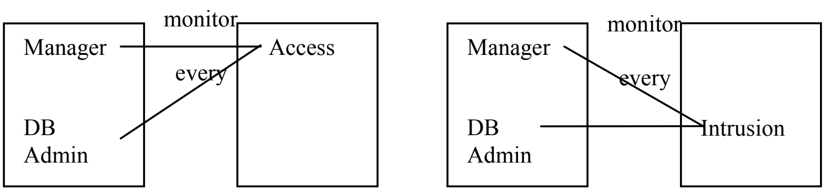

Figure 10. E8.2.1 and E8.2.2 Illustrations

Collectively, E8.2 can be further derived into two possible interpretations:

E8.2.1: The manager and the database administrator shall together monitor every access to the database.

E8.2.2: The manager and the database administrator shall together monitor every intrusion to the database.

- Multiple Verb Phrases

A variant of potential ambiguity presents when there is more than one verb phrase is used as shown in E9 and E9.1.

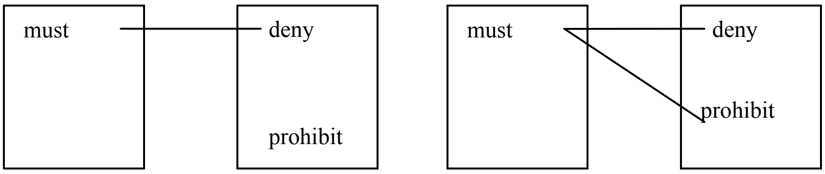

Figure 11. E9 and E9.1 Illustrations

E9: The system must deny unidentified source and prohibit unauthorised access to the database.

E9.1: The system must deny unidentified source to the database and the system must prohibit unauthorised access to the database.

However, when the first verb phrase logically leads to the second, it is likely that the second verb phrase is sensed together to execute right after the first verb phrase as in E10.

E10: The system shall terminate a login session and log a session report.

- Object-Predicative Ambiguity

When conjunctive and separates nouns play a role as object-predicative of an RStat as in E11 and E12, the potential ambiguity involved can be analogous to that is exhibited in E11.1, E11.2, E12.1 and E12.2.

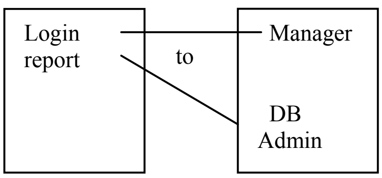

Figure 12. E10 Illustration

E11: The system shall print a login session report to the manager and the database administrator.

E11.1 is one possible interpretation that says printing the report one copy to the manager and another copy to the database administrator.

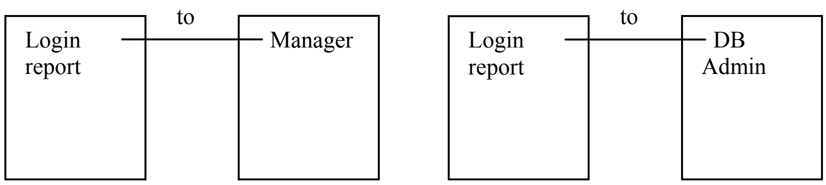

Figure 13. E11.1 Illustration

E11.1: The system shall print a login session report each to the manager and to the database administrator.

However, E11.2 describes a requirement to print only one login session report to be shared by both Manager and Database Administrator.

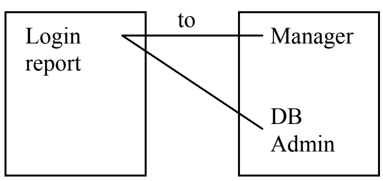

Figure 14. E11.2 Illustration

E11.2: The system shall print a login session report for jointly used to the manager and the database administrator.

As in the case of negation, E12 shows a potentially ambiguous example with negated verb phrase modifying the object predicate.

E12: The system shall not print a login session report to the manager and the database Administrator.

If E12 means the system shall not print the report to both the manager and the database administrator, but may print to other user, then E12.1 best describes the intended meaning.

E12.1: The system shall not print a login session report to the manager and the system shall not print a login session report to the database administrator.

On the other hand, if E12 means to say the system shall not print only one copy of the login session report to be shared jointly by Manager and Database Administrator, then E12.2 best describes the interpretation. An alternate representation of E12.2 is E11.1.

E12.2: The system shall not print a jointly used login session report to the manager and the database administrator.

- Adjective Ambiguity

Another form of potential ambiguity associated with and is resulting from (1) adjective that modifies a noun and are linked by and as in online and offline environment and (2) nouns that are modified by adjectives and linked by and as in online environment and offline environment.

Thus, the potential interpretations of a noun environment modified by two adjectives online and offline are:

i. online and offline environment

ii. environment which is online and environment which is offline

iii. environment which is both online and offline

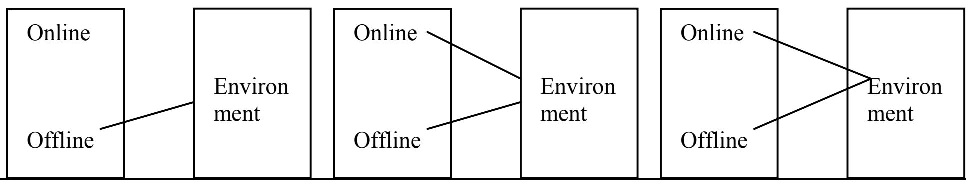

Figure 15. i, ii, and iii illustrations (from left to right)

The potential ambiguity that actually arises in a provision using a noun modified by adjectives joined by and is, however, a function of context and of which kind of domain language used. Besides, when potential ambiguity associated with plural nouns, namely (1) whether the elements of a group are acting or being acted on, individually or collectively, and (2) if the elements of a group are acting or being acted upon individually, whether they must all act, or be acted upon, in unison. In lieu of giving examples of each of the permutations, below are the examples.

E13: The system shall be available on the online and offline environment.

E13.1: The system shall be available on the environment which is online and on the environment which is offline.

E13.2: The system shall be available on the environment which supports altogether online mode and offline mode.

An alternative to having a noun modified by two or more adjectives is to repeat the noun with each adjective as in online environment and offline environment. Doing so prevents the potential ambiguity exhibited in E13.2 when, presumably the environment is unlikely to support both online mode and offline mode at the same time. Another noteworthy example is temporary and part-time employees with the possible interpretations arise: (a) employees who are temporary and employees who are part-time, (b) employees, each of whom is both temporary and part-time. (a) is potentially ambiguous because it is difficult to determine whether an employee who is both temporary and part-time would fall within both the group of temporary employees and the group of part-time employees, or, would be excluded from each group for having both attributes. One simply has no basis for concluding that an employee who is both temporary and part-time would be excluded from both groups.

- The ambiguity of Every *X* and *Y*

The use of every before two or more nouns that are linked by and introduces ambiguity. E14 is potentially ambiguous because it is difficult to determine whether:

i. every manager and database administrator, presumably John, Mary, and Jack are database administrators but only John is appointed in having the login session report.

ii. every manager and every database administrator is entitled to have the login session report

iii. only a person who have both roles of Manager and Database Administrator is entitled to have the login session report.

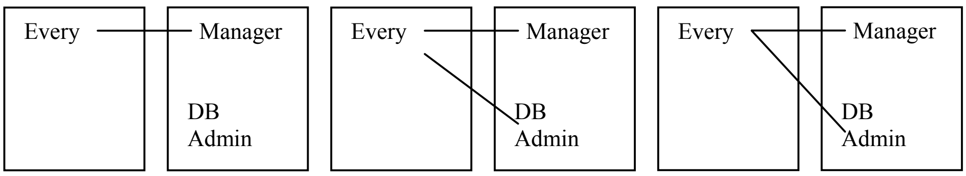

Figure 16. i, ii, and iii illustrations (from left to right)

E14: The system shall print a login session report to every Manager and Database Administrator.

Potential ambiguity arises from E14 leads to E14.1 interpretation or E14.2 interpretation.

E14.1: The system shall print a login session report to every Manager and every Database Administrator.

E14.2: The system shall print a login session report to every person who is both a Manager and a Database Administrator.

The above discussion presents the potential ambiguity caused by and in various grammatical contexts. The use of and is potentially ambiguous, particularly when and can be interpreted as or. It is best for the RA to critically analyse any use of and and or for potential ambiguity, and to do so in a way that reflects client's understanding of the requirements domain.

3.2.2.2 Disjunctive or

Latin has two different disjunctive words, vel for inclusive disjunction and aut for exclusive disjunction. In contrast, English has only one word, or, for disjunction to introduce alternatives. In writing, or does double duty as both inclusive disjunction and exclusive disjunction. However, no instance of or serves as both inclusive disjunction and exclusive disjunction at the same time. Often, the interpretation of or is inclusive as is stated in the Cambridge Grammar of the English Language (CGEL). However, a statement in the form of *P* or *Q* is typically interpreted also as *P* and *Q* are not both true, because we don't generally say *P* or *Q* if we know both *P* and *Q* to be true. The most likely reason for saying *P* or *Q* rather than *P* and *Q* is that the latter would be false or uncertain. For example, when the intend is to invite both Tom and Jack to dinner, one says She invites Tom and Jack to dinner, because it is misleading to say She invites Tom or Jack to dinner. The phrase a $500 fine or ten days in jail is better suited to assessing the meaning of or. To be able to impose both the fine and the jail term, one would need to add or both to the end of the phrase. The statement, Would you like coffee or tea? has an exclusive of or as ordering both coffee and tea for one person would be decidedly eccentric. However, Would you like milk or sugar? has an inclusive or because having one's coffee or tea with both milk and sugar is such a standard alternative to having just milk or just sugar. Context and domain knowledge are necessary to classify any given or. In requirements engineering, each or ultimately requires asking the client its meaning.

For a proper understanding of the ambiguity associated with disjunctive or, one must explore the different grammatical contexts in which or is used. The ambiguity of disjunctive or is of two categories. First, an ambiguity arises when a plural noun is associated with an or coordination, because it is unclear whether (1) all the elements in the sentence are to be attributed to one coordinate or (2) all the elements in the sentence are to be attributed to the other coordinate or (3) can be divided between the coordinates such as in The stocks are obtainable at Wal-mart or Carrefour where both Wal-mart and Carrefour have the stocks and one has a choice between obtaining them at Wal-mart or obtaining them at Carrefour. Second, ambiguity occurs in the context of negation.

As does a conjunctive and, also a disjunctive or potentially causes problems, regardless of where the or is positioned in its sentence. The following discusses the combination of roles disjunctive or may serve in a RStat along with its resulting corresponding potential ambiguity for each role.

- Subject Ambiguity

  When the disjunctive or links more than one nouns that constitute as the subject of a sentence as in E15, the potential ambiguity associated is to decide whether only one of the noun is acting or being acted, or both nouns jointly are acting or being acted. For example in E15, it is difficult to determine whether (1) not both of the manager and the database administrator but only one of them monitors every access to database to avoid role redundancy as in E15.1, or (2) both of the manager and the database administrator have to monitor every access together as in E15.2, or even (3) each of the manager and the database admin shall each monitor every access separately as in E15.3. Note that conjunctive and replaces or in E15.2 to represent the idea of inclusive or precisely. Table 3. derives possible interpretations for E15.

Manager monitors	DB Admin monitors	Both (Manager and DB Admin) monitors	(Manager or DB Admin) but not both of them monitor
True	True	True	False
True	False	False	True
False	True	False	True
False	False	False	False

Table 3. Derivation on the possible interpretations for E15

E15: The manager or the database administrator shall monitor every access to the database.

E15.1: The manager or the database administrator, one but not both of them shall monitor every access to the database.

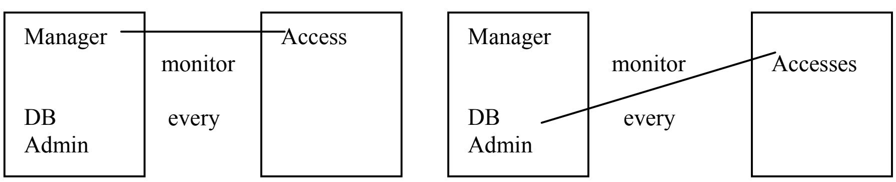

Figure 17. E15.1 Illustration

Another way of representing E15.1 is E15.1.1 is by using either[2] because either suggests choosing only one option from the available options.

E15.1.1: Either the manager or the database administrator shall monitor every access to the database.

E15.2: The manager and the database administrator shall monitor together every access to the database.

E15.3: Each of the manager or the database administrator shall monitor every access to the database separately.

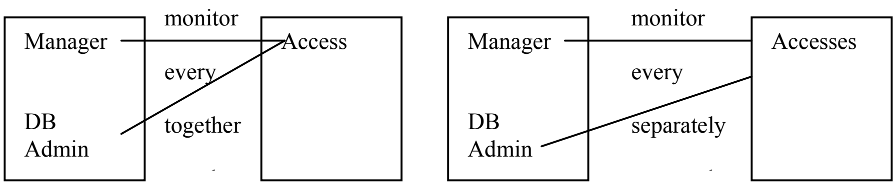

Figure 18. E15.2 and E15.3 Illustrations

E16 shows an example when the direct object is a plural noun as in accesses.

E16: The manager or the database administrator shall monitor accesses to the database.

then the ambiguities involve are:

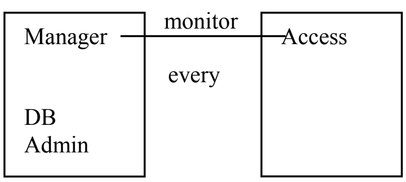

Figure 19. E16.1 Illustration

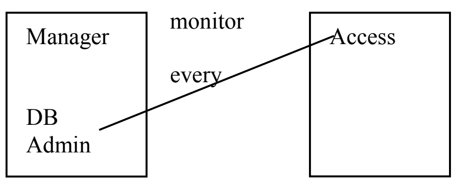

Figure 20. E16.2 Illustration

[2] This use of either is not excluded by Rule VI.3 (see Chapter 6 for Guiding Rules), which suggests avoiding either as it introduces subjective option to be chosen from available options.

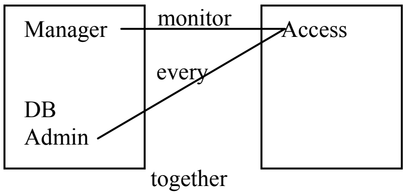

Figure 21. E16.3 Illustration

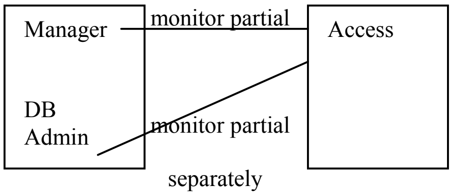

Figure 22. E16.4 Illustration

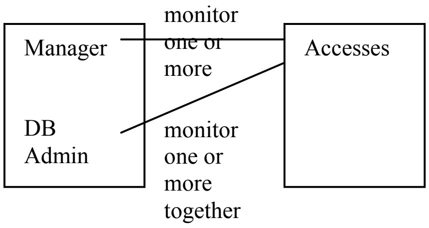

Figure 23. E16.5 Illustration

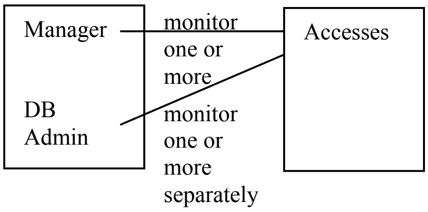

Figure 24. E16.6 Illustration

E16.1: The manager shall monitor every access to the database.

E16.2: The database administrator shall monitor every access to the database.

E16.3: The manager and database administrator shall monitor every access to the database together.

E16.4: Each of the manager and the database administrator shall monitor partial accesses to the database separately.

E16.5: The manager shall monitor one or more accesses to the database and the database administrator shall monitor one or more accesses to the database, together at the same time.

E16.6: The manager shall monitor one or more accesses to the database and the database administrator shall monitor one or more accesses to the database, separately.

Thus, when one or more of the nouns linked by or is a plural noun, the potential ambiguity will be compounded by that associated with plural noun.

- Direct Object Ambiguity

When the direct object consists of singular nouns separated by disjunctive or as demonstrated in E17 and further derived in Table 4., the potential ambiguity involves the difficulty to decide whether the system shall receive both LVL0 data and LVL1 data where each arrives at different time as in E17.1, or the system shall receive both LVL0 data and LVL1 data altogether at the same time as in E17.2, or only one either LVL0 data or LVL1 data regardless of the arrival time as in E17.3.

E17: The system shall receive LVL0 or LVL1 data.

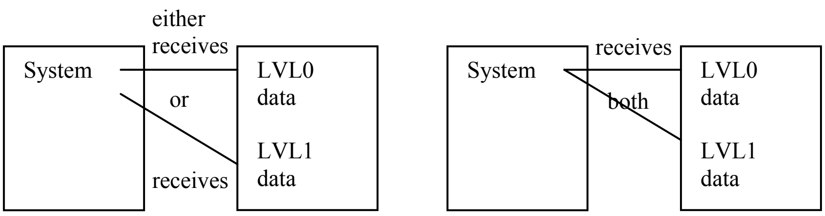

Figure 25. E17.1 and E17.2 Illustrations

E17.1: The system shall receive either LVL0 data or LVL1 data.

E17.2: The system shall receive LVL0 data or LVL1 data, or both LVL0 data and LVL1 data.

E17 is also likely to convey a temporal meaning when the disjunctive or is interpreted to assert inclusive or as such represented in previous example E5.1 and E5.2, in which E5.1 and E5.2 are not equivalent at all.

receive LVL0 data	receive LVL1 data	receive LVL0 and LVL1	receive either LVL0 or LVL1
True	True	True	
True	False		True
False	True		True

False	False	False	False

Table 4. Decision Table for E17

E18 shows an example of negated RStat that is potentially ambiguous. E18.1 and E18.2 are the possible interpretations for E18.

E18: The system shall not receive LVL0 or LVL1 data.

E18.1: The system shall not receive LVL0 data and the system shall not receive LVL1 data.

E18.2: The system shall not receive one or the other of LVL0 data or LVL1 data, but may receive both of LVL0 and LVL1 data together.

- Object-Predicate Ambiguity

There are two kind of potential ambiguity reflected in E19. One kind of potential ambiguity is the indirect object of to followed by the manager or the database administrator. Another kind of potential ambiguity arises when the direct object is plural as in E19 due to uncertainty as to whether the elements constituting the direct object are to be considered individually or collectively. Table 5. derives the decision table for E19.

User / Report	Manager	DB Admin	Manager and DB Admin	Either Manager or DB Admin
all	True	True	True	
all	True	False		True
all	False	True		True
all	False	False	False	False
partial	True	True	True	

partial	True	False		True
partial	False	True		True
partial	False	False	False	False

Table 5. Decision table for E19.1, E19.2, and E19.3

E19: The system shall print login session reports to the manager or the database administrator.

E19.1: The system shall print login session reports to either Manager or Database Administrator, with only one of them have all the printed login session reports.

E19.2: The system shall print login session reports to Manager or Database Administrator or to both of them.

E19.3: The system shall print login session reports to Manager and Database Administrator, with each of them has partially divided the printed login session reports.

- Adjective Ambiguity

Likewise conjunctive and, the potential ambiguity associated with disjunctive or is from adjective that modifies a noun and is linked by disjunctive or as in online or offline environment.

Correspondingly, the potential interpretations of a noun environment modified by two adjectives online and offline are:

i. online or offline environment

ii. environment which is online or environment which is offline, or both

iii. environment which is either online or offline, but not both

Compounding to the ambiguity of disjunctive or is the use of plural nouns such as environments. Example E20 demonstrates the potential ambiguity that actually arises in provision using a plural noun modified by adjectives that are joined by or.

E20: The system shall be available on the online or offline environments.

E20.1: The system shall be available on the environments which are online, or on the environments which are offline, or environments that support both online and offline.

E20.2: The system shall be available either on the environments which are online or on the environments which are offline.

In addition to avoid ambiguity caused by compound noun when the compound noun is modified by two or more adjectives, rewrite the noun for each adjective as in online environment or offline environment.

- The Ambiguity of Every *X* or *Y*

every when followed by a disjunctive or that is used to conjunct two or more nouns gives rise to ambiguity. E21 is potentially ambiguous because it is difficult to determine whether (1) either every manager or every database administrator but not both is entitled to have the login session report as reflected in E21.1 - exclusive or interpretation, or (2) both every Manager and every Database Administrator are entitled to have the login session report as reflected in E21.2 - inclusive or interpretation.

E21: The system shall print a login session report to every Manager or Database Administrator.

E21.1: The system shall print a login session report to either every Manager or every Database Administrator, but not both.

E21.2: The system shall print a login session report to every Manager and every Database Administrator.

From the previous discussion, note that the potential ambiguity discussed are often derived from the difficulty to interpret whether the intent is inclusive or or exclusive or. In essence, disjunctive or raises two issues that require thorough consideration. First, given S_1 is one statement and S_2 is another statement, the use of either S_1 or S_2, rather than simply or, carries no syntactic ambiguity because when or pairs with either, it certainly has exclusive or interpretation.

However writing either S_1 or S_2 in a RStat contributes a semantic ambiguity and invites subjective interpretation. One has to understand the context in order to decide whether the RStat means S_1 or RStat means S_2 and certainly not referring to both of them together. Second, the use of simply S_1 or S_2 causes ambiguity because (1) it is difficult to determine whether or refers to inclusive disjunction, which means both S_1 and S_2 all together, or the or refers to exclusive disjunction, where only S_1 or only S_2, but certainly not any two together.

Even though the distinction between inclusive and exclusive interpretation for or is not a new research focus, not much empirical work has been done. Research work such as [Noveck et al., 2002; Goro et al., 2004] have investigated the condition under which interpretation for inclusive or and interpretation for exclusive or should be anticipated. Table 6 describes interpretation for Inclusive or and Exclusive or.

S_1	S_2	S_1 (inclusive) or S_2 $S_1 \vee S_2$	S_1 (exclusive) or S_2 $[S_1 \vee S_2] \wedge \neg [S_1 \wedge S_2]$
False	False	False	False
False	True	True	True
True	False	True	True
True	True	True	False

Table 6. The truth table for Inclusive or and Exclusive or

Inclusive or $[S_1 \vee S_2]$

Inclusive or describes the idea of when the constituents are true or one of the constituents is true, then the statement is true. The statement is false only when the constituents are altogether false. In inclusive or, choosing one disjunct[3] does not preclude choosing the other disjunct. A clearer indication of inclusive or is to say one or the other or both[4] in the RStat when the requirement writer's actual intension is inclusive or.

Note that *P*, *Q*, are the variables to be replaced by the constant text. The writing convention adopted in the following section is:

- / presents of exclusive or, allowing only one chosen option.
- [*P*] represents the idea of possible occurrence of the referred context *P* inside the brackets.

[3] Disjunct refers to the word or phrase that is linked by disjunction or.
[4] This use of both is not excluded by Rule V.1, which suggests avoiding both when it is combined with a following and.

- (*P*/*Q*/ …) represents the idea of either *P*-occurrence or *Q*-occurrence or any others as identified, but only one occurrence each time.

The following section discusses the scope of several contexts that lead inclusive or to carry conjunctive interpretation.

i. Every Noun [who/which/that Noun-Phrase]

every followed by a noun, and possibly followed by who/which/that and Noun-Phrase, is very likely to have an inclusive or interpretation as observed from E22, E23 and E24.

E22: Every student who speaks French or Spanish likes to travel.

E22 carries interpretations of (1) every student who speaks French likes to travel and every student who speaks Spanish likes to travel (2) either every student who speaks French or every student who speaks Spanish likes to travel (3) every student which speaks both French and Spanish likes to travel.

E23: every environment which is online or offline

Meanwhile, E23 has several interpretations (1) every environment which is online and every environment which is offline (2) either every environment which is online or every environment which is offline (3) every environment which is both online and offline.

E24: every user who has full access or partial access to the database.

Likewise, the potential interpretations for E24 (1) every user who has full access to the database and every user who has partial access to the database (2) either every user who has full access to the database or every user who has partial access to the database (3) every user who has both full access and partial access to the database.

The rational interpretation for E22 is every student who speaks French likes to travel and every student who speaks Spanish likes to travel; as for E23 is every environment which is online and every environment which is offline. Correspondingly, the interpretation for E24 is every user who has full access to the database and every user who has partial access to the database.

ii. (Few/None of the) which precedes Noun-Phrases

The unclear scope of None of the NP leads to the inclusive or interpretation as demonstrated in E25.

E25: None of the users is unauthorised or unidentified.

In essence, E25 means to say none of the users is unauthorised and none of the users is unidentified.

iii. Only ... or ...

The use of Only generally favours an inclusive or interpretation as shown in E26 and E27.

E26: The system shall log only unauthorised or unidentified accesses.

In matter of security concern, the correct interpretation should be the system shall log every unauthorized access or every unidentified access or both of them.

E27: The system shall log only every unauthorised access or every unidentified access or altogether.

iv. Complement of before

In the context of preposition such as before preceding complement generally favours inclusive or interpretation as demonstrated in E28.

E28: The system shall reboot before updating anti-virus or anti-malware programs.

E29 demonstrates a rewriting interpretation from E28, eliminating the inherent potential ambiguity caused by disjunctive or.

E29: The system shall reboot before updating anti-virus program and the system shall reboot before updating anti-malware program.

v. Complement of without

Likewise, preposition such as without preceding complement derives inclusive or interpretation as illustrated in E30.

E30: The system shall reboot without updating anti-virus or anti-malware programs.

The rational interpretation of E30 is E31 that eliminates the inherent ambiguity of disjunctive or.

E31: The system shall reboot without updating anti-virus program and the system shall reboot before updating anti-malware program.

vi. Determiner any

The use of any in the context of a negation derives an inclusive or interpretation. For example,

E32: The system shall not print any event or reconstruction data.

Due to the correlation of the conjunctive interpretation derived from inclusive or, E33 is the ideal interpretation of E32.

E33: The system shall not print even data and the system shall not print reconstruction data.

vii. Context where the preceding sequence is Verb Negation Verb

When the preceding sequence has the pattern of Verb Negation Verb followed by direct object that consists of singular nouns separated by disjunction or as shown in E34, the interpretation resulting is inclusivity as expressed in E35.

E34: The system shall not receive LVL0 or LVL1 data.

E35: The system shall not receive LVL0 data and the system shall not receive LVL1 data.

Exclusive or $[[S_1 \vee S_2] \wedge \neg [S_1 \wedge S_2]]$

Unlike inclusive or, exclusive or presents the idea of when both constituents are true or when both constituents are false, then the statement is false. In other word, choosing one disjunct *excludes* choosing the other disjunct. Generally in spoken English, in order to convey the intention of exclusive or representation, one would give emphasis on his statement. In written English, one can sometimes achieve the effect by using bold type, or capitals, or underlining. To present the RStat in a clear, unambiguous and yet convey the meaning of exclusive or, perhaps one of the ways is to assert *A* or *B* but not both *A* and *B* or alternatively to write as either[2] *A* or *B* in the RStat.

E36 describes two possible cases in which choosing only one of them is a must. If the missing information on the doer is referring to the system, then the possible interpretation will be: the system shall trust the user and authorise him or in contrast the system shall not trust the user.

E36: The user shall either be trusted or not trusted.

Assuming the missing information is referring to the system, then E37 best describes E36.

E37: The system shall trust or shall not trust the user, but not both.

The following section discusses the scope of several contexts that do not allow disjunction or to have conjunctive interpretation.

i. In context of some of followed by a Noun or Noun-Phrase, and joined with disjunction or as demonstrated in E38. The use of some represents indeterminate quantity of the entity it modifies. Compounding to this problem is indeterminacy of inclusive or and exclusive or. It is difficult to determine whether (1) some event data or some reconstruction data or both data shall be printed together, or (2) either some event data or some reconstruction data, but definitely not both data altogether to be printed.

   E38: The system shall print some event or reconstruction data.

ii. Complement of after

   In the context of preposition such as after preceding complement does not necessarily carry inclusive or interpretation. Eg39 doesn't describe clearly whether the reboot happens right after updating both anti-virus and anti-malware altogether, or updating only one of them but not both is enough to trigger the rebooting process. Logically, updating at least one of them is enough to trigger the system's reboot, which is inclusive or.

   E39: The system shall reboot after updating anti-virus or anti-malware programs.

iii. Complement of with

   Likewise, preposition such as with preceding complement does not necessarily mean inclusive or as illustrated in E40.

E40: The system shall reboot with updating anti-virus or anti-malware programs.

E40 is potentially ambiguous because it is not possible to tell whether the reboot happens with updating both anti-virus and anti-malware program, or updating only one of them is necessary to trigger the reboot process.

The discussion above describes contexts that lead to an inclusive or interpretation and contexts that lead to an exclusive or interpretation. However, whenever a RA encounters an or in a RS, the RA will have to ask the client whether his or her intent is an exclusive or or an inclusive or.

3.2.3 Only

Each placement of only in a sentence has a distinct interpretation. The interpretation of a sentence with only depends on what the only modifies, i.e., what follows it.

The following discusses the variety of contexts in which an only can be positioned and the interpretation of each positioning.

- Only preceding the verb it limits.

  E41: John only rinsed the cups.

  E41 tells that John only rinsed the cups; he did not wash them and dry them.

  E42: John rinsed only the cups, and he did not rinse anything else.

  Despite the rewriting of E41 to E42, there isn't sufficient information to tell whether or not John knows how to wash up the cups properly. Correspondingly, the interpretation of E43 is opposed to of E44.

  E43: The system shall only log unauthorised access.

E44: The system shall log only unauthorised access.

E43 describes that the only task the system does is to log it with unauthorised access whereas E44 describes the only access the system logs is an unauthorised one.

- Only modifies subject

  E45: Only John saw David leave.

  When only is placed modifying the sentence's subject, it means that the subject is the sole or solitary doer acting or is acted. The rational interpretation of E45 is the other people except John were too busy to notice when David left.

  Likewise, E46 describes that other user other than the engineer shall not add cask load to the system.

  E46: Only the engineer shall add cask load to the system.

- Only as adverb

  Only is primarily an adverb. With numerical expressions it has different meanings and the meaning depends on the implied direction of growth. If the direction is upward, then only means "no more than", and if the direction is downward, then only means "no less than". In temporal expression, only has a meaning of "not until" and "no longer ago than". In addition, another possible meaning of only is "cannot but" [Carter and McCarthy, 2006].

  E47: Travellers to this remote village can only wonder at its complete peace and serenity.

  E47 means travellers cannot do anything other than wonder at the village complete peace and serenity. E48 give further illustration E47 with additional information.

E48: Travellers to this remote village can only wonder at its complete peace and serenity, they cannot also live with it.

- Adjective only

Only can also be an adjective with a main meaning of "single".

E49: She is the only one they are prepared to confide in.

E50: Database administrator is the only one to do database maintaince.

- Only (if/when)

When only is used before if, either immediately before or in a previous clause, it has a temporal interpretation and has the meaning of "provided that" or "not… unless".

E51: Visitors will be admitted to the ICU only if the proper attire is worn.

Initially when a visitor pay visit to the hospital, he is in his own clothing. When he wishes to visit his friend in ICU, he is not allowed to enter without ICU standard attire. E51 means that unless visitors have worn proper attire, visitors will not be admitted to the ICU. The admittance to ICU is granted when the visitor has worn the proper attire.

E52: The system shall allow access to database only when the user is authorised.

E52 best depicts the condition of user who initially hasn't had authorisation to access database. In addition, E52 has a temporal interpretation in which the system does not allow access to unauthorised user and the system continues to do so. Until the particular user is authorised, the system shall allow his access to database.

- If only

In a conditional statement when only pairs with if and if precedes only, if only expresses a strong wish.

E53: If only I could get enough money together, I could go and work in Australia for a year.

- Only + to-infinitive

Only is used before a to-infinitive in order to introduce an action or event which is sudden and a little unexpected. This use is usually formal and literary.

E54: The music shop ordered 50 copies of the CD, only to discover that the songs were sung in a foreign language.

E55: He stopped the car, only to start it again violently.

- Only just

Only commonly modifies just in references to signify a very recent event where just can have a meaning of "with very little to spare" or 'barely'.

E56: We've only just got here. When did you arrive?

E57: There was so much traffic I only just barely made the flight.

3.2.4 Also

Also is twice as frequent in writing as it is in speech. It may occupy a variety of positions, and each position has different meaning.

- Modifying Verb Phrase

E58: Jack also doesn't agree with Darwin's theory.

E58 means in addition to other things that Jack doesn't do concerning on Darwin's theory, Jack doesn't agree to it. Another example,

E59: Anne is very approachable and treats everyone equally. She also handles staff with domestic problems very well.

- Modifying Noun Phrase Object

  E60: Jack doesn't agree also with Darwin's theory.

  In addition to Jack's disagreement with Lyell's uniformitarian ideas, Jack does not agree also with Darwin's theory.

There are many other words such as even, almost, etc. that are as ambiguous as only, where the position's of the word influence the interpretation of the statement's context. More discussion on these words is covered in [Bach, 1998].

3.2.5 Unless

Unless expresses the idea of if not or except if [Swan, 2005] and is used occasionally instead of if not in a conditional RStat. Unless is sensitive also to temporal ordering in which the referenced event has not happened or did not happen or will not happen until a certain time period arrives or has lapsed, or explicit affecting condition occurs.

E61: Unless you can reduce the weight of that case, I'm afraid you won't be allowed on the flight.

A unless *B* in which *A* and *B* share the same time of reference, is frequently interpreted as truth-functionally equivalent to if not (*B*), then *A*, where not (*B*) means the logical negation of *B* that usually contains somewhere the word not. However, *A* unless *B* is sometimes also interpreted to not (*B*) if and only if *A*. This is when such a person is worried about what happens during the unspecified situation of *A* unless *B* when B is true. Rewriting *A* unless *B* as if not (*B*), then *A* shows that *A* unless *B* is not the same as not (*B*) if and only if *A*.

E62: Setting controls (sort algorithm, speed, initial order, data value, sort speed) will not be accessible during the animation process, unless the animation has been stopped.

E63: Setting controls (sort algorithm, speed, initial order, data value, sort speed) will not be accessible during the animation process, if the animation has not been stopped.

E64: Unless requested by the Client, the animation will not display the display of the temporary array being worked on.

E65: If the client does not request the display of the temporary array being worked on, the animation will not display the temporary array being worked on.

A unless *B* says nothing about what happens if *B* is true. There is a possibility of rewriting *A* unless *B* to if (not) *B*, then *A* is incorrect when the rewriting has different interpretation from the intended meaning, if *B* is true. Therefore, if it is desired to specify that *C* happens if *B* is true, an explicit RStat, if *B* then *C* must be given. In accordance to that, the corresponding additional specifications are Eg43 and Eg44.

E66: Setting controls (sort algorithm, speed, initial order, data value, sort speed), will be accessible if the program has stopped the animation.

E67: If the client requests the display of the temporary array being worked on, the animation will display the display of the temporary array being worked on.

E66 and E67 are the potentially ambiguous interpretations caused by the use of unless. It's difficult to tell whether E66 and E67 are the intended requirements implied from the use of unless or they are merely ambiguous (resulting from more than one interpretation). Compounding this ambiguity is the fact that for some *A* unless *B* has temporal interpretation, in which *A* is true initially and it continues to be true until such time as *B* happens to be true [Chandler, 1982].

The use of unless in E62 and E64 show there exist temporal bond in which:

- In E62, the condition to access the setting control is when the event of stopping the animation is true. Until such event has not become true, then the access to setting control remains prohibited.
- In E64, the event to display of the temporary array being worked on will never happen in the case the user never asks for it. Therefore, until when the user asks the display of the temporary array being worked on, then the event to display is set to happen or true.

In order to understand the exact context of unless in E62 and E64, the human analyst has to explain the intended meaning of the ambiguous E62 and E64 so that they can be rewritten correctly.

3.2.6 Conditional If

A conditional statement deals with an imagined situation which is either (1) a possible situation, (2) an unlikely situation, or (3) an impossible situation. The speaker or writer uses a conditional statement to describe something that can happen, that cannot happen, or that has happened, and then associates that situation with possible consequences or outcomes, or offers a logical conclusion derived from the situation. Conditionality is conveyed chiefly by means of conditional clauses. A conditional clause is most often introduced by the connective if.

E68: If it rains, we are going to stay in the house.

Differences in tense and modality are important to a possible or imagined situation. In the conditional clause, tense choices express different types of potential event where in the main clause, the use of modal verb is to indicate the unfulfilled outcome of those events.

- **If… then …**

It is common to interpret If… then… statement as a truth-functional conditional statement. If… then… conditional statement divides into two constituents, which do not play equivalent roles where If introduces the antecedent and then introduces the consequent. Therefore, $A \to C$ signifies the first constituent A known as antecedent and the second constituent C known as consequent. However, the antecedent does not play equivalent roles with consequent as $A \to C$ is not generally equivalent to $C \to A$. Consider the following two conditionals.

E69: If the server is overloaded, then the server stops. $O \to S$

E69 has temporal interpretation as it involves a time element where the server's Overloaded condition must be true before the server turns to Stop condition.

E70: If the server stops, then the server is overloaded. $S \to O$

In contrast, though E70 appears similar to E69, E70 does not carry temporal interpretation because it indicates property of a system as a result of its requirements being implemented. Hence, E69 is probably one of the requirements that depict possible conditions cause the server to stop. There is a possibility that the server stops due to power failure or other valid reason.

Nevertheless, If… then… is not always truth-functional. Consider the following conditional statements.

E71: If I lived in L.A., then I would live in California.

E72: If I lived in N.Y.C, then I would live in California.

Presently, E71 is true since L.A. is inside California and E72 is false since N.Y.C does not overlap California. Furthermore, when the two constituents in Eg71 are false (the author did not live in L.A., hence he did not live in California), they yield a true conditional. On the other hand, when the two constituents in Eg72 are false (the author did not live in N.Y.C., hence he did not live in California), they yield a false conditional. This shows the conditional connective employed in E71 and E72 are not truth-functional.

A conclusion of the discussion on the potential ambiguity resulting from the use of quantifier, coordinator, and others, is to concentrate on explicitness and precision while authoring a RS. Explicitness is highly desirable so that there is no assumption of background, situational knowledge other than that in the application domain, and the conceptual content of each reference or term will be unambiguous and can be expected to be interpreted in only the one way that the writer desires. When all readers derive the same interpretation, ambiguity is not a problem anymore.

4 State of Practice in NLRS

"Requirements Specification necessitates the use of some specification language" – Pankaj Jalote

Representing requirements in NL is ideal for human communication and definition, but the correctness of any written NLRS is not guaranteed due to the inherent ambiguity of NL. On the other hand, representing requirements in a FL might guarantee the correctness of the written NLRS provided that the RS is not ambiguous before being rewritten in the FL. Unfortunately, a FL is not suitable as a human language due to the FL's complexity.

Despite NL inherent ambiguity, impreciseness, and incompleteness, NL is still preferred by many as a communication facilitator. Hence for decades, research work has been going on that aims to identify and classify techniques and approaches to reduce the inherent ambiguities in NLRSs as summarised in a survey report [Denger et al., 2001]. In general, these approaches can be classified into three categories [Tjong et al., 2006]:

4.1 Approaches that define linguistic rules and analytical keywords [Fabbrini et al., 2000; Fabbrini et al., 2002; Wilson et al., 1996]

Wilson, Rosenberg and Hyatt [Wilson et al., 1996] define the overall quality aspects of RSs and requirements in general. The two quality aspects are:

- quality attributes that define aspects such as completeness, correctness, traceability, uniguity, etc. and

- indicators for RSs and individual RStats that indicate a lack of quality such as imperatives, continuances, directives, options, and weak phrases.

Later, the Software Assurance Technology Centre (SATC) implemented these quality attributes and indicators inside its Automated Requirements Measurement (ARM) tool. A detailed classification of the ARM indicator is in Table 7.

	Imperative	**Continuance**	**Directive**	**Option**	**Weak Phrases**	**Incompletes**
INDICATORS	shall	below:	e.g.	can	adequate	TBD
	must	as follows:	i.e.	may	as appropriate	TBS
	is required to	following:	for example	optionally	be able to	TBE
	are applicable	listed:	figure		be capable of	TBC
	are to	in particular:	table		capability of/to	not defined
	respon-sible for	support:	note:		easy to	not determined
	will	and			effective	but not limited to

	should	:			as required	as a minimum
					normal	
					provide for	
					timely	

Table 7. Standard ARM Indicators [Wilson et al., 1996]

- *Imperatives* are words and phrases that command something must be provided.
- *Continuances* are phrases and words used to introduce the specification of requirements at a lower level.
- *Directives* are a category of words and phrases that point to illustrative information within the requirements document.
- *Options* are a category of words that give the developer latitude in satisfying the specification statements containing them.
- *Weak Phrases* are a category of phrases that are apt to cause uncertainty and leave room for multiple interpretations.

Fabbrini et al. [Fabbrini et al., 2000] distinguish the aspects between requirements sentence quality (RSQ) and requirements document quality (RDQ). Following is the list of indicators for RSQ and RDQ:

- RSQ related indicators include implicit subject sentences, multiple sentence, optional sentences, subjective sentences, underspecified sentences, vague sentences, and weak sentences.

- RDQ related indicators include comment frequency, readability index, under-referenced sentences, and unexplained sentences

From their findings, they develop an automatic tool called QuARS (Quality Analyser of Requirements Specifications) that will support the analysis and quality evaluation of RSs. QuARS incorporates the Quality Indicators for RSQ and RDQ. Table 8. and table 9. show detailed elaboration on each attribute of the Quality Indicators.

	Implicit Sentence	**Multiple Sentence**	**Optional Sentence**	**Weak Sentence**
I N	Demonstrative Adjective: this, these, that, those	>1 subject	possibly	can
D	Pronouns: it, they	> 1 main verb	eventually	could
I C	Preposition: above, below,…	>1 direct complement	in case of	may
A T	Adjective: previous, next, last, first, following, …	>1 indirect complement	if possible	
O			if appropriate	
R S			if needed …	

Table 8. QuARS Indicators [Fabbrini et. al, 2000]

	Subjective Sentence	**Vague Sentence**	**Underreferenced Sentence**
I	Having in mind	Easy	According to
D	Take (into) account	Strong	On (the) basis of
I	Take into consideration	Good	Relatively to
C	Similar	Bad	Compliant with
A	Similarly	Useful	Conformant to
T	Better	Significant	…
O	Worse	Adequate	
R S	As [adjective] as possible	recent	
	…		

Table 9. More QUARS Indicators [Fabbrini et. al, 2000]

In essence, the ARM approach and QuARS approach present quality attributes, a model, and indicators used in evaluating the quality of the existing NLRSs. Frequently used keywords, phrases and sentence structures that cause imprecision are grouped and counted by computer programs. These approaches are thought to be effective in detecting defects and ambiguous NLRStats found in the RSs.

4.2 Approaches that define guideline-rules [Götz and Rupp, 1999; Juristo et al., 2000]

Götz and Rupp developed a rule base that contains all rules needed to detect defects, ambiguities and weak phrases in RSs. They distinguish three main transformation process used to model the original intention of a person in communicating the requirements which namely are:

- deletion that reduces the perception of a person to a scope he or she can deal with,
- generalisation that leads to a detachment of an experience from its context and to assume that the experience is overall valid, and
- distortion that is related to nominalisation i.e. a noun stands for a complex process. Examples of nominalisation include: the recording, the playback, the take off, etc.

These transformation processes form the base to detect defects in RSs. For each process, there are rules developed to solve the problems of the related defects. The rules are believed to be efficient means for validating and specifying NLRSs.

Juristo et al. [Juristo et al., 2000] classified requirements into static and dynamic requirements. They show how to recast any static requirements into the structure of the Static Utility Language (SUL) and how to recast any dynamic requirements into the structure of the Dynamic Utility Language (DUL). Each of SUL and DUL is specified by a formal grammar and is composed of several natural language structures each of which can be translated into predicate logic. Therefore every utterance in either language is not ambiguous. Juristo et al. also define distinct guidelines for static and dynamic requirements to be adapted in reformatting the RSs.

In short, these approaches summarise rules and guidelines to be adapted in preparing NLRSs. On one hand, the guidelines avoid incorrect constructions of NLRSs by detecting the potential defects and ambiguities in NLRSs. Thus, guidelines help avoid the introduction of NL ambiguities by restricting the level of freedom in preparing or writing NLRSs. On the other hand, the definition of rules functions as a checklist by a requirement engineer to decide the correctness of the written NLRSs.

4.3 Approaches that define specific language patterns to be used in writing the NLRSs for different respective domains [Barr, 1999; Denger, 2002; Ohnishi, 1994; Rolland and Proix, 1992].

A language of language pattern is a devised description of language in a more restricted way. There are several types of patterns such as architectural patterns that show the high level architectures of a software system, design patterns that focus on the programming aspects, and even patterns for project management [Martinez et al., 2004].

Ohnishi [Ohnishi, 1994] developed an X-JRDL analyser for a Japanese language, which is based on a concept called requirements frame model for the file system domain. The requirements frame model distinguishes between three different frames, namely Noun Frame, Case Frame, and Function Frame. Each of these frames restricts the vocabulary and the context of the RStats. Ohnishi also states that with the requirements frame model, each RStat can be transformed into an internal representation called CRD (Conceptual Requirements Description). Then, X-JRDL can automatically analyse each requirement description.

The novel idea of Rolland and Proix's approach [Rolland and Proix, 1992] is to model a conceptual schema of the future system by using linguistic mechanism, which is used to abstract real world phenomena. Tailored to the needs of the database development domain, Rolland and Proix define patterns and cases in which the cases originate from Fillmore's case system [Fillmore, 1968]. They classify six types of cases which namely are Agentive, Instrumental, Dative, Factitive, Locative, and Objective. Besides, Rolland and Proix also categorise several classes of verbs and distinguish two main linguistic patterns. The linguistic patterns are a set of patterns that combine cases and classes of verbs. The patterns namely are:

- elementary patterns that allow associating cases to syntactic units of a clause
- sentence patterns that allow associating cases to clauses of a sentence

Rolland and Proix then develop a tool called OICSI that adapt their approach. The tool is based on the French natural language and aims to automate the support of the requirements engineer.

Each of Barr and Denger [Barr, 1999; Denger, 2002] focused on language patterns for embedded system domain. Barr identifies specification patterns or sentence patterns, which shall support the transformation of unstructured natural language requirements into a formal specification language. In his work, he distinguishes two different classes of patterns, which are:

- If-then patterns described within the Rule-Scheme

  The sentence structure of a rule scheme is described by the pattern *if condition b, then consequence k*. Hence, each rule has a condition and a consequence, in which within real-time system, the condition and the consequence have a temporal relationship.
- Patterns expressing an overall valid fact described within schemes for consequences without conditions where the consequence part of a rule is realised if and only if the condition evaluates to true, which is also an overall valid fact.

In addition, as part of Rule-Scheme, Barr identifies Condition Pattern, Order Pattern, Delay Pattern, Consequence Pattern, Stability Pattern, Context Pattern, Priority Pattern, and Exclusion Pattern that are valid in Real-Time Systems.

Denger [Denger, 2002; Denger, 2003] developed an approach for reducing the problem of imprecision in NLRSs with the use of natural language patterns, authoring rules, and document templates. He outlines distinct language patterns such as Functional Requirement Sentence Patterns, Event Patterns, Reaction Patterns, Computation Pattern, Relationship Patterns, Exception Patterns, Patterns for special aspects, and Nonfunctional Requirement Sentence Pattern. Denger even devises a metamodel for functional RStats adaptable in the embedded system domain.

Besides, there is also work in specifying a controlled language for writing the requirements in an almost NL. Fuch and Schwitter [Fuchs and Schwitter, 1996] define a Controlled English, which is a subset of NL with restricted syntax and semantics of full NL and a domain-specific vocabulary. Controlled English allows domain specialists to interactively formulate RStats in domain concept. Fuch and Schwitter have developed a system called Attempto that translates complete RS in Controlled English into discourse representation structures which are structured forms of first-order predicate logic and optionally into Prolog.

4.4 Other Approaches

Other researchers provide guidelines for writing good RSs. Hooks [Hooks, 1994] described common problems found in RSs and suggests guidelines that help avoid the problems. Moreover, she also conducts an in-depth survey on the principal sources of defects in NLRSs and the associated risks. Similarly, Firesmith [Firesmith, 2003] described both characteristics of a good RS and potential problems that occur in writing a RS.

The work a Ambriola and Gervasi [Ambriola and Gervasi, 2003] concentrates on achieving high-quality of NLRSs through *CIRCE (Cooperative Interactive Requirement-Centric Environment)*. *CIRCE* is based on the concept of successive transformations that are applied to the requirements, in order to obtain concrete and rendered views of models extracted from the requirements.

This thesis work identifies disambiguation guiding rules that are sufficient to reduce the informality, imprecision and ambiguity of the NLRSs. The guiding rules are constructed based on the studies of ambiguous RStats found in RSs [Bray, 2002; BPS, 2005; CLS, 1999; DCS, 2002; EVLA, 2003; LAT, 2002; PESA, 2001; Stevenson et al., 2005]. The idea of having guiding rules is to reduce potential ambiguities in writing of any RStat. The guiding rules can be used also to help find potential ambiguities in an existing RS. The present set of guiding rules are expanded from Denger's guiding rules [Denger, 2002]. More discussion on guiding rules can be found in Chapter 6.

5 Research Method

"Although the basic logic of scientific methodlogy is the same in all fields, its specific techniques and approaches will vary, depending upon the subject matter" – Festinger and Katz

This chapter discusses the research process to come up with the guiding rules and SREE. During the research process, the author had to determine which approach brings the best results to the computational application, where and how to locate ambiguity, how many interpretations to assume, and how to represent the correct interpretation. Section 5.1 discusses the choice of the approach adopted in this thesis and the reasons for the choice made. Section 5.2 discusses the set of RSs collected. Section 5.3 explains how ambiguity, imprecision, indeterminacy, and vagueness differ in NLRSs. Section 5.4 outlines the construction of the guiding rules. Finally, Section 5.5 summarises the development of SREE and Section 5.6 discusses the previous attempts to automate the transformation of potentially ambiguous RStats into less ambiguous RStats.

5.1 Existing Approaches To Detect Ambiguity in NLRSs

Research on searching for ambiguities in NLRSs and then disambiguating them is not something new. As discussed in Chapter 4., there are approaches that vary from detecting ambiguity in previously written RSs and correcting the found ambiguous RSs by means of a verification process by RAs, to avoiding the introduction of ambiguity while authoring NLRSs. Approaches to detecting and resolving ambiguity in writing NLRSs can be classified into three categories [Tjong et al., 2006c]:

- Approaches that define linguistic rules and analytical keywords:

  Identification of ambiguous words or phrases to serve as indicators that help to find potentially ambiguous RStats. The indicators signify also the linguistic aspect of an RStat in order to be considered and uniguous.

- Approaches that define guideline rules:

  Guideline rules help to avoid introducing ambiguities into a NLRS. Hence, guideline rules restrict the author's freedom in preparing NLRS in ways that help avoid an incorrect construction of the NLRS. The rules can be used also as a checklist in inspecting the correctness of an NLRS.

- Approaches that define language patterns:

  A language pattern prescribes the use of a language in a more restricted way. Some approaches that define language patterns inherit from Fillmore's Case Grammar [Fillmore, 1968]. Literally, the pattern defined for a domain distinguishes the vocabulary to be used, the relationships among any group of words, and the context of the RSs in that domain.

From the studies, the author considered the merits of each approach, particularly any approach that can be implemented by software. She decided to define guiding rules that help to avoid introducing ambiguity at the first place while authoring RSs. Meanwhile, this approach can be used also to support inspections of RSs, whether the inspection process is automated or not. Automatic ambiguity detection by a tool helps a RA to search for ambiguity as identified in the guiding rules.

5.2 Corpus of NLRSs

The author found two case studies and several RSs from different domains to make a corpus of NLRSs that she used in this thesis. The case studies and RSs are:

- RS for The Lift Controller, A Case Study [Bray, 2002]:

  The Lift Control System is responsible in managing and monitoring the lift activities such as sending lift, calling lift, directing lift, lift waiting time, lift operating time, and lift safety requirements to ensure no accident ever occurred.

- RS for The Yacht Race Results (YRR) Program, A Case Study [Bray, 2002]:

  The YRR system is responsible for documenting each boat's details and each race's record, and producing reports.

- RS for Batch Poster System Detailed Business Requirements [BPS, 2005]:

  The BPS is responsible for inventory data entry, user access control, data maintenance, report handling, automatic database cleaning, and file interfacing.

- Cask Loader Software General Requirements Document [CLS, 1999]:

  The Cask Loader Software (CLS) aims to assist nuclear utilities in the task of loading spent fuel into casks for permanent storage. CLS will enable also optimising of loads for radiation and heat loading within regulatory limits.

- Data Cycle System [DCS, 2002]:

  The Data Cycle System (DCS) is a RS for a Stratospheric Observatory for Infrared Astronomy (SOFIA) project to provide a uniform, extensible, and supportable framework for all aspects of SOFIA's current and future scientific instruments.

- EVLA Array Operations Software Requirements [EVLA, 2003]:

The EVLA array operations range from operating the EVLA in a manner that supports the scientific work of the EVLA, ensuring the safety of the equipment and people working on the equipment, and assisting technical groups in maintaining peak instrumental performance.

- Large Area Telescope (LAT) Science Analysis Software (SAS) Level III Specification [LAT, 2002]:

  The LAT SAS provides basic utilities for data manipulation and visualisation.

- PESA High-Level Trigger Selection Software Requirements [PESA, 2001]:

  The ATLAS High Level Trigger Selection Software prepares data to be run in offline and online environment for development, testing, integration, reconfiguration, verification, validation, and optimisation. There is a priority ranking of the importance of the individual requirements, but all requirements are expected to eventually be met by the software.

- Sort Algorithm Demonstration Program Software Requirements Specification [Stevenson et al., 2005] :

  The Sort Algorithm Display Program is intended to teach how various sorting algorithms operate and to save substantial startup and training time of new staff.

Each of the RSs originates from a different domain. Analysing RSs from different domains helps recognise different kinds of ambiguity.

5.3 Searching For Ambiguous NLRSs

Defining guiding rules is not entirely new research. However, not much work has been done, particularly to derive guiding rules directly from ambiguous RSs, based on an examination of several industrial strength RSs.

5.3.1 Definition of: Ambiguity, Imprecision, Vagueness, Indeterminacy

The following section describes this thesis's scope for each of ambiguity, imprecision, vagueness, and indeterminacy in order to determine whether or not each of them is distinctive.

5.3.1.1 Definition of Ambiguity

Ambiguity in a RStat occurs when the RStat is susceptible to more than one interpretation influenced by the relationship with the external world rather than on objective knowledge. Literally, an ambiguous RStat is not truth functional because of the confusion in choosing the correct interpretation out of its multiple interpretations. Chapter 3 has already covered ambiguity broadly.

5.3.1.2 Definition of Imprecision

The concept of imprecision is relevant the value of a data object. A value of a data object is called *precise* when the set of possible values is singleton; otherwise, it is called imprecise. In other words, imprecise data generalises the concepts of multiple acceptable scenarios which is also known as ambiguity. Imprecision is not measurable by probabilistic degree. There are three categories of imprecision [Virrantaus, 2003]:

- *ontological imprecision* where the exact truth cannot be achieved when questioned on the preciseness of the object of the reality, for example the existence of UFOs (Unidentified Foreign Objects).

- *epistemological imprecision* where the inability of human being to identify the real world object precisely because the measuring equipment is unable to measure precisely enough because of limited resolution. For example, at the present time, we cannot send a manned rocket to a far planet such as Pluto.
- *linguistic imprecision* where the verbal expression or written script is imprecise in a way that the expression carries several meanings. Linguistic imprecision is known also as ambiguity.

Thus, imprecision is analogous to ambiguity. Imprecision in a RS occurs when (1) the RS is ambiguous and is thus unable to specify one out of multiple interpretations or (2) the RS is indeterminate or uncertain due to definite criteria's not being set at the time the RS is written.

5.3.1.3 Definition of Indeterminacy or Uncertainty

The concept of uncertainty is relevant to the degree of truth of its attribute value. Uncertainty is tightly associated with probability, classical accuracy, and error definition. Data are said to be uncertain when the fullest amount of belief, known also as confidence, cannot be given to the data.

Indeterminacy refers to a state of affairs in linguistic study in which there is uncertainty on the part of a native speaker, or disagreement between native speakers, as to what is grammatical or acceptable, or as to how and where a boundary line between different types of structure might best be drawn [Crystal, 2003].

Indeterminacy in a RS occurs when the RS is undefined, unestablished, or not precisely outlined due to lack of clarity in specification of attributes of the RS, or in anticipation of future usage. The source of indeterminacy varies from insufficient or uncertain data to variability over time or space.

5.3.1.4 Definition of Vagueness

Vagueness is another form of uncertainty in which which boundaries lack precision. Vagueness arises when the extent of the applicability of a word is doubtful. Russell [1923] was the first to have discussed of them in the ancient paradox of bald person. There are men that are certainly bald, and there are men that are certainly not bald. However, there are men for whom it is difficult to say if they are bald or not. In vagueness, there is a borderline that is fuzzy enough that it is impossible to decide on which side of the bordeline an object belongs. The aspect of vagueness is descriptive for a given category. For example, consider the opposites short and tall. Clearly, anyone with height less than 1.4 m is short while anyone with height more than 1.9 m is not short, i.e, is tall. However, someone with height between 1.4 m and 1.9 m cannot be classified. On the other hand, it might be possible to express a partial or gradual membership to the category.

Vagueness that is associated with a qualitative concept to distinguish among objects that exhibit continuous variation in the observables relevant to the description of the concept is known as *threshold vagueness*. This phenomenon underlies the famous Sorites paradox. If we remove one grain of sand from a heap, is what remains still a heap? The vagueness that arises from the inability of the world to divide into exactly defined categories is known as *partiality vagueness*. For instance, a frog is described as green, even though its mouth and tongue are pink and its eyes are black [Bennett, 2005].

Vagueness in a RS results from trying to refer to an object as having an absolute value when the object has range of several values. Consider, for example,

E73: The system shall support a number of concurrent user sessions limited only by system resources.

E73 is vague in several ways. How many concurrent user sessions shall the system support? What kind of concurrent user sessions shall the system support? What system resources are limited? With all the vagueness in the sentence, it is difficult to understand the intended meaning of the sentence.

An indeterminate RStat has one interpretation under which the RStat is true and another interpretation under which the RStat is false. Indeed, the idea of indeterminacy can be applied to the problem of vagueness in several technically different ways such as in the definition of tall. Suppose that Ben is a borderline case for tall. In the indeterminacy view, the sentence Ben is tall is indeterminate in truth value. It is neither determinately true nor determinately false.

Another example,

E74: The Science Analysis Software performs prompt processing of Level 0 data to produce Level 1 event data.

The use of prompt is indeterminate and hence leads to vagueness. It is difficult to measure the acceptable processing speed that should be considered as prompt because prompt in one context maybe different in other context. One way of interpreting prompt is through the author's partial understanding of the meaning of a vague word prompt, and that her knowledge of the meaning of prompt remains incomplete with respect to borderline cases.

Vagueness is a strong indication also of an assumed shared knowledge and denotes in-group membership in which the referent of a vague pronoun is assumed to be known by the reader, as demonstrated in E75.

E75: Reconstructed objects derived from seeds with the same geometric position or which have in common some data from which they were reconstructed, should be treated as mutually exclusive when testing a physics signature.

The use of pronoun they in E75 is very vague and also ambiguous because it is hard to tell whether they refers to the some data, they refers to the seeds, or they refers to reconstructed objects.

This thesis collapses ambiguity, imprecision, uncertain, indeterminacy, and vagueness into one term "ambiguity" because the distinction among the terms does not affect the nature of guiding rules. They are all to be avoided, and each is recognisable by its linguistic patterns.

5.3.2 How To Decide If a RStat Is Indeed Ambiguous

An ambiguous RStat has more than one interpretation. When the thesis author began the ambiguity analysis, she decided that an RStat *S* is ambiguous when she found more than one interpretation for *S*. She distinguished syntactic ambiguity and semantic ambiguity. She used the ARM and QuARS indicators to find ambiguities in *S* she might have missed. At first she was surprised when she discovered that her first reading of *S* was different from her later reading of *S*. She realised that among the reasons for the change in reading were subconscious disambiguation in her first reading, lack of context understanding, and lack of inspecting ambiguous requirements knowledge. She realised that the ambiguous RStats she observed in actual RSs occur in common practice in written RSs. So, whenever she found a new kind of ambiguity, she looked for indicators of that kind of ambiguity, and she documented both the kind of ambiguity and its indicators. Occasionally, one of her advisors, Berry or Hartley, found an ambiguity that she had not.

5.4 Constructing Guiding Rules

The aim of the guiding rules is to restrict the flexibility of NL to avoid ambiguity. Because the problems of ambiguity, indeterminacy, and vagueness affects all kinds of RStats, the distinction between functional and nonfunctional Rstats does not affect the nature of guiding rules.

The two objectives of guiding rules are:

- as an aid for writing requirements:

  A person who applies the guiding rules in writing RStats should produce less ambiguous RStats.

- as an aid for inspecting written requirements:

  A person who applies the guiding rules in inspecting written RStats should find more ambiguities.

From identified ambiguities in the corpus of RSs and elsewhere, a collection of guiding rules were defined to avoid these ambiguities in writing RSs. Whenever a new kind of ambiguity was detected, a new guiding rule to avoid that ambiguity was added. The inconsistency in rules is always possible. We are aware and know that any given sentence might be considered fine by one rule, may not be considered fine by another rule. We do not see this inconsistency as a critical issue because the purpose of identifying ambiguities is to find sentences about which we need to ask the client questions on the ambiguities. Once the questions are resolved, then the ambiguous sentence are rewritten

The adaptability and practicality of the guiding rules was validated by using them to rewrite the ambiguous RStats of the corpus RSs into less ambiguous RStats.

5.5 Automated Ambiguity Detection: SREE

Finally, the author developed SREE, an experimental tool that assists a RA in identifying ambiguity in NLRSs. SREE automatically detects instances of many kinds of potential ambiguity in an input RS, based on the guiding rules. For each instance of potential ambiguity detected by SREE, the user has to decide whether the instance is a true ambiguity. SREE is able to scan, search, browse, and tag many RSs much faster than a human user. Furthermore, SREE can work tirelessly while a human cannot. Although SREE can make mistakes that a typical human doesn’t, such as false positives, a human analyst can overlook many ambiguities, since he or she subconsciously disambiguates or gets tired.

5.6 No Automated Transformation

SREE tries only to recognise ambiguities. SREE makes no attempt to transform any potentially ambiguous RStat into a less ambiguous form. Disambiguating an ambiguous RStat requires human understanding that SREE cannot have. Moreoever, even ascertaining that a potential ambiguity is a true ambiguity requires human understanding. Therefore, SREE's total purpose is to detect instances of potential ambiguity in its input.

Because, humans are highly creative in finding new ways to be ambiguous in new software applications, there is no limit on the number of guiding rules that are needed. However, the author hopes that at some point, she will cease to find new kinds of ambiguities and the rate of adding new rules will drop off.

6 Disambiguation Guiding Rules

"There are two guiding rules in the art of putting things from one language into another. The one is that the writer of the other language is to be made to come over to us, so that he may seem to be one of us; the other is that we are to go over to him and make the necessary adjustments to his conditions, his way of writing, his special turns of thought." C. K. Ogden, W. Terence Gordon

This chapter describes the strategy in constructing the guiding rules with a discussion about the nature of each rule. The unit of application of each of most rules is a single RStat *S*. Each rule that says to avoid a construction offers an alternative construction for saying the same thing less ambiguously; the alternative construction is signalled by "Instead,".

6.1 Important Consideration on the use of Guiding Rules

Defining guiding rules from ambiguous RStats allows us to closely examine each RStat that appears to match the pattern of a guiding rule. If the RStat is truly ambiguous, the alternative construction of the matched rule is applied to rewrite the RStat.

Note that the guiding rules cannot be blindly followed, because there are always exceptions for which the rules cannot and should not be followed.

6.2 Guiding Rules for using Natural Language

The guiding rules form a rule base for the use of NL to specify RStats to avoid introducing ambiguity while authoring RStats.

I. Structuring Domain-Related Context

Rule I1: Avoid writing undefined domain specific vocabulary and abbreviation in RS.

Write Instead: List of vocabulary for the domain specific defined in RS.

Discussion:

List of vocabulary can contain acronym or abbreviation along with a corresponding explanation for each acronym. Acronym eases the initialisation of a long naming for an object term and an acronym list helps a RA to minimise the long writing for that object term. For example, OS, which can mean Operating System and can mean also Open Source. Both Operating System and Open Source are commonly used terms in software development. In this case, glossary works as subpart of RS that provides a space for the requirement engineer to write down explanation on each of the acronym and term used in the respective domain.

II. Structuring Sentences

Rule II.1: Avoid writing long and complicated *S* that has adjectives, adverbials, and complex phrases to camouflage the intended process.

Write Instead: *S* as a simple yet short affirmative declarative sentence that precisely describe the actor, the one main process to trigger, and the object corresponding to the process. *S* shall not contain complex phrases, adjectives and adverbials.

Discussion:

Long sentence complicates the identification of main process, main subject, and the intended object. On the other hand, adjective and adverbial contribute also to camouflage the complication as it is difficult to identify the exact word the adjective or adverb intends to modify. Simple and short declarative sentence with one main verb improves readability and understandability of the main process, main subject, and the intended object in a sentence.

E77 is an example of a long and complicated RStat with coordination ambiguity. On the other hand, E76 shows an example of simple and short, yet understandable RStat as suggested by Rule II.1 and E76 clearly identifies the intended functionality of the system that is to store the processed data.

E76: The system shall store 20 GB of processed data per day.

E77: The software will follow the applicable regulatory and utility technical requirements in its speculated calculations and selection process.

In E77, first, it is difficult to tell whether E77 means to say

- regulatory technical requirements and utility technical requirements

or alternatively

- regulatory requirements and utility technical requirements

Second, it is also difficult to tell whether applicable modifies only regulatory or both regulatory and utility technical requirements. Finally, the use of will cause the RStat appear as an expression of a preference that is not a requirement.

Rule II.2: Avoid writing *S* in passive voice, especially in which no doer of the action is specified.

Write Instead: *S* in active voice with the doer of the action as the subject of *S*.

Discussion:

Writing *S* in passive voice contributes vagueness in that *S* because it is difficult to identify who initiates the process or what kind of condition that triggers the process.

Thus, identifying the doer of the action or the initiator of the process, the intended process or action to be triggered, and the object in *S*, helps clarifying the readability in *S*. E78 and E79 show examples of RStat written in passive form.

E78: The report's item list shall be printed.

E78 doesn't say who initiates the printing of the report's item list and how to initiate the printing job either automatically or manually by certain instruction. If the system application is the actor or doer of the printing job, then it is better to state this actor clearly. Hence, E78.1 is the suggested rewriting of E78.

E78.1: The system shall print the report's item list.

Likewise, E79 shows violation of rules II.2. There is no identifiable person who approves a cask load, and no identified doer who deletes the fuel bundles loaded.

E79: After a cask load has been approved, the fuel bundles loaded are deleted from the database.

However without the access of the requirements writer, a thorough analysis on the requirements' context shows that the engineer is responsible to check, approve, or disapprove the status of each cask load. The engineer is then to update or delete the fuel bundles loaded from the database. With this information, the suggested change of E79 is E79.1.

E79.1: After the engineer has approved a cask load, the engineer deletes the fuel bundles loaded from the database.

Rule II.3: Avoid writing *S* of the form There is *X* in *Y*. or *X* exists in *Y*.

Write Instead: *S* of the form *Y* has *X*.

Discussion:

Writing *S* of the form There is *X* in *Y*. or *X* exists in *Y* may contribute to implicit ambiguity other than the real intend of *S* as in E80.

E80: There is event data in ROS to hold the data received from the ROD of the detector.

It is difficult to identify whether ROS or event data is responsible to hold the data received from ROD of the detector. If E80 means to say ROS does hold the data received from ROD of the detector, and as according to Rule II.3, E80 should be rewritten to E80.1.

E80.1: ROS has event data and ROS shall hold the data received from the ROD of the detector.

If E80 means to say the event data is to hold the data received from the ROD of the detector, and as according to Rule II.3, E80 should be rewritten to E80.2.

E80.2: ROS has event data to hold the data received from the ROD of the detector.

Thus, writing *S* of the form *Y* has *X* simplifies and eases the interpretation of *S* that says *Y* the subject, contains component *X* the object.

Rule II.4: Avoid separating RStats that have relationship or dependency into each uniquely identifiable RStat.

Write Instead: Group together RStats that have dependency relationship.

Discussion:

This rule contradicts the idea that each RStat shall be uniquely identifiably. Separating dependent RStats into each unique RStat complicates the readability of existing relationship or dependency between RStats, and causes vagueness if the reference or modifier stated in that RStats is not repeated. The existing temporal order bond among RStats might be broken also. Thus, this rule ensures RStats that have some relationship or dependency, temporal or not, with each other are grouped together.

E81 shows three RStats that have dependency relationship and are grouped together into one RStat.

E81: The SDP shall provide the Level 1 data to the P1 sites. The Level 1 data shall arrive at the sites no later than 24 hours after completion of processing in the SDP. Then, the SDP shall provide the Level 0 data to the P1 sites.

Blindly rewrite E81 into three separately identifiable Rstats:

E82: The SDP shall provide the Level 1 data to the P1 sites.

E83: The Level 1 data shall arrive at the sites no later than 24 hours after completion of processing in the SDP.

E84: Then, the SDP shall provide the Level 0 data to the P1 sites.

without inserting P1 before sites in E83 will cause the sites to become vague. Moreover, the temporal order bond between E82, E83, and E84 has been broken. It is not possible to know that the processes described by E82, E83, and E84 must be done in the order written. On the other hand, if separation of individual RStats is required, then each separated RStat must have additional text to describe its context. For example, in E83, the sites must be changed to the P1 sites and the fact that E83 temporally follows E82 and precedes E82 must be described.

E85 and E86 further demonstrate RStats that together fall under the province of Rule II.4 and should be grouped together. Both E85 and E86, are written in passive form and hence both E85 and E86 violate rule II.2.

E85: The system shall be designed to accommodate the addition of a propulsion subsystem. The propulsion subsystem shall be capable of transferring the system from the circular parking orbit to the operational orbit.

E86: Between the time an elevator is called at a floor and the time it opens its doors at that floor, the elevator can arrive at that floor at most twice.

III. Structuring Nouns

Noun can stand as a complex process of definition. An exactly specified and unambiguous noun in a RStat aids interpretation in analysing that RStat. Following are the suggested rules to check insufficiently and incompletely specified noun in a RStat. The RA is to ensure each noun in a RStat has completely specified its reference to a defined person, a defined group of persons, or a defined real world object, or a defined group of objects.

Rule III.1: Avoid writing *S* that contains noun without binding a reference to that noun.

Write Instead: *S* that contains noun with precise referential to the specified noun.

Discussion:

Unreferenced noun in *S* introduces potential ambiguity to the noun, especially when there are more than one noun entities in *S*. Thus, specifying a definite referential for each occurrence of the noun while writing *S* is a must.

As illustrated in E87, just data itself is very vague because the term data is so commonly used to refer to an item of information. Furthermore, when there are more than one types of data specified in an RS, e.g. the lending statistic data, the observational data, etc., this gives rise to potential ambiguity. So, data in E87 is potentially ambiguous because data may refer to the lending statistic data or the observational data or some other type of data.

E87: Data shall be presented graphically to the user.

E87 violates Rule II.2 and Rule III.1. It is difficult to tell what data is to be presented (supposing there are different types of data), who presents the data, and to whom the data is presented.

The suggested rewriting of E87 is E87.1 in which E87.1 is in an active form where both doer of the present action, the exact data type, and the recipient object to whom the data is presented are identified.

E87.1: The system shall present the lending statistic to level A-users graphically.

Another example, E88 shows violation of Rule II.2, VI.10, VI.2, and III.1 by its use of passive form, all, should, and no noun referential.

E88: All data should be validated after entry and any errors should be advised.

It is difficult to tell what data is to be validated when there are different types of data), who validates the data, what kind of validation is to be done. Furthermore there is no reference to whom the errors are to be advised. Judging from the requirement's context, we take guesses that the doer of the data validation is the manager. Hence, the suggested rewriting of E88 is E88.1. Note that E88.1 is rewritten in an active form where the doer of data validation process action and the exact data type are identified.

E88.1: The manager shall validate each boat data after entry and the manager shall advise each error notified from validation process.

Rule III.2: Avoid writing *S* that contains anaphora or pronoun such as they or them to refer to collective sets of people or objects, and it to a singular object.

Write Instead: *S* that contains pronoun with correct antecedent to the pronoun.

Discussion:

Anaphoric expression such as pronoun is potentially ambiguous as pronoun may relate to more than one antecedent, or denote more than one referent. Unclear pronoun reference makes sentences confusing, vague, and difficult to determine the correct antecedent. When a RA comes across a pronoun, he or she is to analyse and replace pronoun with its antecedent.

E89 shows an example of pronoun it without its antecedent reference.

E89: The system shall have the ability to monitor the number of hosts connected to it.

The use of it in writing *S* is very ambiguous especially when it can refer more than one stated entity in the RStat. If it refers to the system, then the suggested change of E89 is E89.1.

E89.1: The system shall have the ability to monitor the number of hosts connected to the system.

E90: Reconstructed objects derived from seeds with the same geometric position or which have in common some data from which they were reconstructed, should be treated as mutually exclusive when testing a physics signature.

The use of pronoun they in E90 is very vague and also ambiguous because it is hard to tell whether they refers to the some data, or they refers to the seeds, or even they refers to reconstructed objects.

Rule III.3: Avoid writing *S* that contains data noun ranges with hidden assumptions.

Write Instead: *S* that contains a noun range in definite number type.

Discussion:

Vagueness occurs also in mathematical concepts expressed by the data noun integer, real, float, or double. Each type is different in nature and the effect is critical, particularly in database design domain. Thus, a RA is to carefully take into consideration an appropriate data type, and to define the exact data noun type as the correct range.

E91 shows an example of hidden noun data ranges that may induce vagueness due to hidden assumption of the noun data ranges.

E91: The system shall receive inputs from 10 up to 100.

Inputs denotes plural ambiguity and we do not know whether Inputs refers to the system expects many inputs entered at one key-stroke, or the system expects input entered repetitively. Furthermore, it is also difficult to tell whether the Inputs must be integer or real type, or even possible to accept floating point. If we know expected input is in the context of age where an age is a round integer figure, it's better to put this information inside the sentence as in E91.1.

E91.1: The system shall receive age input from 10 up to 100.

E92 gives another example where 20.35 concurrent accesses is unacceptable because rationally, the number should be integer type. The RA holds responsibility in identifying noun or entity in the RStats to not contain any hidden assumption data range.

E92: The system shall allow at most 20.35 number of concurrent accesses to the database.

Rule III.4: Avoid writing *S* with a compound name in referring two different kinds of noun.

Write Instead: *S* that derives a complete naming for each defined compound noun.

Discussion:

Disseminating equally a compound name for a noun such as in regulatory and utility technical requirements is a nuisance. Frankly speaking, it's difficult to say whether it actually means regulatory technical requirements and utility technical requirements or it means regulatory requirements and utility technical requirements. Furthermore, the use of coordinator such as conjunctive and and disjunctive or facilitates superfluous manipulation of a compound name in referring two different kinds of noun. Coordinator itself contributes to coordination ambiguity (which is discussed in **Chapter 4.** and the following **Structuring Conjunction** section) and pairing up cooordinator with a compound name in *S* is not a good practice. Thus, writing precise naming for each noun helps avoid potential ambiguity arising from a compound noun. For example, avoid stop or start message, instead write stop message or start message.

E93 shows an example of potentially ambiguous RStat introduced by the use of compound name for referring different kind of nouns.

E93: The user can enter all boat classes and series for all races.

Note that the context where E93 originates is The Yacht Race Result (YRR) program. E93 violates Rules III.4 and III.1 by its use of compound name and none referential bound to user. The writing of E93 is very ambiguous because classes and series share a compound noun reference which is boat. E93 violates also Rule VI.10 with the use of all classes, series, and all races which are in plural.

E93.1: The manager shall enter each boat class and each boat series for each boat race.

IV. <u>Structuring Adjective</u>

Adjective generally modifies a noun or a pronoun by describing, identifying, or quantifying a word. From the RS analysis finding, Adjective is in fact the main contributor in vagueing a RStat, and hence adjective should not be used. Unfortunately, it is very common to assert adjective in writing a non-functional RStat.

Rule IV.1: Avoid writing *S* containing any vague adjective such as prompt, fast, routine, velc[5].

Write Instead: *S* with a measurable time unit to replace the specified vague adjective.

Discussion:

Adjective such as prompt, fast, routine, velc. is normally used to describe the timing of a process e.g.:

[5] "velc." as ("vel cetera") is to inclusive "or" as "autc." as ("aut cetera") is to exclusive "or", as well as "etc." ("et cetera") is to "and"

E94: The Science Analysis Software performs prompt processing of Level 0 data to produce Level 1 event data.

Without understanding the context of system-to-be-built and precise specification, it's hard to explain the term prompt for "how prompt the acceptable measure is", fast for "how fast the acceptable scale is", and routine for "how routine the frequency is". Hence, replacing the vague adjective with an actual amount of time in a measurable time unit will inhibit ambiguity as in E94.1.

E94.1: The Science Analysis Software performs within 0.1 seconds the processing of Level 0 data to produce Level 1 event data.

Another example, E95 shows violation of Rules IV.1 and VI.2 by its use of prompt and should. The use of prompt is very vague because there isn't sufficient information to tell how speedy the system responds is considered prompt.

E95: The system should give prompt respond to all user inputs.

The suggested change of E95 is E95.1 with a precise time unit defined for prompt.

E95.1: The system shall give within 1 second respond to each user input.

Rule IV.2: Avoid writing *S* containing any vague adjectives such as ancillary, relevant, necessary, routine, velc.

Write Instead: *S* that contains a complete description to explain the specified vague adjective.

Discussion:

Adjective such as ancillary, relevant, necessary, routine, velc. requires the reader to do his or her own requirements analysis to make *S* a complete RStat.

E96: In support of high-level processing, the SAS extracts from the LAT and SC Level 0 data ancillary information relevant to event reconstruction and classification.

As illustrated in E96, it's not possible to analyse what kind of information is categorised as ancillary and which information is relevant to the context of event reconstruction and classification. Hence, replace the vague adjective with a complete description of whatever is ancillary, relevant, routine, velc., e.g.:

E96.1: In support of high-level processing, the SAS extracts the Ground Observational Data from the LAT and SC Level 0.

Another example, E97, violates Rule IV.2 because the word routine requires the reader to do requirements analysis to determine what sort of processing is really intended. Moreover, it is not clear if the missing information is timing or functional information.

E97: The SAS is responsible for routine Level 2 processing of the LAT data.

If the missing information is about the timing of the processing, then a suggested rewriting is E97.1.

E97.1: The SAS is responsible for daily Level 2 processing of the LAT data.

If the missing information is about the function of the processing, then a suggested rewriting is E97.2.

E97.2: The SAS is responsible for the Level 2 processing of the LAT data that computes the maximum, minimum, and average values.

In essence, notice that the word routine is a signal for two different rules, Rules IV.1 and IV.2 where vagueness has multiple uses. Hence, determination of the complete description may require consulting the stakeholders.

Rule IV.3: Avoid writing *S* containing common, generic, customary, velc.

Write Instead: *S* that contains description on the scope of the commonality, genericity, customariness, velc.

Discussion:

Common, generic, and customary have more than one scope that leads to ambiguity. To remedy the potential ambiguity arising from the use of vague adjective, it is necessary to correctly describe the property of a system-to-be-built's entity. For example, in E98, it is difficult to know if the instrument geometry is that which is known world wide in any analysis module or is that which is assumed in the specific analysis modules appearing in the system being specified by the RS.

E98: The simulation shall use instrument geometry that is defined and is common to all analysis modules.

Furthermore, E98 contains a violation of Rule VI.10. The suggested rewriting of E98 is E98.1.

E98.1: The simulation shall use instrument geometry that is defined and conforms to each analysis module.

Another example,

E99: All output messages shall be categorised (e.g. error, warning, debug) and reported via a common mechanism.

In E99, common is potentially ambiguous, because it can mean same or everyday. In the absence of access to E99's author, we assume that the intended meaning is same. E99 violates also Rule VI.9 or VII.2. Hence, E99 can be rewritten as E99.1.

E99.1: Each error, warning, and debug message shall be categorised and reported via the same mechanism.

Precise elimination of the vagueness of analysis module requires asking the RStat's author what he means.

V. Structuring Conjunction

The principal function of a conjunction is to relate two or more words, phrases, and clauses. There are three types of conjunction, namely:

- Coordinating conjunction such as and, but, or, nor, for, so, yet, to join individual words, phrases and independent clauses.
- Subordinating conjunction such as after, although, as, because, before, how, if, velc. to join dependent clause and indicate the nature of the relationship among the independent clause and dependent clause.
- Correlative conjunction such as both...and, either...or, neither...nor, not only...but also, so...as, whether...or, to link equivalent sentence elements.

Conjunction is also known as the main contributor to coordination ambiguity, and coordination ambiguity is a very common form of syntactic ambiguity in English. During the RS analysis, conjunction or coordinator such as and and or, are the most common cause of coordination ambiguity. Therefore, the following rules suggest better ways to write RStat that aim to prevent stepping into coordination ambiguity. Moreover, chapter 3. has already discussed the potential ambiguity varied from coordination conjunctive and and disjunctive or.

Rule V.1: Avoid writing *S* containing both *X* and *Y*.

Write Instead: *S* in the form *X* and *Y*.

Discussion:

Correlative conjunction both *X* and *Y* is used to link *X* and *Y* that play role as the compound subject or as compound object as in E100.

E100: The system should print reports for both users and clients.

The use of both is redundant because just and is enough to bind two components into one compound entity. Furthermore, since both is just simply and therefore discards both. E100 violates also rule III.1, where there is no precise referential for reports and Rule VI.11 on plural ambiguity. Rule V.1, VI.11, and III.1 suggest rewriting E100 into E100.1 and E100.2:

E100.1: The system should print inventory report for users.

E100.2: The system should print inventory reports for clients.

Another example, E101 shows violation of Rules V.1, VI.2, and II.2 by its use of both, will, and passive form. The violated rules suggest rewriting E101 into E101.1. Note that the plurality of employees doer is also changed to singular with the use of each.

E101: The Cask Loader software will be used primarily by employees of nuclear utilities, utilising both BWR and PWR technology.

E101.1: Each employee of nuclear utilities shall use the Cask Loader software, utilising BWR and PWR technology.

Rule V.2: Avoid writing *S* containing *X* but *Y*.

Write Instead: *S* in the form of *X* and *Y*.

Discussion:

The use of but varies and results different interpretations, which namely are:

- To suggest a contrast that is unexpected in light of the first clause, for example John is poor but he has a happy life.
- To suggest in an affirmative sense what the first part of the sentence implied in a negative way, for example Mary is never absent in History class but she is not here today.
- To connect two ideas with the meaning of giving a conditional exception, for example Nobody but Mary passes the History test.

For example, E102 demonstrates the use of but in a RStat and Rule V.2 suggests to rewrite E102 into E102.1.

E102: The LVL1 result will also provide secondary RoIs which did not pass the thresholds, but do pass lower thresholds.

E102.1: The LVL1 result will also provide secondary RoIs which did not pass the thresholds and do pass lower thresholds.

Since but is just another way of saying and, therefore a RA is to avoid using but and rewrite but into and when the RStat means to say and.

Certainly, one would concede that there is in fact some sort of contrast between E102 and E102.1. The replacement of but to and may have left something out because the original implies there is certain contrasted constraint between the secondary RoIs which did not pass the thresholds and the secondary RoIs which do pass lower thresholds. For this research's purpose, what matters is that the truth-value of the transformed RStat should always agree with the truth-value of the original RStat.

Another example, E103 shows violation of Rules V.2, VI.2, and V.5 by its use of but, can, and not only.

E103: The Cask Loader software can provide not only cask loading tracking support, but optimisation as described above.

The use of can in E103 is very ambiguous, because it is difficult to determine whether E103 is indeed a RStat or is only a suggestion. If E103 is indeed a RStat, use shall and not other words as suggested in Rule VI.2. Furthermore, the term as describe above is also potentially ambiguous as without a thorough analysis to the RS, it is not possible to know which description it actually refers to when there are more than one optimisation criteria specified in RS. After identifying the correct referential bind to the term as describe above, the suggested change to E103 is E103.1.

E103.1: The Cask Loader software shall provide cask loading tracking support and optimisation of loads for radiation and heat loading.

Rule V.3: Avoid writing *S* containing *X* and/or *Y*.

Write Instead: *X*, *Y*, or both[6].

Discussion:

The writing of *X* and/or *Y* is confusing because it is hard to distinguish whether it means only *X*, or only *Y*, or *X* or *Y*, or even *X* and *Y*. For example,

E104: An authorised user shall have the ability to edit and/or void a log entry.

The potential ambiguity of and/or confuses the reader whether to interpret E104 as E104.1, or as E104.2, or as E104.3, or even as E104.4. If E104 really means to say E104.1, then write E104.1, or likewise E104 to E104.2, or E104 to E104.3, or E104 to

[6] This use of both is not excluded by Rule V.1, which suggests avoiding both when it is combined with a following and.

E104.4. Since and/or is truth functionally equivalent with or and and/or carries the same logical interpretation as or, then rewrite *X* and/or *Y* to *X* or *Y*.

Rule V.3 suggests rewriting E104 into E104.3. Note that the rewriting of E104.3 avoids also the confusion of compound noun a log entry by distributing a log entry to each modifying verb edit and void.

E104.1: An authorised user shall have the ability to edit a log entry.

E104.2: An authorised user shall have the ability to void a log entry.

E104.3: An authorised user shall have the ability to edit a log entry or void a log entry.

E104.4: An authorised user shall have the ability to edit and void a log entry.

Another example, E105 has the same confusion resulting from the use of and/or. The suggested change according to Rule V.3 is to rewrite E105 into E105.1. Likewise, note that the rewriting of E105.1 avoids also the confusion of compound noun a login attempt by distributing a log attempt to each modifying verb discontinue and to cancel.

E105: The administrator shall have the ability to discontinue and/or cancel a login attempt during the login process.

E105.1: The administrator shall have the ability to discontinue a login attempt or cancel a login attempt during the login process.

Rule V.4: Avoid writing *S* containing *X*/*Y*.

Write Instead: *S* with *X* or *Y*.

Discussion:

Even though / means or, an occasional writer or reader believes that / means and, and he or she would be surprised when presented with a / replaced by an or. Thus, write or precisely when the RStat means to say or. For example,

E106: An authorised user shall have the ability to postpone/delete a scheduled task.

The suggested change according to Rule V.4 is to rewrite E106 into E106.1.

E106.1: An authorised user shall have the ability to postpone or delete a scheduled task.

Another example, E107 shows violation of Rules V.4 and VI.10 by its use of / and all.

E107: The system shall delete all triggered/acknowledged nonrecurring reminders.

The violated rules suggest rewriting E107 into E107.1. Note that both triggered and acknowledged modify the compound noun nonrecurring reminder. To avoid the potential ambiguity of compound noun, nonrecurring reminder follows immediately after triggered and after acknowledged.

E107.1: The system shall delete each triggered nonrecurring reminder or each acknowledged nonrecurring reminder.

Surprisingly in some cases, what results after rewriting an RStat is not what the writer intended, the stakeholder who owns a rewritten RStat must be asked if the new RStat is what he or she intended.

Rule V.5: Avoid writing *S* containing any correlative conjunction that equivalently means and, e.g., not only, but also, as well as, velc. that provides additional commentary.

Write Instead: Simply and to replace the specified correlative conjunction in *S*.

Discussion:

The use of not only, but also, as well as, velc. is unessential because it obscures the true intended meaning of an RStat. Thus, write precisely by omitting the unessential commentary contributed from correlating conjunction. For example,

E108: A reward system must be established not only for the individuals, but also for organisations and teams of employees.

According to the violated Rules V.5, the suggested rewriting of E108 is E108.1, E108.2, and E108.3:

E108.1: A reward system must be established for the each individual.

E108.2: A reward system must be established for each organisation.

E108.3: A reward system must be established for each team of employees.

E109 is another example that demonstrates the use of correlative conjunction as well as which causes ambiguity.

E109: The system shall process data received from users and clients as well as to produce a standard report on it.

Rule V.5 suggests to rewrite as well as to just simply and as indicated in E109.1.

E109.1: The system shall process data received from each user and each client and to produce a standard report on it.

E109.1 actually contains more than one RStats and carries the potential ambiguity resulting from the use of it. However, blindly rewrite EY1 into each separate RStat as in E109.2 and E109.3 doesn't eliminate the potential ambiguity of it as an anaphora.

E109.2: The system shall process data received from each user and each client.

E109.3: The system shall produce a standard report on it.

The use of it in E109 is potentially ambiguous because it is difficult to determine whether it refers to data received or refers to processed data which is obtained right after the system finishes processing the received data. If it refers to the received data, then the suggested rewriting of E109.3 is to E109.4. However, if it refers to the data resulting after processing, then the suggested change of E109.3 is E109.5.

E109.4: The system shall produce a standard report on the processed data.

E109.5: The system shall produce a standard report on the data received from each user and each client.

VI. Structuring Words and Phrases

For many years, dictionaries of English have recorded and defined the meaning of words, though they often differ considerably in which phrases they include. In general, recurrent words and phrases contribute to the textual organisation in a sentence and the meaning of a word is observable from the other words round about the word especially repeated patterns of co-occurrence. The meaning of a word is not independent of the environment, including the co-text, in which the word occurs. The meaning of a word is described as potentially ambiguous if the word is open to more than one interpretation or explanation. Hence, the following section sets out guideline rules in determining proper way to use certain words and phrases aim to reduce their word potential ambiguity.

Rule VI.1: Avoid the use of determiner the in distinguishing specific entity.

Write Instead: The actual element or group of element that have interaction with the specified system behaviour.

Discussion:

The use of the can be potentially ambiguous as in the user when there are groups of users with different authority such as the administrator, database designer, analyst, clerical staff, velc. and each of the users have different level of access to the system. It is potentially ambiguous because the specified behaviour of the system might not be actually true for all entities of the referred group of entities. Thus, identify precisely the type of user involves or interacts with the specified system's behaviour on that particular system's process. However, if the RStat means to include every type of users, regardless of their different level of access, then writing the user is acceptable.

Rule VI.2: Avoid writing *S* containing should, will, would, may, might, ought to, except as an expression of a preference that is not a requirement.

Write Instead: *S* using shall if *S* is indeed a RStat.

Discussion:

The word should or alternatively recommended mean that there exist valid reasons in particular circumstances to ignore a particular item with a careful consideration on its implication [Bradner, 1997]. The use of should in *S* causes *S* appears less important where fulfilling the requirements stated in *S* is optional. For example,

E110: The lift should not be stopped from fast mode but should always be switched to slow mode for at least 1 second before stopping.

E110 violates rules II.2, V.2, and VI.2 by its use of passive form, but, and should. The violated rules suggest rewriting E110 to E110.1.

E110.1: The control system shall not stop the lift from fast mode and shall always switch to slow mode for at least 1 second before stopping.

Another example, E111 show the improper use of should and E111.1 shows the rewriting of should to shall.

E111: The control system should not violate safety requirements.

E111.1: The control system shall not violate safety requirements.

Thus, if *S* is supposed to be a functional requirement, then the RA is to rewrite *S* using shall.

Rule VI.3: Avoid writing *S* containing a subjective option introduced by a keyword such as either, whether, otherwise, velc.

Write Instead: *S* that specifies under what condition each option happens.

Discussion:

When keywords such as either, whether, and otherwise, occur in *S* in a coordinate structure, they play role as conjunction to introduce alternative conditions. The use of either, whether, and otherwise, is acutely ambiguous as the intended process or action is offered as option rather than apparent process in *S*. If SREE is to process this RStat, SREE has to have the intelligence to understand semantically and decide the suitable option presented in *S* such that in E36.

It is definitely not possible for SREE or any ambiguity detection tool to decide when the user is to be trusted and when the user is not to be trusted. The term the user is also potentially ambiguous and violates rule V1.1, because there might be different level of authorisation that decide what kind of user is to be trusted. In addition, E36 violates rule II.2 that suggests avoiding passive form writing. Thus, write precisely under what condition each option happens. The violated rules suggest rewriting E36 into two distinct RStats which are E112 and E113. E112 identify the condition when not to trust a user, and E113 identifies the type of user to trust on.

E112: The system shall not trust an unidentified user.

E113: The system shall trust the ICA-group user.

Another example, E114 shows violation of Rule VI.3 and III.1 by its use of whether and none of noun reference bound to the term the new version.

E114: The system shall inform the user whether the new version is required or recommended.

The use of whether is potentially ambiguous because it invites a subjective decision to the reader of E114 in which different reader may have different decision. Thus, the violated rule suggests rewriting E114 to E114.1 and E114.2, specifying each respective condition for E114.1 and E114.2 to happen.

E114.1: The system shall inform the user the new version of a web-based solution is required after 1 month expiration.

E114.2: The system shall inform the user the new version of a web-based solution is recommended 2 weeks before expiration.

Rule VI.4: Avoid writing *S* containing an indefinite timing introduced by the keyword eventually, at last, finally, velc.

Write Instead: *S* specifying strict sequencing of events with no timing or specify timing with tolerances, both in measurable units.

Discussion:

The specification of indefinite timing contributes to vagueness and can be critical, because in a case as described in E115, it is obvious that the server has to receive data. Since there isn't any time specified i.e. how long the server must wait to get data, may cause the server will be in waiting mode infinitely, especially if the data is not delivered. Furthermore, the term data violates also rule III.1 and is very ambiguous because it's hard to tell without a referential noun binding to the data, the server must receive.

E115: When a client makes a one-way send, the server must eventually receive data.

The violated rules suggest rewriting E115 to E115.1.

E115.1: When a client makes a one-way send, the server must receive the sent data no later than 24 hours.

Another example, E116 shows violation of Rules VI.4 and III.1 by its use of finally and no noun reference bound to data. After analysing the RStat's context, the noun referential binding the term data is Level 0 and thus, the suggested change of E116 is E116.1.

E116: The SDP shall finally be able to receive data from LAT mirrored sites.

E116.1: The SDP shall be able to receive Level 0 data from LAT mirrored sites no later than 24 hours after completion of processing.

Thus, specify the timing defined in *S* precisely or tolerably timing in a measurable unit.

Rule VI.5: Avoid writing *S* containing a noun phrase containing maximum or minimum to define an uncertain time or unit space. Maximum or minimum plays role as an adjective modifying the main noun.

Write Instead: *S* specifying a definite time or data unit.

Discussion:

The terms maximum or minimum leaves an open interpretation which leads to subjective judgment on the allowed unit quantity, and consequently contributes to vagueness as illustrated in E117.

E117: The system shall return minimum results to the user.

Given E117, it's difficult to analyse how many result is acceptable as minimum. In addition, since E117 is an independent RStat, then E117 is very vague due to the unclear term results and unidentified the user. Therefore, the RA has to carefully analyse the context of E117 to find out the exact referential binding for results and also for the user, and to identify a measurable unit for minimum term. However, if the RStat means to say the system must return a search result to every user who performs a search through the system, then the term the user is acceptable.

E117.1: The system shall return at least 1 search result to the user.

Another example, E118 shows violation of Rule VI.5 by its use of minimum and the suggested change of E118 is E118.1. It is difficult to determine the acceptable refresh rate to be considered as a standard for minimum term. Note that E118 contains also plural ambiguity of stations and to avoid plural ambiguity, the plural term stations has been rewritten to singular term station preceded with each as shown in E118.1.

E118: The operator stations and technician stations must be capable of minimum refresh rates.

E118.1: Each operator stations and each technician station must be capable of minimum refresh rates not less than 10 per second.

Thus, write down the exact measurable or at least tolerable unit quantity to avoid vagueness introduced by maximum or minimum.

Rule VI.6: Avoid writing *S* containing phrases such as as much as possible, as many as possible, as little as possible, or as few as possible.

Write Instead: *S* with the phrase with a more detailed, complete characterisation of what is as much or as little as possible.

Discussion:

The use of as much as possible, as many as possible, as little as possible, or as few as possible, in *S* can be a nuisance because they invite vagueness of subjective interpretation. It's difficult to determine how many or what satisfies the term as much as possible or as many as possible and what is the acceptable limit for the term as little as possible or as few as possible. E119 shows an example of vague Rstat resulted from the use of as many as possible. E119 violates Rule VI.6 and VI.2 by its use of as many as possible and should.

E119: Simulated LVL1 output should accommodate as many ATLAS events as possible.

Given only E119, it is difficult to determine what and how many events the simulated LVL1 output is to accommodate. Hence, identifying the exact events is very critical and if E119 is supposed to be an absolute RStat, rewrites should to shall. The suggested change according to the violated rules is E119.1.

E119.1: Simulated LVL1 output shall accommodate at least Second Level Trigger, Event Builder, and Event Filter.

Thus, write down *S* with the exact measurable or at least tolerable unit quantity.

Rule VI.7: Avoid writing *S* containing under-specified phrases such as if necessary, if desired, as desired, velc. that invite subjective interpretation.

Write Instead: *S* specifying exact case as the functional requirement.

Discussion:

Under-specified phrases such as if necessary, if desired, as desired, velc. is normally used to code semantic ambiguity implicitly. Hence, this eliminates the need to generate all semantic interpretations that subsequently invite a perceived or subjective interpretation on which case to be considered.

E120 and E121 show examples of RStats containing the phrase as desired. The use of as desired is very confusing because it is difficult to tell the exact time or condition to implement the desire. Furthermore, the phrase as desired seems to signal that the RStat is not an absolute requirement of the specification, especially when the word may is used along as may signifies a truly optional RStat.

E120: User may block off certain cells within a cask, as desired.

E121: User may import the needed data from vendor specification sheets.

Rule VI.8: Avoid misplacing only, also, or other limiting word in *S*.

Write Instead: *S* where any only, also, or any other limiting word is written before the phrase the only, also, or other limiting word is intended to limit.

Discussion:

It is a very common mistake to misplace only or also immediately before the main verb of the containing sentence no matter which word is limited by the only or also. Hence, this leaves the reader is uncertain about what word is really limited by the only or also. For example,

E122: An associate can only view the report which contains the payment details entered by the associate himself.

Due to the misplacement of only, E122 actually means that the only task the associate does is to view the report and no other else. This might not be true when the associate has other tasks such as to prepare and sign the payment contract, to enter the payment details, and so on. If in contrast, what E122 really means to say is the associate can view only the report and can not do any changes i.e. modify, update, delete, on the report, then rewrites E122 to E122.1.

E122.1: An associate can view only the report which contains the payment details entered by the associate himself.

Another example of E123 shows the misplacement of only that contributes to potential ambiguity.

E123: HLT and LVL1 simulation should only use the same information to characterise LVL1 output data.

If E123 means to say the only thing that HLT and LVL1 simulation concerns is to use the same information, then E123 is the correct way to represent the stated RStat.

However, if HLT and LVL1 simulation does not concern only the use of same information, but HLT and LVL1 simulation has other to concern on as well, such as in reconstruction, configuration, detector and testbed. Then the violated rule suggests rewriting E123 into E123.1.

E123.1: HLT and LVL1 simulation should use only the same information to characterise LVL1 output data.

Rule VI.9: Avoid writing *S* containing some, many, few, several, e.g., velc. to describe a set of objects by example rather than by describing the set itself.

Write Instead: *S* that identifies precisely each element of the set.

Discussion:

In essence, some, many, few, several, denotes multiplicity. When each of them serves as a subject or as the main adjective of the subject, the subject ends up being plural and requires a plural verb. In some cases, it is much easier to describe sets of objects by example. Unfortunately, at least in English, there are no singular words corresponding to some, to many, and to few, and to several. The use of some, many, few, several, is confusing yet potentially ambiguous because it does not give any indication as to how many objects that are available. For example,

E124: Some of the software packages (e.g. each HLT algorithm, the selection control, the data access) shall be documented for both the user, developer, and maintainer.

Note that E124 violates also some other rules, i.e., Rules V.1 and VII.2. Instead, specify the specific instances that are supposed to be in the set as in E124.1.

E124.1: Each HLT algorithm, the selection control, and the data access shall be documented for the user, developer, and maintainer.

Another example, E125 contains violations of Rule VII.2 and Rule VI.9 or a violation of RuleVII.1.

E125: All users of the system shall login using some form of unique identification (e.g., username and password).

If the purpose of the information in the pair of parentheses is to give the only form of unique identification possible, then a suggested rewriting is E125.1.

E125.1: Each user of the system shall login by using his username and his password.

If the purpose of the information in the pair of parentheses is to give one possible form of unique identification, and it is truly the case that *any* form of unique identification is to be used for login, then a suggested rewriting is E125.2.

E125.2: Each user of the system shall login by using some form of unique identification.

Rule VI.10: Avoid writing *S* containing all, any, or both modifying a direct object when the intent of *S* is to describe what happens to each instance of the set that is described by the modified direct object.

Write Instead: *S* where each in place of all, any, or both to describe each instance of the set.

Discussion:

All and any are universal quantifier equivalents in that each is used to describe properties that hold for all members of some set, whereas both also generally describes all members in that same set. The use of all forces the use of plural which is potentially ambiguous in its own right. The use of any is confusing, because any can be interpreted as an existential quantifier instead of the desired universal quantifier, e.g.:

E126: The operator log will record all warning messages prompted by the system.

or

E127: The operator log will record any warning messages prompted by the system.

Instead, write each in place of all or any e.g.:

E128: The operator log will record each warning message prompted by the system.

However, not every instance of all or both should be replaced by each. When the intent of *S* is to describe something that happens to the entire set which is the direct object, use all or both, e.g.:

E129: The system must put all displayed text into one file, in order to facilitate software maintenance for developers and to ease future translations to local languages.

The difference between the use of all in E129 and the use of all in E126 is that in E129, the intention is to specify something that happens to the entire set of displayed texts, while in E126, the intention is to specify something that happens to each element of the set of warning messages. It is hard to describe this difference in a rule.

Thus, if *S* means to describe each instance of the set, then use each. Some might have wondered whether every should or could be used instead of each. Each and every appear to be interchangeable, as each is singular. However, when it is a must to emphasise the completely distributive nature of the event where each member in the common noun set is affected individually, use each. On the other hand, when a partially distributive event structure is sufficient in which some members are affected individually or they are affected in subgroups, use every. For example, given 5 apples,

(a) Ricky weighs every apple.

(b) Ricky weighs each apple.

In (a), Ricky may weigh each of apple1, apple2, apple3 by itself, but may weigh apple4 and apple5 together. While in (b), Ricky has to weigh the apples one by one, which is a completely distributive event where no two apples can be weighted in the same subevent. The evidence that each and every differ in meaning is comprehensively discussed in [Tunstall, 1998; Lawler, 2005].

But, when the intend of *S* is to describe the entire set, then use all or both, and include either altogether or together to further clarify the intent as in E129.1. Therefore, when the intent is clear, the potential ambiguity of all can be avoided.

E129.1: The system must put all displayed text together into one file, in order to facilitate software maintenance for developers and to ease future translations to local languages.

Another example, E130 contains a violation of each of Rule VI.10, Rule VI.11, Rule VII.2, and Rule VI.9.

E130: All mission elements shall withstand all environments (e.g., EMI, shock, and thermal) to be encountered from component fabrication.

If EMI, shock, and thermal are only some of the possible environments that can be encountered during component fabrication, then a suggested rewriting of E130 is E130.1.

E130.1: Each mission element shall withstand each environment that can be encountered during component fabrication.

If, on the other hand, EMI, shock, and thermal are all of the possible environments that can be encountered during component fabrication, then an alternative suggested rewriting of E130 is E130.2.

E130.2: Each mission element shall withstand EMI, shock, and thermal environments.

Rule VI.11: Avoid writing *S* containing a plural noun either subject or object.

Write Instead: *S* with a singular noun only.

Discussion:

A plural subject in *S* causes uncertainty whether the persons or entities constituting the subject are to act individually or collectively. It is difficult to determine how many predicate or object instance is related to each subject instance. In E131, it cannot be determined if each person in the room lifts his or her own table or if the all the people in the room as a group lift one table.

E131: All persons in the room lift a table.

Instead, use only singular subject in *S* as illustrated in E131.1 and E131.2.

E131.1: Each person in the room lifts his or her own table.

and

E131.2: The set of all person in the room lifts one table.

If using a plural subject is a must, then reserve it for describing properties of the entire set of subject instances, e.g.:

E131.3: All persons in the room together lift one table.

E132 and E133 show the potential ambiguities resulting from the use of all in writing a plural subject, in violation of Rule VI.11. Note that E132 has also (1) a violation of Rule VII.2, against the use of a pair of parentheses to enclose essential information and (2) a violation of Rule VI.9, against the use of e.g. to describe example elements of a set of objects instead of describing the set.

E132: All login attempts shall be done so in a secure manner (e.g., encrypted passwords).

E133: All pipeline products shall contain keywords, which describe the pipeline modules used to create them.

The violated rules suggest rewriting E132 and E133 into E132.1 and E133.1, respectively.

E132.1: Every login attempt shall be done with an encrypted password.

E133.1: Every pipeline product shall contain keywords that describe the pipeline modules used to create the pipeline product.

The change embodied in E133.1 assumes that each pipeline product is built from several pipeline modules. If each pipeline product is built from exactly one pipeline module, then E133 should be changed to E133.2.

E133.2: Every pipeline product shall contain the keyword that describes the pipeline module used to create the pipeline product.

Further example,

E134: Execution controls (start, pause, step, stop) will not be accessible if the setting is not made.

E134 has violated rules II.2, III.1, VI.2, and VII.2. Strictly speaking, it is difficult to tell what the setting that E134 refers to. Carefully analyse the domain's context, the referential binding to the setting is data value. The violated rules suggest rewriting E134 into E134.1.

E134.1: Start control, pause control, step control, stop control, shall not be accessible if the program does not make the data value setting.

Rule VI.12: Avoid writing *S* containing *A* unless *B*.

Write Instead: *S* containing If *not* (*B*), then *A*.

Discussion:

Unless is similar in meaning to if not and is frequently used in certain types of conditional sentences [BBC, 2007]. However, though less frequently, some people interpret unless to be if and only if not [Chandler, 1982]. There is an evidence that an occasional person uses *A* unless *B* as (*B*) if and only if *A* instead of If *not* (*B*), then *A* where *not* (*B*) means the logical negation of *B*. For example,

E135: Unless the user has the administrator's authorisation, the user will not be able to access the database.

E135 violates current Rule VI.12, Rule VI.2, and violates also Rule III.1 that suggests specifying an exact referential binding to user. The violated rules suggest rewriting E135 to E135.1.

E135.1: If the database user does not have the administrator's authorisation, the database user shall not be able to access the database.

Strickly speaking, *A* unless *B* says nothing about what happens if *B* is true. Therefore, if it is desired to specify that *C* happens if *B* is true, an explicit RStat, If *B*, then *C* must be given. The corresponding additional specification for E135 is E136.

E136: If the database user has the administrator's authorisation, then the database user shall be able to access the database.

Writing an unless RStat as its logical equivalent will force the person who misinterprets the unless RStat to see what the RStat really means. Thus, analyse carefully the true intend of unless whether it means to say if not or if and only if not, and rewrite it precisely.

Another example:

E137: The system will display registration alert unless the user has registered.

Likewise, E137 violates current Rule VI.12, Rule VI.2, and also Rule III.1. The suggested rewriting of E137 is E137.1.

E137.1: The system shall display registration alert if the authorised user has not registered.

In addition, the corresponding explicit RStat for E137 is E138.

E138: The system shall not display registration alert if the authorised user has registered.

As discussed in Section3.2.5, the use of unless in E135 and E137 show there exists temporal bond in which:

- In E135, the condition where the database user does not have a database authorisation will prohibit the event where database user to access the database to happen. Until a time such condition is met, then it triggers or allows the event to happen.

- In E137, the event where registration alert shall keep on showing up continuously happens until the condition where user does the registration is met or true.

Rewriting *A* unless *B* as if not *(B)*, then *A* shows that *A* unless *B* is not the same as not *(B)* if and only if *A*. Thus, writing an unless RStat as its logical equivalent will force the person who misinterprets the unless RStat to see what the RStat really means.

Rule VI.13: Avoid writing *S* containing any of meanwhile, whereas, on the other hand, velc. to lengthen the sentence.

Write Instead: *S* as short and simple declarative sentence as defined in Rule II.1.

Discussion:

Each such phrase is usually used to combine two or more related RStats. Each should be avoided as unnecessarily complicating or lengthening the containing RStat without providing any essential information. For example,

E138: Each officer can print the report by selecting an associate. Meanwhile, an associate can only view the report which contains the payment details entered by the associate himself.

Instead, rewrite *S* without the meanwhile, whereas, on the other hand, velc., as in E138.1 and E138.2.

E138.1: Each officer can print the report by selecting an associate.

E138.2: An associate can view only the report that contains the payment details entered by the associate himself.

Note that E138 has also a misplaced only that is moved to the correct place in E138.2 according to Rule 27. Moreover, which is changed to that in accordance with English rules.

Follow-up, E139 contains violation of Rules VI.13 and VI.2 by its use of should and whereas.

E139: The user manual should document the expected results whereas the user interface should provide information or warning indicating what changes will occur when a user changes the regional setting.

The violated rules suggest rewriting E139 as E139.1 and E139.2.

E139.1: The user manual shall document the expected results.

E139.2: The user interface shall provide information or a warning indicating what changes will occur when a user changes the regional settings.

Another example of the use of meanwhile:

E140: If the payment is with payee's details, then the system will treat each payment separately meanwhile if users choose "No", all the payment records will be grouped together to become one cheque.

In E140, meanwhile combines two RStats into one long RStat, in violation of Rule VI.13. Even though the second RStat has a plural subject and sues all in apparent violation of Rule VI.11, the RStat is describing a property of the entire set of payments, that they are grouped into one payment. Therefore, the suggested change of only Rule VI.13 is applied to split E140 into two RStats, E140.1 and E140.2.

E140.1: If the payment is with the payee's detail, then the system will treat each payment separately.

E140.2: If the user chooses “No”, all the payment records will be grouped together to become one cheque.

However, just splitting E140 into E140.1 and E140.2 does not make E140.1 and E140.2 into independent RStats, because they nevertheless have a temporal relationship; E140.2 follows E140.1. Rule II.4 suggests grouping together requirements that show any such temporal dependency. Once joined into one RStat, E140.1 and E140.2 regain the lost temporal context.

Rule VI.14: Avoid rewriting *S* by any of the other rules when between, among, amongst, is used in *S* to differentiate one action or process from another action or process described in the same RStat.

Write Instead: *S* as short and simple declarative sentence. If *S* must use keywords such between, among, amongst, make sure *S* has identifiable singular subject as the doer of the process, the intended process or action to be triggered, and the object.

Discussion:

The idea of rule VI.14 may contradict previously defined rule II.1. However, keywords between or among such as has a strong bound to the relationship established between the pair of actions or processes. This rule prevents upsetting any relationships that exist. For example,

E141: Restrictions between different types of data access, either logical or physical, made at LVL2 must be valid if the data are passed on to the online environment are stored and retrieved in the offline environment.

E141 is a long and complicated RStat that contains violations of Rule II.1., II.2, VI.3, and VI.10. The reference to logical or physical access is not clear, because there is insufficient information to tell whether logical access and physical access are the only two kinds of accesses. There is no explanation of who passes the data, of the kinds of data that can be passed, and of when the data are to be passed. The author takes guesses and rewrite E141 as E141.1 according to the recommendations of the violated rules.

E141.1: Restrictions between logical or physical data access made at LVL2 must be valid when observational data that the server passes to the online environment are stored and retrieved in the offline environment.

Rewriting E141 as E141.1 may have obscured the relationships that exist between type of data access other than logical and physical. Rule VI.14 prevents this possibly obscuring rewriting.

Another example,

E142: The system must prohibit direct public access between external networks and any system component that stores cardholder information.

E142 violates Rule VI.10. The use of between indicates that there is a relationship between external networks and cardholder information. Rule VI.10 suggests rewriting E142 as E142.1.

E142.1: The system must prohibit direct public access between each external network and each system component that stores cardholder information.

Rewriting E142 as E142.1 may have obscured the relationships between heretofore unidentified external networks and system components and thus may have modified the meaning of the original RStat. Rule VI.14 prevents these obscuring changes that upset relationships that exist between the arguments of the between.

Rule VI.15: Avoid writing *S* containing misuse words of specific, particular, respective, certain, velc. that superficially appears to make that RStat sounds precise.

Write Instead: *S* with the exact referential entity binds that denote distinctiveness.

Discussion:

Keywords such as specific, particular, respective, certain, in *S* is to indicate distinctive behaviour of the referred entity. However, the improper use and incomplete referential bind to the distinctiveness cause the RStat sounds seemingly precise. E143 shows the vague RStat introduced by the use of specific. Furthermore, E143 violates rules VI.16 and II.2,

E143: Report forms will be generated to meet specific utility needs.

E143 doesn't describe which exactly utility needs that are to be met and who generates the report forms. Note that the plurality of report forms has violated also rule VI.11. The violated rules suggest rewriting E143 to E143.1.

E143.1: The Cask Loader software shall generate report forms to meet funding utility needs.

The following E144 shows an example of a very ambiguous RStat that appears to be seemingly precise.

E144: The software will ensure that specific individual utility standards are utilised for special technical requirements.

It is difficult to tell which individual utility is indeed specific, who decides the individual utility standard to be specific, and how to determine the decision is correct. Besides, it is also hard to tell which technical requirements are considered special, who and how to determine that technical requirement is indeed special.

Rule VI.16: Avoid writing *S* containing any incomplete information briefing on the constraint that is imperative to that RStat.

Write Instead: *S* containing complete information that is imperative as a RStat. Identify exactly the doer, the process or action to happen, and the object influenced by the process in *S*. If there is a limitation to the process, identify precisely the referential bound to that limitation in *S*.

Discussion:

An incomplete *S* causes vagueness as the RA has to complete the missing information with his or her own interpretation. For example,

E145: The user will experience no practical limit to the number of cask models that the database will contain.

Note that E145 contains violations of Rules VI.2 and VI.16 by its use of will and no practical limit. The use of phrase no practical limit in E145 is very confusing, because it is difficult to judge whether there will be some other kind of limitation except from practical limitation. The author takes guesses that the no practical limit phrase comes from the idea of modeling infinite capacity with a finite container that is large enough for all practical purposes. Hence, the RA and client have to sit down and figure out how large the input ever going to be and then allocate space that is some percentage bigger. The author suggests that even the percentage must be agreed upon.

Another example, E146 is similar to E145 with the same inaccuracy due to incompleteness in *S*. Note also that E146 contains violations of Rules VI.2 and VI.16 by its use of will and no practical limit.

E146: The user will experience no practical limit to the number of fuel assemblies that the database will contain.

Likewise, the use of phrase no practical limit in E146 is confusing, because it is difficult to judge whether there will be some other kind of limitation except from practical limitation, e.g. theoretical, hypothetical, etc.. Besides, as what can be understood from E146 is that the database has the capacity to contain any number of fuel assemblies. This consequently gives implication that E146 means to say the database has an infinite capacity to store even very big number of fuel assemblies. Unfortunately, it's not possibly to tell from given E146, whether the implication is correct or not. The incomplete E146 doesn't really convey the exact RStat to give precise interpretation.

Rule VI.17: Avoid writing *S* containing phrase such as on the fly, but not limited to, velc. that requires the user of system-to-be-built to possibly further redefine the RStat at the system's run time.

Write Instead: *S* with tolerable and specifiable condition that the RStat should describe.

Discussion:

Phrase such as on the fly on the RStat allows certain flexibility, depending on the system's user's judgment whether to further make additional definition to the existing RStat during the system's run time or not. This flexibility causes potential ambiguity when the user of the system is more than one person because different user may have different opinion in deciding whether to add more definition to the existing RStat during the system's run time or not. Even if there is no conflict of interest between users, there might be inconsistency if more than one user does the additional definition on the RStat during the system's run time. For example,

E147: New fuel assembly types and cask models will be able to be defined on the fly.

E147 means to prompt the engineer, which is the user in this RStat, to define each new fuel assembly type and each cask model during the run time. If more than one engineers make the definition at the same time, then it will cause data inconsistency. In addition, E147 violates also rules II.2, VI.2, VI.11 and VI.17. The violated rules suggest rewriting E147 to E147.1.

E147.1: The engineer shall define each new fuel assembly type and each cask model.

Another example, E148 violates rules VI.17, VI.2, and III.1 by its corresponding use of on the fly, may, and user. As specified in BCP 14, the use of may in an RStat indicates that the RStat is an optional [Bradner, 1997]. Furthermore, analysing from the requirements' context, the referential bind to the term user is the engineer because the engineer is responsible for adding and updating the information on each cask model in the database.

E148: User may block off certain cells within a cask on the fly.

According to the violated rules and the referential information, the suggested change of E148 is E148.1.

E148.1: Engineer shall block off each unused cell within a cask.

VII. Structuring Symbols

Rule VII.1: Avoid writing *S* containing any pair of parentheses, braces, or brackets, i.e., ( ), { }, or [ ], that encloses unnecessary text.

Write Instead: *S* without any unnecessary text that is enclosed by a pair of parentheses, braces, or brackets.

Discussion:

Human commonly write unnecessary commentary inside a pair of parentheses or a pair of brackets that aims to further supplement the preceding word or phrase in a sentence. However, the difficulty to assess whether the information inside the pair of parentheses is indeed important information or is only an superfluous commentary, is one point to determine. The difficulty to decide whether or not to consider the information inside the parentheses is another point to remark.

If the information inside parentheses is not necessary, remove the whole information along with the parentheses. A requirement writer is to avoid writing *S* containing any parentheses or brackets. For example,

E149: The Web Browser is of a version that supports the running (execution) of Java applets, and of the version the Java applet is compiled.

The running of Java applets has the same operational meaning as execution of Java applets. This rule suggests omitting execution, and therefore E149 is to be rewritten into E149.1.

E149.1: The Web Browser is of a version that supports the running of Java applets, and of the version the Java applet is compiled.

Another example,

E150: The program is designed to be clearly viewable (understandable) on a screen size of 17 inch.

E150 is a non-functional requirement describing the visibility of the program on a 17 inch screen. Both viewable and understandable are vague adjectives because they open subjective opinion on the acceptable margin for viewable and for understandable. Note that E150 violates also rule II.2, hence the violated rules suggest rewriting E150 into E150.1.

E150.1: The design of program shall have 100% viewability on a screen size of 17 inch.

Rule VII.2: Avoid writing *S* containing any pair of parentheses, braces, or brackets, i.e., ( ), { }, or [ ], that encloses necessary text.

Write Instead: *S* where the necessary text is moved to its own RStat and remove pair of parentheses, braces, or brackets.

Discussion:

A RA sometimes erroneously writes important information inside parentheses and considers the information as commentary. For example,

E151: The motor polarity must not be changed whilst the lift is moving. (This could wreck the winding gear).

E151 violates rules VII.2 and II.2 by its use of parentheses and passive form. The violated rules suggest rewriting E151 to E151.1. Note that since must carries the same representation as shall as specified in BCP 14 [Bradner, 1997], then the use of must is retained in E151.1.

E151.1: The control system must not change the motor polarity whilst the lift is moving as the change shall wreck the winding gear.

E152 shows an example of important feature cut that is placed inside parentheses. Rule VII.2 suggests removing the parentheses and moving cut to its own RStat as rewritten in Rule E152.1.

E152: The administrator shall have the ability to copy (or cut) text to the system clipboard and paste the text to other text-accepting component.

E152.1: The administrator shall have the ability to copy text or cut text to the system clipboard and paste the text to other text-accepting component.

Another example,

E153: The program will operate with the window maximised or minimised allowing no intermediate adjustment (resizing).

The purpose of putting resizing inside a pair of parentheses is to clarify what no intermediate adjustment means, avoiding possible vagueness. E153 is a good example indicating the preference of exclusive or interpretation. Since it is clearly a necessity to include resizing, this rule suggests to rewrite E153 into E153.1.

E153.1: The program will operate with the window maximised or minimised allowing no window resizing.

Rule VII.3: Avoid writing *S* containing any pair of parentheses, braces, or brackets, i.e., ( ), { }, or [ ], in which the purpose of the pair of parentheses, braces, or brackets is to cause *S* to mean two or more RStats.

Write Instead: the text inside parentheses to become another RStat, and remove pair of parentheses, braces, or brackets.

Discussion:

Without careful analysis to the information inside parentheses, one may just consider the information to be useless or unnecessary commentary and ignore the information which is supposed to be another RStat. For example,

E154: Turning the switch down (up) turns the light on (off).

Instead, rewrite *S* as a sequence of as many RStats that *S* means as in E155 and E156.

E155: Turning the switch down, turns the light on.

E156: Turning the switch up, turns the light off.

Another example,

E157: The lift should never be allowed to move above the top floor or below the bottom floor. (There is an emergency shut down system that will stop the motor if the lift goes above the top floor or below the bottom floor (by more than 10 cm) but this shut down system is beyond the scope of the control system.)

Instead, rewrite *S* as a sequence of as many RStats that *S* means as in E158 and E159.

E158: The control system shall not allow the lift to move above the top floor or below the bottom floor.

Even though, there is an additional commentary specifying the emergency shut down system is beyond the scope of the control system, however this information still serves as an RStat to the emergency shut down system and not to the control system.

E159: The emergency shut down system shall stop the motor if the lift goes above the top floor or below the bottom floor.

The guiding rules presented in this section deal with the use of NL in RStats. Each of the guiding rule is constructed based on the analysis of existing RSs regarding the problems arising with the use of NL, and each of the guiding rules is tailored to examples to clarify the derivation and practicality.

7 Validation: Analysis and Rewriting of RSs

The guiding rules defined in Chapter 6 are validated by applying them to inspect RStats from RSs in the corpus. Since the author has no access to any RS writer, the validation process is based solely on the author's understanding of the RS at hand. The aim of guiding rules is the reduction of potentially ambiguous RStats in the RS, regardless of whether the RStat is a functional RStat or a non-functional RStat.

In order to evaluate the guiding rules, the author rewrote potentially ambiguous RStats found in [BPS, 2005; CLS, 1999; DCS, 2002; EVLA, 2003; LAT, 2000; PESA, 2001] by applying the guiding rules. Whenever a RStat does not follow one of the guiding rules, the way in which the RStat does not follow the rule is explained and the suggestions of the violated rule are followed to guide the rewriting of the RStat.

E160 shows a good example of a long and complicated RStat that violates Rule II.1. There are more than one processes to be executed and more than one object output corresponding to the stated processes. Compounding E160's complication is a coordination ambiguity.

E160: The simulation shall simulate passage of incident particles through the instrument and spacecraft, and produce an output representing the true energy depositions in the instrument and spacecraft, as well as the particle parentage tree at least as deep as the daughter e+ and e- in photon interactions.

The suggested rewriting of E160 is E161 and E162.

E161: The simulation shall simulate passage of incident particles through the instrument and spacecraft.

E162: The simulation shall produce an output representing the true energy depositions in the instrument and spacecraft, as well as the particle parentage tree at least as deep as the daughter e+ and e- in photon interactions.

E163 – E165 are the example RStats that violate rule II.1 by their uses of potentially ambiguous adjectives to complicate the RStats. The use of sufficient in E163, near in E164, and necessary in E165, cause vagueness in each of the RStats. There isn't enough information to detail the sufficiency amount in E163, nearness scale in E164, and necessity criteria in E165.

E163: The Spacecraft Contractor shall transmit sufficient information to allow OSEM operations personnel to operate the system safely and successfully for its design life.

E164: The Science Analysis Software shall provide near real time monitoring information to the IOC.

E165: The CL software will create the necessary reports and records to comply with NRC, DOE, and utility plant requirements.

E166 is written in passive voice and hence violates Rule II.2. Inside the parentheses is important information about the condition that triggers a warning message. Since the information inside parentheses is important, the information shall be moved to the RStat as suggested in Rule VII.2.

E166: Warning message will be shown if associates choose the wrong options for the Payee Address (i.e. the Payee Address of the particular Payee for subsequent Payments does not match with the address which has been selected for the first Payment within the same batch).

The violated rules suggest rewriting E166 into E167.

E167: If the Payee Address of the particular Payee for subsequent Payments does not match with the address which has been selected for the first Payment within the same batch, the system shall show a warning message.

E168 shows another example that violates rule II.2 by its use of passive voice.

E168: Summary on the number of payments and total amount for Direct Credit and Cheque will be displayed on the footer of the report.

The violated rule suggests rewriting E168 as E169.

E169: The BPS application shall display summary on the number of payments and total amount for Direct Credit and Cheque on the footer of the report.

The term data in E170 does not bind an exact reference, which violates Rule III.1. Furthermore, data covers a broad range, which contributes to potential ambiguity. The author analyses the detailed business requirements in Batch Poster System that the term data can be used to refer payee information, payment information, and vendor information.

E170: The application will only hold the data for 10 days.

E170 contains also violation of rule VI.2 from the use of will as it causes misconception of E170 as preference that is not a requirement. On the other hand, E170 violates Rule VI.8 from the misplacement of only that causes E170 to mean the application's task is to hold the data and no other task. The true intend use of only in E170 is to limit data holding for 10 days. Hence, the violated rules suggest rewriting E170 into E171 where the exact reference binding to data and the reference binding to application are both identified.

E171: The BPS application shall hold the payment data only for 10 days.

E172 violates rule VI.10 because any shall not be used to describe each instance of the set. Moreover, E172 violates also rule II.2 with the use of passive, and rule VII.2 in which the important information is specified inside parentheses.

E172: Any payments (for both normal and Import File) that are 10 days old will be automatically deleted due to space precaution.

The violated rules suggest rewriting E172 into E173. Note that the 10 days old is changed 10 day old to in accordance to English rules.

E173: The BPS application shall automatically delete each normal payment and each Import File that are 10 day old due to space precaution.

E173 can be rewritten into E173.1 and E173.2 for further clarity by eliminating the coordination ambiguity.

E173.1: The BPS application shall automatically delete each normal payment that is 10 day old due to space precaution.

E173.2: The BPS application shall automatically delete each Import File that is 10 day old due to space precaution.

E174 is another example that violates Rule VI.10 and II.2 by the use its use of all and passive voice.

E174: All data produced by SAS shall be cataloged and archived in a secure manner at the SDPF.

The violated rules suggest rewriting E174 as E175.

E175: The Science Analysis software shall catalogue each data produced by SAS and shall archive each data produced by SAS, in a secure manner at the SDPF.

Note that the term secure is vague as there is no precise definition on the criteria satifies secure manner. The writer of this RS is required to supply the fit criteria of secure manner.

Another example, E176 shows violation of rule VI.10 by its use of all and rule II.2.

E176: All hardware and software shall be designed to meet applicable operational, reliability, environmental, and safety requirements.

The modification of E176 to E177 takes into account the recommendation of the violated rules.

E177: The Spacecraft Contrator shall design each hardware and each software to meet operational requirements, reliability requirements, environmental requirements, and safety requirements.

Another example, E178 gives an example RStat that violates rule VI.11 due to plural ambiguity. The violated rule suggests rewriting E178 into E179.

E178: The system shall have the ability to link maintenance database entries to log entries.

E179: The system shall have the ability to link each maintenance database entry to each log entry.

The change embodied in E179 assumes that each maintenance database entry is linked to one individual log entry. However, if E178 means the set of maintenance database entries altogether is linked to one individual log entry, then E178 should be changed to E180.

E180: The system shall have the ability to link all maintenance database entries into one log entry.

E181 contains violation of rule VI.9 by its use of some and rule VII.2 by its use of parentheses to describe a set of objects by example.

E181: The even reconstruction shall take input raw data from some instrument (engineering models, calibration unit, and flight instrument).

Instead, specify that specific instances that are supposed to be in the set as illustrated in E182.

E182: The even reconstruction shall take input raw data from engineering models, from calibration unit, and from flight instrument.

E183 gives an example RStat that demonstrates the potential ambiguity of parentheses. The information inside parentheses is important in expressing the vagueness of adverbs dynamically and statically, and hence the information should not be kept inside parentheses. The violated rule VII.2 suggests rewriting E183 into E184.

E183: The user shall be able to view the plot dynamically (real-time) or statically (offline).

E184: The user shall able to view the monitor point plot in real-time or the user shall be able to view the monitor point plot in offline.

Note that the change embodied in E184 removes the vague adverbs and moves the information inside parentheses into its own RStat. The change identifies also the exact noun reference binding to the plot and avoid the potential ambiguity caused by disjunction or.

E185 is the example RStat that illustrates the parentheses enclosing unnecessary information, which violates rule VII.1.

E185: The SAS shall provide command line (scriptable) and graphical interfaces for the LAT response simulation environment.

Instead, rewrite E185 as E186, eliminating the parentheses with the unnecessary text.

E186: The SAS shall provide command line interface and graphical interface for the LAT response simulation environment.

E187 gives example RStats that together fall under the province of rule II.4 and should be grouped together.

E187: For validation of the entire SAS processing chain, SAS shall provide a tool to generate celestial gamma-ray sources as input to the instrument simulation. The instrument simulation shall include the capability to model pulsed gamma0ray emission and GRB.

E188 and E189 contain violation of rule VI.15 by its use of specific, certain that creates misconception of precise RStat.

E188: The user shall have the ability to create a report based on specific purposes: Observer's Log, Guard Check-in Log, Antenna Visit Log, and Operator Notes.

We suggests rewriting E188 into 4 different RStats which are E188.1, E188.2, E188.3, and E188.4, removing the misconception of specific keyword and pluralness of purposes.

E188.1: The user shall have the ability to create a report based on Observer's Log.

E188.2: The user shall have the ability to create a report based on Guard Check-in Log.

E188.3: The user shall have the ability to create a report based on Antenna Visit Log.

E188.4: The user shall have the ability to create a report based on Operator Notes.

Unfortunately in E189, though rule VI.15 managed to identify the vagueness of certain, however there is neither explanation on the kind of data preparation steps nor the explanation on the kind of requested data. Hence, we are unable to suggest rewriting E189 into a more precise RStat.

E189: HLT algorithms should be able to indicate that certain data preparation steps are to be performed on the requested data.

E190 and E191 give the example RStats that violate rule VI.16 due to under-reference and incomplete information. In E190, the use of acronym TBR that means To Be Resolved indicates that E190 is yet to resolve certain formatting issue that is not decided at the time the RS is produced.

E190: The interfaces to the databases shall be independent of the high-level analysis software and provide the Level 1 and Level 2 and higher-level data in TBR formats.

Correspondingly, the use of common in E191 violates rule IV.3 as common contributes to ambiguity. E191 violates also rule VI.9 by its use of e.g. in describing a set of objects by example. Compounding to this ambiguity is the incompleteness of E191 in describing a RStat as signified by "(list to be completed)".

E191: Common data classes shall be provided for use by HLT algorithm, i.e. all classes needed to represent raw data, classes for reconstructed objects: e.g. track, cluster, … (list to be completed); classes for trigger elements: e.g. electron, photon, jet, … (list to be completed).

E192 shows an example of the conjunctive and with multiple verb phrases carrying temporal order. Clearly, it is not possible for both verb phrases to trigger their actions at the same time. The action triggered by the first verb phrase has to happen before the the execution of the action triggered by the second verb phrase.

E192: The system shall automatically download and install the new software.

To avoid the coordination ambiguity caused by multiple verb phrases, E192 is to be rewritten to E193. Note that there is not enough information to identify what new software E192 refers to, and hence the vagueness of new requires clarification from the RS writer.

E193: The system shall automatically download the new software and then the system shall automatically install the downloaded software.

8 SREE

This chapter discusses the goals, design, and implementation of SREE, a prototype tool for finding instances of indicators of many, but not all, kinds of ambiguity described in the guiding rules. As mentioned elsewhere in this thesis, the main purpose of SREE is to aid a user of SREE to identify potential ambiguities in a RS. SREE searches for instances of potential ambiguity in its input, concentrating on achieving a 100% recall rate and a precision rate of as close as possible to 100%. The potential ambiguities that SREE searches for are the ones whose indicators are listed in SREE's ambiguity indicator corpus (AIC). Consistent with the goal of 100% recall, SREE shall notify its user of each occurrence of one of these indicators. However, the fact that not every occurrence of a potential ambiguity is truly ambiguous causes SREE to have less than 100% precision.

In essence, SREE is a lexical analyser. When SREE finds a word in its input matching one of the indicators in its AIC, then SREE notifies its user by printing out a message describing the kind of potential ambiguity suffered by the word. Section 8.1 explains the goals of SREE, and Section 8.2 discusses why SREE does not use a parser. Section 8.3 lists the types of potential ambiguity that SREE searches for and describes the use of SREE with a scenario showing how SREE finds a potential ambiguity in a RStat. Section 8.4 explains how well each of the guiding rules is handled by SREE. Section 8.5 explains the design of SREE. Section 8.6 illustrates the input and output of SREE. Finally, Section 8.7 discusses properties of SREE.

8.1 Goals of SREE

SREE was designed to achieve 100% recall and a precision of not much less than 100%. SREE's recall is the percentage of the instances of potential ambiguity that are inside SREE's scope that SREE actually finds. SREE's precision is the percentage of the potential ambiguities that SREE finds that are truly ambiguous as identified by SREE's human user. The goal is for SREE is to have 100% recall even if it costs less than 100% precision. SREE cannot have the intelligence necessary to determine if a potential ambiguity is a true ambiguity, but a human user does have the intelligence. So, it will be better for SREE to recall all occurrences of a potential ambiguity and let its user to decide whether or not each potential ambiguity is truly ambiguous. However, for any ambiguity type for which SREE cannot achieve 100% recall, it is better for SREE not to look for that kind of potential ambiguity at all. If SREE is unable to achieve 100% recall for a particular type of potential ambiguity, then SREE's reporting the instances it can find is not very useful to the user, because, then, in any case, the user has to look for the other instances manually in the original input. The user might as well look for all instances of that type of potential ambiguity. Of course, the user does not have to look at all for instances of any potential ambiguity for which he or she *knows* that SREE has 100% recall.

8.2 Lexical Analyser in SREE instead of Syntactic Analyser

This section discusses the author's initial work in constructing a SREE based on a syntactic analyser or part-of-speech (POS) tagger and the reasons why she chose a lexical analyser over a syntactic analyser to be the basis of SREE. Initially, she obtained a collection of words as a dictionary corpus from WordNet 3.0. She constructed also an ambiguity indicator corpus (AIC) that consists of a list of all ambiguity indicators that she found in the corpus of RSs and in previous research, as discussed in Chapters 4 and 5. Then, she developed a syntactic analyser as an experimental program, which uses the two corpi as the source of references in assigning to each word a POS tag. Basically, the two main functions that the syntactic analyser performs are:

- Scanning, a.k.a. lexical analysis

  The scanner reads the input RS that normally consists of multiple RStats. Each RStat is transformed to a stream of tokens, in which each token is a word. The scanner works by scanning each character one by one until it reads a separator, i.e., a space, a tab, or a new line, which signifies the end of one token. For example the RStat The system shall store 20 GB of processed data per day consists of eleven tokens, namely The, system, shall, store, 20, GB, of, processed, data, per, and day.

  Next, the program looks for each token in the alphabetically arranged dictionary corpus and gets its POS tag. SREE uses hashing by the first letter of a token to search for an input token in the dictionary corpus. Given 26 letters, A–Z, SREE has 26 hashtables, and each hashtable associates a token with a list of all of its possible POSs. So, when SREE finds a token inside the dictionary, SREE returns all of the token's POS tags. In case of multiple POS tags, the correct POS tag is chosen by the parser described below. Numerals such as 0–9 are labeled as "Digit". For any unidentified token or token that is not found in the dictionary, SREE assigns an "Identifier" tag.

- Parsing, a.k.a. syntactic analysis

  For the development of the parser, the author developed the context-free grammar, written in extended Backus-Naur Format (EBNF) [Watt and Brown, 2000], for parsing declarative RStats and conditional RStats that is shown in Table 10. The parser implements a recursive-decent parsing algorithm. The grammar is not left recursive and thus avoids the problem of infinite recursion in recursive-descent parsing. Basically, the parser reads the scanner-generated stream of tokens with their lists of possible POS tags and parses the stream according to the grammar. For any token, if its first POS tag does not match that of the current point in the grammar, the parser tries to continue the parse according to the the next available POS tag for the token.

Declarative RStat	
Declarative RStat	::= Subject Predicate .
Subject	::= aNoun | NounPhrase
NounPhrase	::= Determiner aNoun
	| Determiner [Adverbial] [aAdjective *] aNoun
	| Pronoun [Adverbial] [aAdjective *] aNoun
Predicate	::= VerbPhrase Complement [Adverbial]
VerbPhrase	::= shall [not] aVerb
	| must [not] aVerb

	| shall [not] be able to aVerb
	| must [not] be able to aVerb
Complement	::= aNoun
	| NounPhrase
	| [Adverbial] aAdjective aNoun
	| PrepositionalPhrase
Adverbial	::= aAdverb *
	| PrepositionalPhrase
PrepositionalPhrase	::= aPreposition aNoun
	| aPreposition NounPhrase
Conditional RStat	
Conditional RStat	::= Condition Reaction
	| Reaction Condition
Condition	::= If Subject VerbSingular Complement
	| If and only if Subject VerbSingular Complement
	| Unless Subject VerbSingular Complement
VerbSingular	::= aVerb (s|es)
Reaction	::= [then] Subject VerbPhrase Complement
Legend	
'|' indicates alternatives	
'*' indicates the previous item may be repeated zero or more times	
'(' and ')' indicates grouping parentheses	

Table 10. Grammar for Declarative RStat and Conditional RStat

The author conducted a parsing experiment on the YRR [Bray, 2002] case study, which contains 16 RStats with 418 words, and the Lift Controller case study, which contains 22 RStats with 478 words. The unknown words are 1 out of 418 from YRR, and 3 out of 478 from Lift Controller. The parse time is between 0.7 and 0.9 second.

Although the parser is designed to parse according to the grammar shown in Table 10, it is unable to correctly parse more than about 80% of its input due to ambiguity of POSs and the difficulties of identifying proper nouns and of distinguishing plural nouns from singular indicate present tense verbs. The parser may be improved with a more heuristic strategy that does profound search and match and has a machine-learning algorithm. Nevertheless, there is no guarantee of being able to achieve 100% accuracy in parsing and POS tagging. A SREE based on a parser would therefore not be able to achieve 100% recall of any potential ambiguity whose recognition depends on a correct parse or a correct POS assignment.

However, using a pure lexical strategy allows SREE to achieve 100% recall of any potential ambiguity for which its indicators can be put into the AIC. That lexical analysis enables 100% recall (See discussion on page 148 - 149 about SREE's conception in lexical analysing and SREE's definition on recall) shows that lexical analysis is much better than syntactic analysis with its 80% recall. Therefore, the author prefers SREE to be based on lexical analysis with its 100% recall even at the expense of less than 100% precision. The human user of SREE can use his or her knowledge to decide whether any potential ambiguity is truly ambiguous or is a false positive.

8.3 Usage of SREE

For reasons that are discussed in Section 8.2, SREE does only lexical searches for potential ambiguities. There are two kinds of potential ambiguities from the user's viewpoint:

i. Those in the scope of SREE, because they are lexically identifiable with 100% recall

ii. Those not in the scope of SREE, because they cannot be identified lexically with 100% recall

Therefore, a user of SREE can know (1) the kinds of potential ambiguities are inside SREE's scope and that SREE will reliably find and (2) the kinds of potential ambiguities that the user has to look for manually because they are outside SREE's scope. Some example scenarios for the use of SREE to help find potential ambiguities in a RS are:

1. A boat can not be entered into the system unless it has a boat class or sail number.

   SREE recognises can, unless, it, and or to be potential ambiguities appearing in the AIC.

2. HLT algorithms should be able to indicate that certain data preparation steps are to be performed on the requested data.

   SREE recognises algorithms, steps, should, and that, to be potential ambiguities appearing in the AIC. Athough also certain should be a potential ambiguity, SREE is unable to recognise certain as a potential ambiguity because certain does not appear in the AIC and is thus outside SREE's scope. The user of SREE has to add certain to the AIC to enable SREE's recognition of certain as a potential ambiguity in the future.

3. Design shall be consistent with specified aircraft loadings.

   SREE recognises consistent to be a potential ambiguity. The user determines that specified aircraft loadings is ambiguous because there is no direct specification referred to specified, and the typical reader will not be able to guess what are the intended aircraft loadings. SREE is unable to recognise specified and loadings to be potential ambiguities because specified and loadings do not appear in the AIC. Again, the user of SREE has to add these potential ambiguities to the AIC to enable SREE's recognition of specified and loadings as potential ambiguities in future.

8.4 Implementation of Disambiguation Guiding Rules in SREE

The different categories of potential ambiguity indicators in SREE's AIC as discussed in Section 8.5 come from the guiding rules in Chapter 6. The following list identifies for each guiding rule the subset of its potential ambiguities that is inside SREE's scope.

- **Rule I.1 (Domain Context)** is outside SREE's scope because SREE does not have the intelligence to figure out which acronyms have been defined and which acronyms have not been defined. It is up to SREE's user to verify that for each acronym or abbreviation appearing in a RS, the acronym or abbreviation has an explanation in the RS's glossary.
- **Rule II.1 (Long RStat)** is inside SREE's scope with the potential ambiguity indicators and, or, whereas, and meanwhile.

- **Rule II.2 (Passive RStat)** is outside SREE's scope because without a parser or POS tagger in SREE, SREE is unable to identify any form of passive writing such as on the occurrence of be preceding *VerbPastParticiple*, or on the occurrence of is or are preceding *VerbPastParticiple*.
- **Rule II.3 (Implicit RStat)** is inside SREE's scope with potential ambiguity indicators there is and exists in.
- **Rule II.4 (Dependent RStat)** is outside SREE's scope because SREE is not intelligent enough to understand the dependency or relationship between one RStat to another RStat. However, potential ambiguity indicators it and they, which generally used to denote under-referenced anaphora from related RStats, are detectable by SREE.
- **Rule III.1 (Referential Noun)** is outside SREE's scope because SREE will never be able to guess which noun does not have a noun referential binding
- **Rule III.2 (Anaphoric Noun)** is inside SREE's scope with potential ambiguity indicators it, they, them, and that.
- **Rule III.3 (Indefinite Noun)** is outside SREE's scope because SREE is unable to decide whether or not the noun is in an exact range. For example, deciding whether any data noun is in float type instead of long type requires a human's analysis of the difference between long and float.
- **Rule III.4 (Compound Noun)** is inside SREE's scope with potential ambiguity indicators and and or, each of which is to combine two different nouns or two different verbs.

- **Rule IV.1 (Vague Adjective)** is inside SREE's scope with potential ambiguity indicators prompt, fast, and routine.
- **Rule IV.2 (Vague Adjective)** is inside SREE's scope with potential ambiguity indicators ancillary, relevant, necessary, and routine.
- **Rule IV.3 (Vague Adjective)** is inside SREE's scope with potential ambiguity indicators common, generic, and customary.
- **Rule V.1 (Conjunction)** is inside SREE's scope with the potential ambiguity indicator both.
- **Rule V.2 (Conjunction)** is inside SREE's scope with the potential ambiguity indicator but.
- **Rule V.3 (Conjunction)** is inside SREE's scope with the potential ambiguity indicator and/or.
- **Rule V.4 (Conjunction)** is inside SREE's scope with the potential ambiguity indicator /.
- **Rule V.5 (Conjunction)** is inside SREE's scope with potential ambiguity indicators not only, but also, and as well as.
- **Rule VI.1 (Determiner)** is outside SREE's scope SREE is unable to analyse lexically whether or not a use of the is appropriate for referring to a group of entities either collectively or distributely. In fact, to analyse the appropriate use of the requires semantic knowledge.
- **Rule VI.2 (Weak Auxiliary)** is inside SREE's scope with potential ambiguity indicators should, will, would, may, might, and ought to.

- **Rule VI.3 (Subjective Option)** is inside SREE's scope with potential ambiguity indicators either, whether, and otherwise.
- **Rule VI.4 (Indefinite Time)** is inside SREE's scope with potential ambiguity indicators eventually, at last, and finally.
- **Rule VI.5 (Indefinite Unit)** is inside SREE's scope with potential ambiguity indicators maximum and minimum.
- **Rule VI.6 (Indefinite Unit)** is inside SREE's scope with potential ambiguity indicators as much as possible, as many as possible, as little as possible, and as few as possible.
- **Rule VI.7 (Indefinite Specification)** is inside SREE's scope with potential ambiguity indicators if necessary, if desired, and as desired.
- **Rule VI.8 (Vagueness)** is inside SREE's scope with potential ambiguity indicators only, also.
- **Rule VI.9 (Indefinite Quantifier)** is inside SREE's scope with potential ambiguity indicators some, many, few, and e.g..
- **Rule VI.10 (Indefinite Quantifier)** is inside SREE's scope with potential ambiguity indicators all, any, both.
- **Rule VI.11 (Plural Noun)** is outside SREE's scope because without a parser, it is not possible to differentiate whether a word is a plural noun or a singular verb, each with an s ending. However, SREE's AIC has a plural corpus that contains 11,287 plural nouns, each with an s ending. Thus, SREE takes the instances in the plural corpus as indicators of the plural noun potential ambiguity.

- **Rule VI.12 (Vagueness)** is inside SREE's scope with potential ambiguity indicator unless.
- **Rule VI.13 (Lengthy Phrase)** is inside SREE's scope with potential ambiguity indicators meanwhile, whereas, and on the other hand.
- **Rule VI.14 (Dependent Instance)** is outside SREE's scope in order to prevent upsetting any existing relationship between the RStats, which cannot be detected anyway by SREE.
- **Rule VI.15 (Vagueness)** is inside SREE's scope with potential ambiguity indicators specific, particular, respective, certain.
- **Rule VI.16 (Imperative Phrase)** is outside SREE's scope because SREE is unable to detect whether an RStat is imperative or incomplete. However potentially ambiguous keyphrase and keywords, such as no practical limit, TBD, TBA, and TBS, are identified as potential ambiguity indicators in the incomplete corpus in the AIC. Thus, SREE will report when it reads any of these potential ambiguity indicators in a RStat.
- **Rule VI.17 (Imperative Phrase)** is inside SREE's scope with potential ambiguity indicators on the fly, and but not limited to.
- **Rule VII.1, VII.2, and VII.3 (Symbols)** are inside SREE's scope with potential ambiguity indicators ( ), { }, and [ ].

To sum up, SREE adopts the guiding rules that are inside SREE's scope and will search with 100% recall for occurrences in any RS of potential ambiguity indicators that are in the AIC. The potential ambiguity indicators mentioned in each rule make up

SREE's AIC, which is discussed in the following section. See 8.5 for more details on SREE's handling of some of these potential ambiguity indicators.

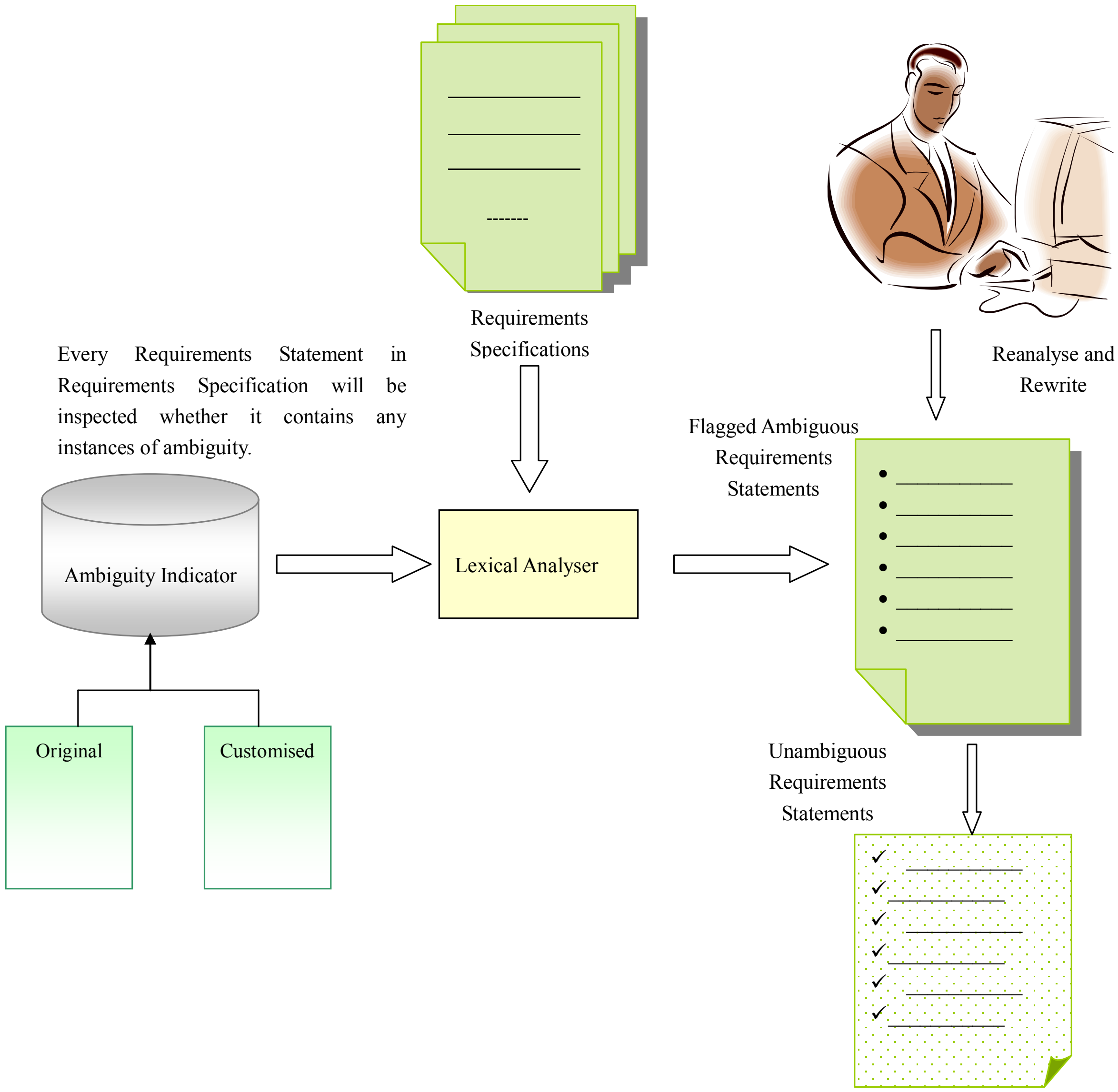

Figure 26. The Architecture of SREE

8.5 Design of SREE

This section describes the architecture of SREE. The architecture that permits SREE to be modular, extensible, and easy to use is shown in Figure 26. Basically, the lexical analyser used is that described as the scanner in Section 8.2. The main processes inside SREE are:

I. The **AIC** contains the corpus of indicators of potentially ambiguous keywords, keyphrases, and symbols. Although it may not be possible to have an AIC that contains an indicator of every possible potential ambiguity due to the richness of NL, SREE allows its user to add new indicators to its AIC. There are two categories of AICs in SREE, the original indicator corpus (OIC) and the customised indicator corpus (CIC). The OIC contains ten corpi, each in a separate file and each named appropriately for the nature of the potential ambiguities indicated by its contents: *Continuance*, *Coordinator*, *Directive*, *Incomplete*, *Optional*, *Pronoun*, *Plural*, *Quantifier*, *Vague*, and *Weak*. Each of these corpi has its own list of indicators. SREE automatically loads these corpi into the AIC each time a user starts up SREE. The indicators in these corpi are:

i. Continuance

The *Continuance* corpus contains potential ambiguity indicators such as :, as follows, below, following, in addition, in particular, listed, meantime, meanwhile, on one hand, on the other hand, whereas. For example,

E194: The facilities shall handle the following peak profile: 7 international aircrafts with an average passenger loading of 200 and 5 cargo planes an hour and 10 domestic aircrafts per hour with an average passenger loading of 100 and 2 Helicopters per hour.

SREE reports following to be a potential ambiguity and is a type of continuance because following introduces further specification at a lower level. Meantime, in

E195, since SREE matches ":" (colon) to the indicator : that appears in the *Continuance* corpus, SREE reports : to be potentially ambiguous. But, it is up to the user to decide whether the instance is a potential or a true ambiguity.

E195: Domestic: Shall be made available no more than 15 minutes after landing.

ii. Coordinator

The *Coordinator* corpus contains the potential ambiguity indicators: and, and/or, and or. E196 illustrates an RStat example that has an and occurrence, which contributes to a coordination ambiguity. However, in E197, the use of and does not introduce any coordination ambiguity despite SREE's detection of and as a potential ambiguity.

E196: The business community (potential investors in the project) like the current airport location and have voiced their opposition to building the new airport 100km north of the city.

The use of and is generally known to cause a coordination ambiguity. Therefore, whenever there is an occurrence of and, SREE reports and to be a potential ambiguity and of the type coordinator.

E197: The breakdown between domestic and international is 50:50.

iii. Directive

The *Directive* corpus contains potential ambiguity indicators such as e.g., etc., figure, for example, i.e., note, and table. For instance, SREE reports E198 and E199 to contain figure which is one of the potential ambiguity indicators as defined in the *Directive* corpus.

E198: The MCSS is functionally composed of a number of individual subsystems, as shown in Figure 5-1.

E199: Each MCSS rack shall contain a bus bar connected to the MSOCC ground, as shown in Figure 5-2.

iv. Incomplete

The *Incomplete* corpus contains the potential ambiguity indicators: TBA, TBC, TBD, TBE, TBS, TBR, as a minimum, as defined, as specified, in addition, is defined, no practical limit, not defined, not determined, but not limited to, to be advised, to be defined, to be completed, to be determined, to be resolved, and to be specified. E200 illustrates the use of as specified and E201 illustrates the use of as a minimum, which are then reported by SREE as potential ambiguities.

E200: Security measures as specified in Appendix A shall be put in place to screen in-going and outgoing passengers, baggage and cargo. These shall be in line with the recently revised International Security Standards adopted by the International Civil Aviation Organization (ICAO).

E201: This subsystem shall perform a self-test on the MCSS upon power up. The self-test shall, as a minimum, verify the operational status of all controllers and perform a memory check on all RAM memory.

v. Optional

The *Optional* corpus contains the potential ambiguity indicators: as desired, at last, either, eventually, if appropriate, if desired, in case of, if necessary, if needed, neither, nor, optionally, otherwise, possibly, probably, and whether. An example of a RStat that contains a potential ambiguity indicator from the *Optional* corpus is shown in E202.

E202: The MCSS shall be capable of operating on either one or both of its independent power supplies at any one time.

vi. Plural

The *Plural* corpus contains a list of 11,287 plural nouns, each ending in s. We differentiate the terms "*pluralnoun*" from "*plural noun*". The term "*pluralnoun*" is what is detected by SREE as a result of its use of the *Plural* corpus. The term "*plural noun*" is the collection of nouns, which are of plural types. SREE has 100% recall of *pluralnouns* as defined in its Plural corpus, but not of all of uEnglish *plural nouns*.

Reading every word with s ending to be a plural noun may give 100% recall of plural nouns but it gives also a very low precision because not every word with s ending is a plural noun. Hence, the *Plural* corpus is meant to help SREE identifies the plural nouns with s ending and also some of the irregular plural words, with 100% recall and not too low precision. It is not SREE's intention to search all kinds of English plural nouns and yet SREE is not able to report potential ambiguities caused by every kind of English plural nouns. SREE searches only words that are defined as plural in its *Plural* corpus. Basically, the words defined as plural noun are common plural words ends with s or es. In order to avoid an overly long list of writing this set of plural nouns in this documentation, the user of SREE can refer to the complete list in SREE's OIC. While SREE is comparing any token it has read with potential ambiguity indicators of the *Plural* corpus, if SREE detects any of these indicators, SREE reports a message to its user and lets the user to decide whether the reported potential ambiguity is truly ambiguous. E203 and E204 are examples of RStats that contain potential ambiguities as defined in the *Plural* corpus.

E203: The LCT shall make and break connections between sets of inputs and outputs as identified in a command file.

E204: This subsystem shall require the use of passwords to set up a session to access configuration tables and files.

The fact that the *Plural* corpus consists of only 11,287 plural nouns means the *Plural* corpus doesn't contain a complete list of plural nouns. Although, it is ultimately impossible to have a complete list of plural nouns and thus a complete corpus in the AIC due to the richness of NL, a user of SREE may always add any plural noun that is not listed in the *Plural* corpus to SREE's CIC.

vii. Pronoun

The *Pronoun* corpus contains the potential ambiguity indicators: anyone, anybody, anything, everyone, everybody, everything, he, her, hers, herself, him, himself, his, i, it, its, itself, me, mine, most, my, myself, nobody, none, no one, nothing, our, ours, ourselves, she, someone, somebody, something, that, their, theirs, them, themselves, these, they, this, those, us, we, what, whatever, which, whichever, who, whoever, whom, whomever, whose, whosever, you, your, yours, yourself, and yourselves. E205 and E206 give examples of RStats that contain potential ambiguity indicators as defined in the *Pronoun* corpus.

E205: The business community (potential investors in the project) like the current airport location and have voiced their opposition to building the new airport 100km north of the city.

E206: The noise level in built-up areas shall meet EPA specifications. These are contained in Annex APA.

viii. Quantifier

The *Quantifier* corpus contains the potential ambiguity indicators: all, any, few, little, many, much, several, and some. E207 and E208 give examples of RStats that contain potential ambiguity as defined in the *Quantifier* corpus.

E207: Air-bridges for all aircrafts with seating capacity greater than 70 shall be provided.

E208: Background diagnostics tests shall not change any existing switch connections.

ix. Vague

The *Vague* corpus contains the potential ambiguity indicators: /,< >, ( ), [ ], { }, ;, ?, !, adaptability, additionally, adequate, aggregate, also, ancillary, arbitrary, appropriate, as appropriate, available, as far as, at last, as few as possible, as little as possible, as many as possible, as much as possible, as required, as well as, bad, both, but, but also, but not limited to, capable of, capable to, capability of, capability, common, correctly, consistent, contemporary, convenient, credible, custom, customary, default,definable, easily, easy, effective, efficient, episodic, equitable, equitably, eventually, exist, exists, expeditiously, fast, fair, fairly, finally, frequently, full, general, generic, good, high-level, impartially, infrequently, insignificant, intermediate, interactive, in terms of, less, lightweight, logical, low-level, maximum, minimum, more, mutually-agreed, mutually-exclusive, mutually-inclusive, near, necessary, neutral, not only, only, on the fly, particular, physical, powerful, practical, prompt, provided, quickly, random, recent, regardless of, relevant, respective, robust, routine, sufficiently, sequential, significant, simple, specific, strong, there, there is, transient, transparent, timely, undefinable, understandable, unless, unnecessary, useful, various, and varying. The use of only in E209 causes potential ambiguity as defined in the *Vague* corpus. Meantime, the use of specific in E210 contributes vagueness and causes potential ambiguity as defined in the *Vague* corpus.

E209: This subsystem shall only execute troubleshooting diagnostics under LCT control.

E210: This subsystem shall not permit connection commands for a specific connection to disrupt any existing connections.

x. Weak

The *Weak* corpus contains the potential ambiguity indicators: can, could, may, might, ought to, preferred, should, will, and would. The use of will in E211 and E212 contribute to potential ambiguity as defined in the *Weak* corpus.

E211: This subsystem will identify at least 11 control message errors.

E212: The MSOCC Transition Plan requires all equipment installed in the future to generate their own timing signals. In the meantime, the MCSS will provide timing signals for all equipment within the MSOCC that require this capability.

Bear in mind that the user is not allowed to modify or delete any of the original corpi in the OIC. A user may add to the CIC any potential ambiguity indicator that he or she may find that is not in the AIC. He or she may also remove from the CIC indicators that have proved less than helpful.

II. The **Lexical Analyser** scans a RS, RStat by RStat, and scans each RStat, token by token, for any occurrence of any indicator in the AIC. During the scan, the lexical analyser of SREE recognises tokens from its input RS and compares each token with each indicator in the AIC. If SREE finds a match, it reports the token as a potential ambiguity.

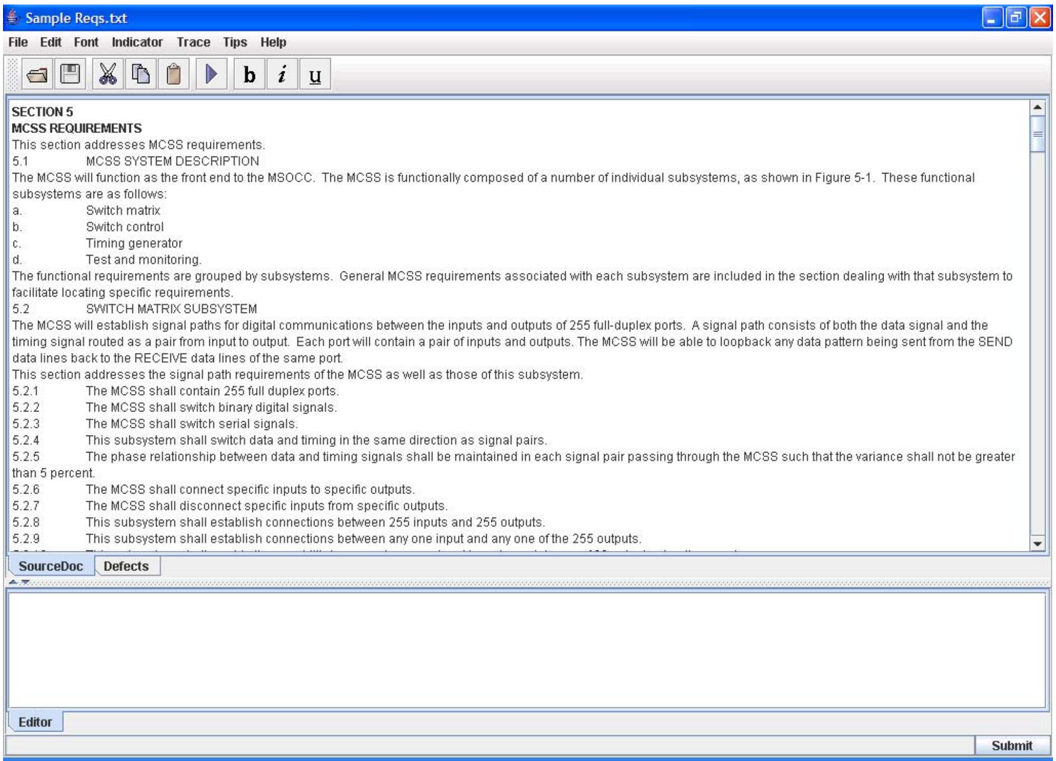

Figure 27. Lexical Analyser that loads a RS to the SourceDoc tabbed pane[7].

[7] Tabbed Pane is known also as a Window or Tab.

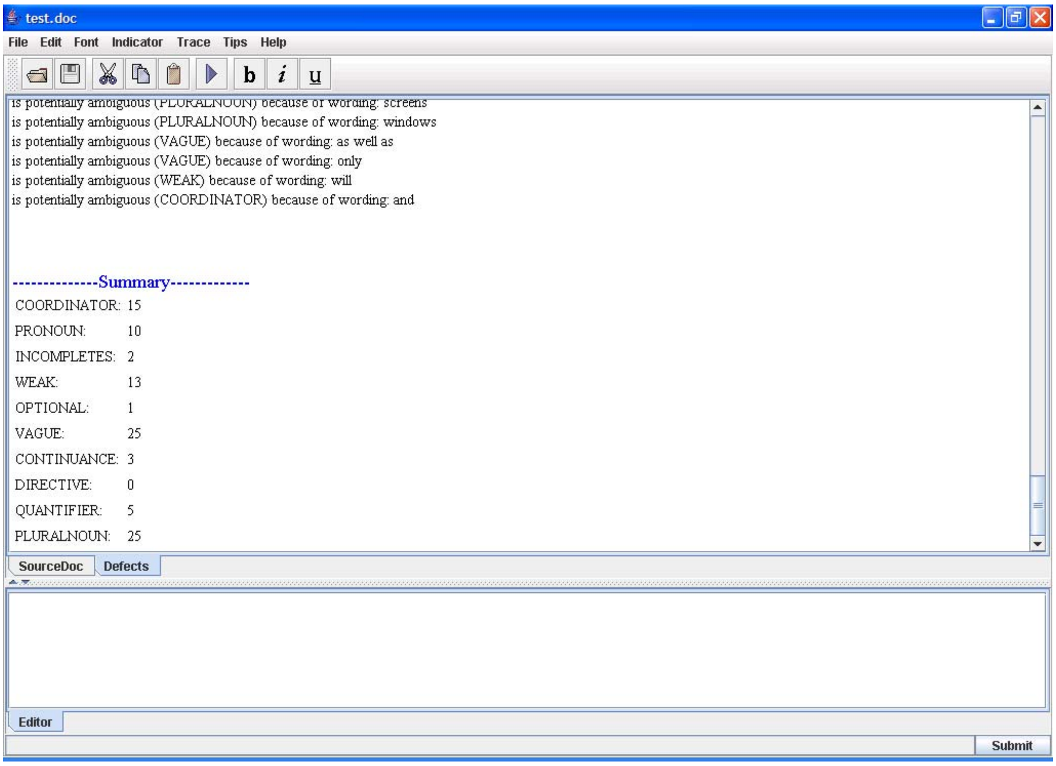

Figure 28. In the Defects tabbed pane, the Lexical Analyser reports a RStat with a description of the type of potential ambiguity

The user may click on the link in the line that contains a potential ambiguity instance, and SREE displays the instance's RStat in the Editor tabbed pane. The Editor tabbed pane provides a workspace for the user to examine and possibly rewrite the RStat. To allow a user to see a potential ambiguity's context, SREE displays the RStats preceding and following the RStat containing the potential ambigutity, as illustrated in Figure 29.

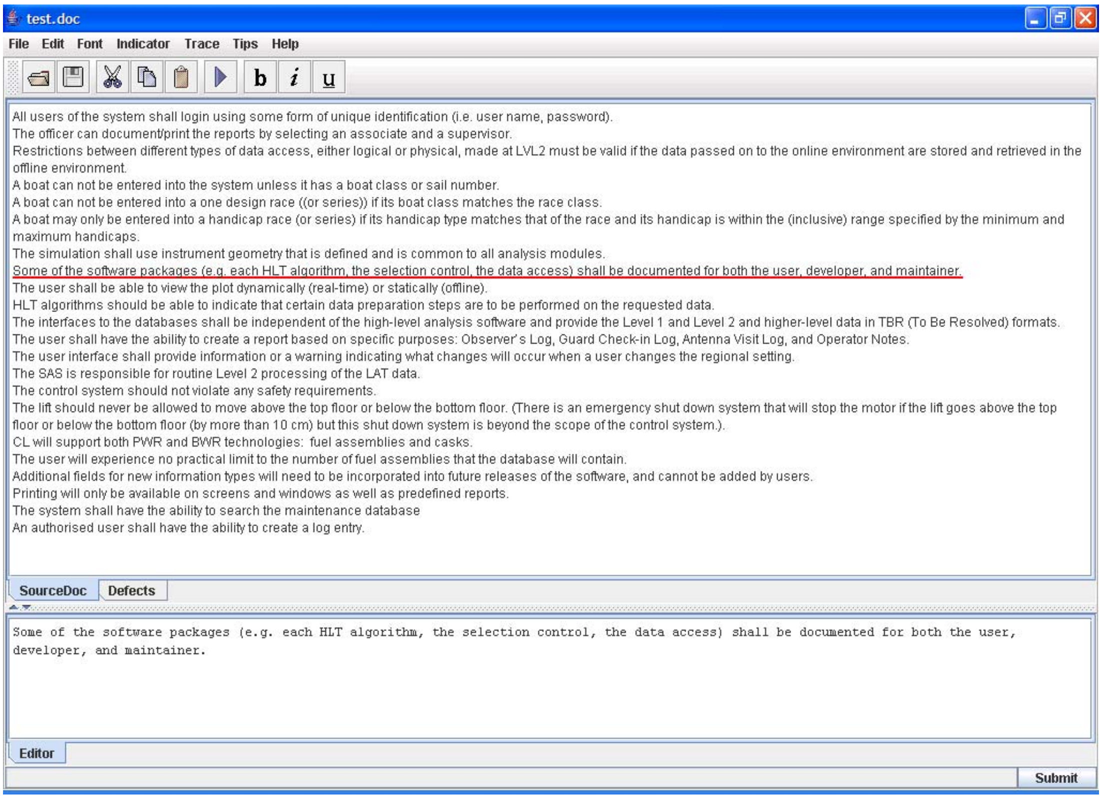

Figure 29. In the Editor tabbed pane, SREE displays the clicked RStat along with the preceding and following Rstats

A = an instance of ambiguity in the AIC

C = {A_1, A_2,…, A_n}

W = a word in an RStat

RStat = {W_1, W_2,…, W_n}

RS = {$RStat_1$, $RStat_2$,…, $RStat_n$}

scan.findInLine = method that takes an RS as input and scan each RStat into line

by line. scan.findInLine is a method from java.util.Scanner

pattern = an instantiation from Pattern class from java.util.regex.Pattern

foundAmbiguity = null // foundAmbiguity is set either to **True** when it finds A, or to **False** when it doesn't find A

For each A in C,

if(hasPunct(A)) // Punctuation: One of !" #$%&'( )*+,-./:;<=>?@[\ ]^_`{ | } ~

foundAmbiguity = scan.findInLine(A)

if(foundAmbiguity is null) {

//to match pattern e.g. (X X) where X X are words

//for the category of NOT_ALPHA, see java.util.regex.Pattern

pattern = Pattern.compile(NOT_ALPHA + keyword + NOT_ALPHA)

foundAmbiguity = scan.findInLine(A)

}

```
if(foundAmbiguity is null) {
        //to match pattern e.g. (X X) where XX are more than one line long of
        phrases or sentences
        //for complete explanation on LINE_END, see java.util.regex.Pattern
        pattern = Pattern.compile(NOT_ALPHA + keyword + LINE_END)
        foundAmbiguity = scan.findInLine(A)
}
if(foundAmbiguity is null) {
        //to match pattern e.g. X X) where X X are phrase or sentence continued
        from previous line
        //for complete explanation on LINE_BEGIN, see java.util.regex.Pattern
        pattern = Pattern.compile(LINE_BEGIN + keyword + NOT_ALPHA)
        foundAmbiguity = scan.findInLine(A)
}
```

Algorithm 1. The method of *scanAmbiguity* in SREE

The *scanAmbiguity* method given in Algorithm 1 shows that SREE utilises *java.util.Scanner* and *java.util.regex.Pattern* in Java™ and customises the class according to SREE's needs for matching keywords, keyphrases, and symbols in the AIC. For each potential ambiguity found by the *scan.findInLine()* method, SREE reports the name of the corpus that contains the matched word. The corpus's name effectively classifies the potential ambiguity.

8.6 Input and Output of SREE

SREE accepts the following inputs:

- A written RS: In SREE's main frame, as shown in Figure 30, the user is asked to enter a valid URL that points to the RS, as shown in Figure 31 to cause the RS to be loaded and processed as in Figure 32. The RS file should be a plain text file in either the *.txt* or *.doc*[8] format.

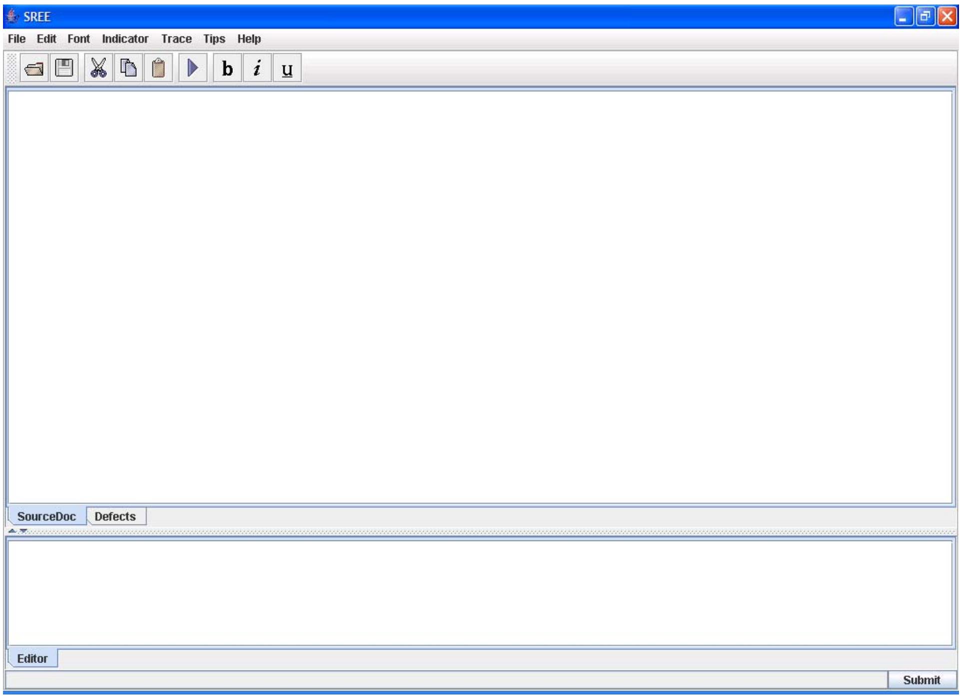

Figure 30. SREE GUI Frame

[8] SREE supports only MS. Word 2000 and MS. Word 2003.

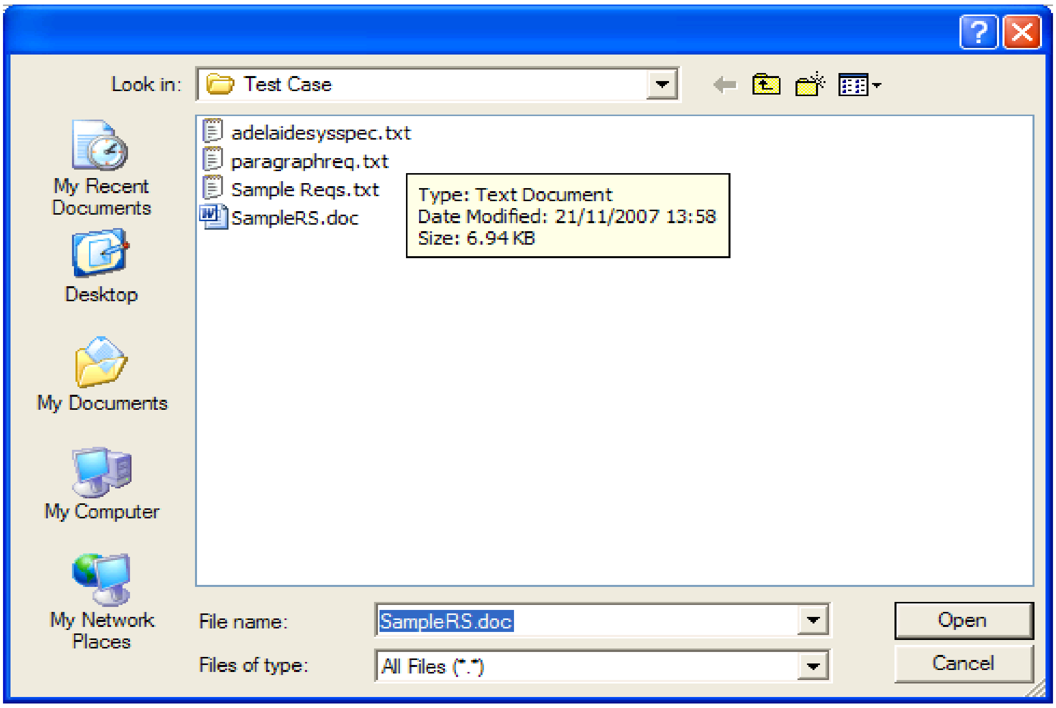

Figure 31. Open File Dialog that allows opening RS of *.txt* or *.doc* type

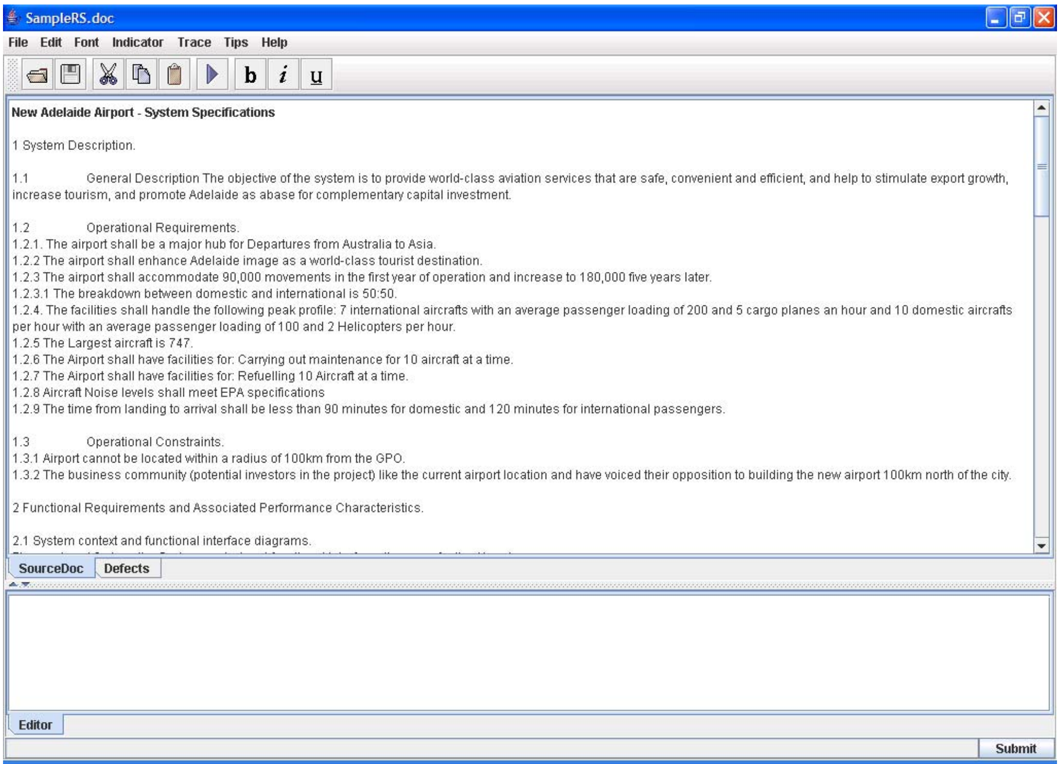

Figure 32. SREE loads New Adelaide Airport RS into the SourceDoc tabbed pane

- Additions to the AIC: As explained in previous section, the user is able to add potential ambiguity indicators into SREE's CIC whenever he or she finds a new kind of ambiguity with a lexically identifiable indicator. He or she may update or delete any indicator already in the CIC. However, he or she is not allowed to change the OIC.

SREE displays the following output:

- A list of RStats with potential ambiguities, as shown in Figure 28. For each RStat identified, SREE highlights the potential ambiguities it found in the RStat. The user may save this list for further analysis.

8.7 Properties of SREE

The properties that SREE should have are:

- 100% Recall

Recall is the percentage of the instances of potential ambiguity that are inside SREE's scope that SREE actually finds. Being inside SREE's scope is the same as being in SREE's AIC. SREE searches for only indicators it has in its AIC and reports every occurrence of any of them in the input. Therefore, SREE achieves 100% recall for potential ambiguities in its scope. For any potential ambiguity for which no indicator is in the AIC, SREE allows its user to add to its CIC indicators of the potential ambiguty.

- Precision of not too much less than 100%

Precision is the percentage of the potential ambiguities that SREE finds that are truly ambiguous as determined by SREE's human user. Whenever the user of SREE decides that a potential ambiguity that SREE finds is not truly ambiguous, the user ignores the potential ambiguity. Alternatively, the user may decide that the potential ambiguity is truly ambiguous. Imprecision is caused by SREE's inability to differentiate true ambiguities from potential ambiguities.

- Tailorability

SREE displays a list of those RStats that have potential ambiguities. Each potential ambiguity is classified by type. When the user clicks on a RStat in the Defects tabbed pane, SREE displays that RStat along with the preceding and succeeding RStats from the RS in the Editor tabbed pane. The user may rewrite the RStat and then save it. The user may rerun SREE on the modified RStat to ascertain that no new errors or ambiguity are introduced.

- Report Generator

At the end of SREE's analysis of an input, SREE displays a report that gives a summary list of the potential ambiguities it found for each category of indicator in the AIC.

Chapter 9 shows that SREE has the desired properties and that it correctly identifies a high percentage of all potential ambiguities. SREE notifies its user of all potential ambiguities it finds and leaves to its user the job of performing disambiguation.

9 Comparison of SREE with Ambiguity Finding Tools

The importance of correctly documenting requirements has led the software industry to produce a number of tools for activities ranging from the creation to the management of RSs. However, there are only a few tools, such as ARM, TIGER Pro, and QuARS that that help to identify ambiguities in a RS as one of the kinds of defects that can occur in a RS. SREE was developed as an experimental tool to help identify the potential ambiguities described by the guiding rules described in Chapter 6. Chapter 8 explains the rationale for SREE's being based on a simple lexical analyser with 100% recall rather than on a potentially more powerful syntax analyser with only about 80% recall. This chapter compares SREE with some of the other tools. It compares SREE's effectiveness in finding potential ambiguities in three industrial RSs to that of TIGER Pro and of ARM.

9.1 Overview of Ambiguity Finding Tools

ARM, TIGER Pro, and QuARS are tools that help to find potential ambiguities in NLRSs.

- ARM (Automated analysis of Requirement Specifications)

ARM is software developed by NASA's SATC (Software Assurance Technology Center) to assess the quality of a RS and to identify risks that a low quality RS may introduce to the development of its specified system [Wilson et al., 1997]. Among the signs of low quality found by ARM is ambiguity. ARM assesses the structure of its input RS and of the RS's individual RStats. It assesses the vocabulary used to state the requirements. The different categories of indicators in ARM are:

- Imperatives are those words and phrases that command that something must be provided.
- Continuances are phrases that follow an imperative and precede the definition of lower level requirement specifications.
- Directives are words or phrases that indicate that the document contains examples or other illustrative information.
- Options are those words that give the developer latitude in the implementation of the specification that contains them.
- Weak Phrases are clauses that are apt to cause uncertainty and leave room for multiple interpretations.
- Incomplete is the category of words and phrases that indicate that the specification of requirements is not fully developed.

- TIGER Pro (Tool to InGest and Elucidate Requirements Professional)

TIGER Pro [Kasser, 2004] is a tool developed by Joseph E. Kasser to parse a requirements document that has been exported from an ASCII text file or from a DOORS[9] [Telelogic, 2008] database. The tool is then to examine each requirement and produce a summary of defects the tool finds in the requirement. The examination is based on a set of *poor words*. The tool searches for five types of potential defects in the requirements. Basically, TIGER's concept in defining the types of defects is

i. to find each RStat that contains more than one requirement by searching for each RStat with more than one occurrence of shall, which is a defect of Type 1. However

[9] DOORS is a commercialised requirements management software developed by Telelogic AB

when shall is not used in a RStat, TIGER marks the RStat as containing a defect of Type 4.

ii. to find each instance of a keyword such as and that possibly combines multiple requirements, which is a defect of Type 2.

iii. to find each instance of a keyword such as all that is not verifiable, which is a defect of Type 3.

The user is to determine if a potential problem is a real problem and then, if so, he or she fixes it.

- QuARS (Quality Analyser of Requirements Specifications)

QuARS [Fabbrini et al., 2000; Fabbrini et al., 2002] tries to detect linguistic defects and to highlight each RStat containing any. A highlighted RStat contains at least one instance of an indicator of linguistic defects. QuARS uses lexical and syntax analysis to search for instances of these indicators in its input.

9.2 Differences among ARM, TIGER Pro, and SREE

SREE shares the same basic goal with ARM, TIGER Pro, and other ambiguity finding tools, which is to avoid writing ambiguous RSs. However, SREE's scope includes more ambiguity indicators than the others. Table 1. gives the summary of differences among SREE, ARM, and TIGER Pro.

Features	ARM	TIGER Pro	SREE
Creation of new ambiguity indicator	√	X	√
Updation, Deletion, Saving the previously created indicator	X	X	√ (only in CIC)
Graphical analysis	X	√	X

Requirement's priority setting	X	√	X
Tailorability of defective RStats – display of the preceding and succeeding RStats from the defective RStats	X	X	√
Saving user's rewriting or correction on the defective RStat	X	X	√
Report summary of defective RStats	√	√	√
Input file type	.txt	.txt	.txt and .doc

Table 11. Differences among ARM, TIGER Pro, and SREE

ARM allows its user to add indicators to its indicator corpus. However, ARM does not save the newly added indicators because each time the user loads ARM, it loads only the default original corpus of indicators. In contrast, SREE not only allows its user to add new ambiguity indicators but also allows its user to save, update, and delete the indicators that he or she has added to SREE's CIC.

9.3 Experiments

This section reports the running of test cases on SREE, ARM, and TIGER Pro for the purpose of comparing their effectiveness in finding potential ambiguities. The test cases are two RSs, namely the New Adelaide Airport RS and the MCSS RS, and one test case, called "the random test case", containing a combination of RStats from different domains. Since the author could not gain access to the QuARS, the test cases could not be run on QuARS.

In the description of the results of the experiments, the phrase "Uncaught ambiguity" signifies an ambiguity not caught by the tool at hand.

Detailed definitions are given of "recall", "precision", and "false positive" in Section 8.7. Given an input RS, any token T in any RStat in RS may be potentially ambiguous, that is, T matches an indicator in SREE's AIC. The user of SREE has to determine whether any potentially ambiguous T is truly ambiguous.

Let:

S be an input RS

n be the number of potentially ambiguous tokens in **S** as determined by a human user of SREE based on what SREE is supposed to recognise as potentially ambiguous.

m be the number of potentially ambiguous tokens in S that are determined by the human user of SREE to be truly ambiguous; necessarily **m ≤ n**

p be the number of tokens in **S** that SREE matches with the indicators of its AIC

SREE's Recall for S = **p/n * 100%**

SREE's Precision for S = **m/p * 100% = m/n * 100%**

SREE's False Positives = **p - m**

Note that for any kind of ambiguity for which no potential ambiguity indicator exists in the AIC or for which no indicators have been discovered, SREE's Recall of that ambiguity is 0%.

9.3.1 Random Test Case

The author of this thesis conducted an experiment on a set of 22 randomly selected RStats from different domains. The author determined by manual inspection that 20 of the RStats contain potential ambiguities in SREE's scope. She determined that the remaining 2 RStats contain no ambiguities, and thus are termed *uniguous*. The RStats contain 462 words altogether. This combination of 20 potentially ambiguous RStats and 2 uniguous RStats were presented as inputs to be processed by SREE. The purpose of the experiment was to compute SREE's recall and precision rates for these 22 RStats. The remainder of this section shows each RStat and the calculations of SREE's rates for it. It concludes with the calculations of SREE's overall rates for the entire set.

```
Line No. 1: All users of the system shall login using some form of
unique identification (i.e. user name, password).
is potentially ambiguous (PLURALNOUN) because of wording: users
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (QUANTIFIER) because of wording: some
is potentially ambiguous (VAGUE) because of wording: (i.e. user name,
password)
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that unique is an ambiguous adjective, but some is not ambiguous in the RStat. However, SREE doesn't recognise unique to be potentially ambiguous because unique isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 2: The officer can document/print the reports by selecting an
associate and a supervisor.
is potentially ambiguous (WEAK) because of wording: can
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that reports is an ambiguous plural noun, but and is not ambiguous. However, SREE doesn't recognise reports to be potentially ambiguous because reports isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 3: Restrictions between different types of data access,
either logical or physical, made at LVL2 must be valid if the data
passed on to the online environment are stored and retrieved in the
offline environment.
is potentially ambiguous (OPTIONAL) because of wording: either
is potentially ambiguous (PLURALNOUN) because of wording: types
is potentially ambiguous (VAGUE) because of wording: logical
is potentially ambiguous (VAGUE) because of wording: physical
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 6, n = 6, m = 5.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/6 * 100% = 83%**

SREE's False Positives = **p - m = 6 – 5 = 1**

The user determined that restrictions is an ambiguous plural noun, but types is not ambiguous despite its being plural. However, SREE doesn't recognise restrictions to be potentially ambiguous because restrictions isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 4: A boat can not be entered into the system unless it has a
boat class or sail number.
is potentially ambiguous (WEAK) because of wording: can
```

```
is potentially ambiguous (VAGUE) because of wording: unless
is potentially ambiguous (PRONOUN) because of wording: it
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

```
Line No. 5: A boat can not be entered into a one design race ((or
series)) if its boat class matches the race class.
is potentially ambiguous (PLURALNOUN) because of wording: matches
is potentially ambiguous (PLURALNOUN) because of wording: series
is potentially ambiguous (WEAK) because of wording: can
is potentially ambiguous (VAGUE) because of wording: ((or series))
is potentially ambiguous (PRONOUN) because of wording: its
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that series is not ambiguous because it is not a subject. matches is also not an ambiguous plural noun, but matches is a singular verb. However, SREE recognises matches to be potentially ambiguous because matches is defined as a potential ambiguity indicator in the AIC.

```
Line No. 6: A boat may only be entered into a handicap race (or series)
if its handicap type matches that of the race and its handicap is
within the (inclusive) range specified by the minimum and maximum
handicaps.
is potentially ambiguous (PLURALNOUN) because of wording: handicaps
is potentially ambiguous (PLURALNOUN) because of wording: matches
is potentially ambiguous (PLURALNOUN) because of wording: series
is potentially ambiguous (WEAK) because of wording: may
is potentially ambiguous (VAGUE) because of wording: (or series) if
```

```
its handicap type matches that of the race and its handicap is within
the (inclusive)
is potentially ambiguous (VAGUE) because of wording: maximum
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (VAGUE) because of wording: only
is potentially ambiguous (PRONOUN) because of wording: its
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 11, n = 11, m = 9.

SREE's Recall = **p/n * 100% = 11/11 * 100% = 100%**

SREE's Precision = **m/p * 100% = 9/11 * 100% = 82%**

SREE's False Positives = **p - m = 11 – 9 = 2**

The user determined that series is not ambiguous and matches is not an ambiguous plural noun but matches is a singular verb. However, SREE recognises matches to be potentially ambiguous because matches is defined as a potential ambiguity indicator in the AIC.

```
Line No. 7: The simulation shall use instrument geometry that is
defined and is common to all analysis modules.
is potentially ambiguous (INCOMPLETES) because of wording: is defined
is potentially ambiguous (PLURALNOUN) because of wording: modules
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: common
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 5.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/6 * 100% = 83%**

SREE's False Positives = **p - m = 6 – 5 = 1**

The user determined that modules may not necessarily be potentially ambiguous. However, SREE recognises modules to be potentially ambiguous because modules is defined as a potential ambiguity indicator in the AIC.

```
Line No. 8: Some of the software packages (e.g. each HLT algorithm,
the selection control, the data access) shall be documented for both
the user, developer, and maintainer.
is potentially ambiguous (PLURALNOUN) because of wording: packages
is potentially ambiguous (QUANTIFIER) because of wording: some
is potentially ambiguous (VAGUE) because of wording: (e.g. each hlt
algorithm, the selection control, the data access)
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that and is not ambiguous. However, SREE recognises and to be potentially ambiguous because and is defined as a potential ambiguity indicator in the AIC.

```
Line No. 9: The user shall be able to view the plot dynamically (real-
time) or statically (offline).
is potentially ambiguous (VAGUE) because of wording: (real time) or
statically (offline)
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that dynamically is potentially ambiguous. However, SREE doesn't recognise dynamically to be potentially ambiguous because dynamically isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 10: HLT algorithms should be able to indicate that certain
data preparation steps are to be performed on the requested data.
is potentially ambiguous (PLURALNOUN) because of wording: algorithms
is potentially ambiguous (PLURALNOUN) because of wording: steps
is potentially ambiguous (WEAK) because of wording: should
is potentially ambiguous (PRONOUN) because of wording: that
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that certain is potentially ambiguous, but that is not in the RStat. SREE doesn't recognise certain to be potentially ambiguous because certain isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 11: The interfaces to the databases shall be independent of
the high-level analysis software and provide the Level 1 and Level 2
and higher-level data in TBR (To Be Resolved) formats.
is potentially ambiguous (PLURALNOUN) because of wording: formats
is potentially ambiguous (PLURALNOUN) because of wording: interfaces
is potentially ambiguous (VAGUE) because of wording: (to be resolved)
is potentially ambiguous (VAGUE) because of wording: high level
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 5.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/5 * 100% = 100%**

SREE's False Positives = **p - m = 5 – 5 = 0**

The user determined that TBR is an ambiguous of acronym indicating an incomplete specification, and databases is an ambiguous plural noun. However, SREE doesn't recognise TBR and databases to be potentially ambiguous because TBR and databases aren't defined as potential ambiguity indicators in the AIC.

```
Line No. 12: The user shall have the ability to create a report based
on specific purposes: Observer's Log, Guard Check-in Log, Antenna
Visit Log, and Operator Notes.
is potentially ambiguous (PLURALNOUN) because of wording: notes
is potentially ambiguous (PLURALNOUN) because of wording: purposes
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5– 3 = 2**

The user determined that even though notes is a potential ambiguity indicator in the *Plural* corpus and and is categorised as a potential ambiguity indicator of the *Coordinator* corpus in the AIC, notes and and are not ambiguous in the RStat.

```
Line No. 13: The user interface shall provide information or a warning
indicating what changes will occur when a user changes the regional
setting.
is potentially ambiguous (PLURALNOUN) because of wording: changes
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: what
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that will isn't ambiguous. Even though will is categorised as a potential ambiguity indicator as defined in the *Weak* corpus of AIC, in this case, will doesn't play the role as the main verb that signifies a functional requirement.

```
Line No. 14: The SAS is responsible for routine Level 2 processing of
the LAT data.
is potentially ambiguous (VAGUE) because of wording: routine
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
Line No. 15: The control system should not violate any safety
requirements.
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (WEAK) because of wording: should
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that requirements is an ambiguous plural noun. However, SREE doesn't recognise requirements to be potentially ambiguous because requirements isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 16: The lift should never be allowed to move above the top
floor or below the bottom floor. (There is an emergency shut down
system that will stop the motor if the lift goes above the top floor
or below the bottom floor (by more than 10 cm) but this shut down
system is beyond the scope of the control system.)
is potentially ambiguous (CONTINUANCE) because of wording: below
is potentially ambiguous (WEAK) because of wording: should
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: (there is an
emergency shut down system that will stop the motor if the lift goes
above the top floor or below the bottom floor (by more than 10 cm) but
this shut down system is beyond the scope of the control system.)
is potentially ambiguous (VAGUE) because of wording: but
is potentially ambiguous (VAGUE) because of wording: more
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 9, n = 9, m = 7.
SREE's Recall = **p/n * 100% = 9/9 * 100% = 100%**
SREE's Precision = **m/p * 100% = 7/9 * 100% = 78%**
SREE's False Positives = **p - m = 9 – 7 = 2**

The user determined that will and more are not ambiguous. Although SREE correctly recognises below to be a potential ambiguity, in this case, below should be categorised as an ambiguous adjective instead of as a potential ambiguity indicator of the *Continuance* corpus in the AIC. Also top should be categorised as a potential ambiguity. However, SREE doesn't recognise top to be potentially ambiguous because top isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 17: CL will support both PWR and BWR technologies: fuel
assemblies and casks.
is potentially ambiguous (PLURALNOUN) because of wording: assemblies
is potentially ambiguous (PLURALNOUN) because of wording: technologies
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 6.
SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**
SREE's Precision = **m/p * 100% = 6/6 * 100% = 100%**
SREE's False Positives = **p - m = 6 – 6 = 0**

The user determined that casks is an ambiguous plural noun. However, SREE doesn't recognise casks to be potentially ambiguous because casks isn't defined as a potential ambiguity indicator in the AIC.

```
Line No. 18: The user will experience no practical limit to the number
of fuel assemblies that the database will contain.
is potentially ambiguous (INCOMPLETES) because of wording: no
practical limit
is potentially ambiguous (PLURALNOUN) because of wording: assemblies
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: practical
is potentially ambiguous (PRONOUN) because of wording: that
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that practical is part of the no practical limit phrase and practical isn't ambiguous in the RStat even though practical is categorised as a potential ambiguity indicator as defined in the *Vague* corpus in the AIC.

```
Line No. 19: Additional fields for new information types will need to
be incorporated into future releases of the software, and cannot be
added by users.
is potentially ambiguous (PLURALNOUN) because of wording: fields
is potentially ambiguous (PLURALNOUN) because of wording: releases
is potentially ambiguous (PLURALNOUN) because of wording: types
is potentially ambiguous (PLURALNOUN) because of wording: users
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 6.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/6 * 100% = 100%**

SREE's False Positives = **p - m = 6 – 6 = 0**

adjective. However, SREE doesn't recognise additional and future to be potentially ambiguous because additional and future aren't defined as potential ambiguity indicators in the AIC.

```
Line No. 20: Printing will only be available on screens and windows as
well as predefined reports.
is potentially ambiguous (PLURALNOUN) because of wording: screens
is potentially ambiguous (PLURALNOUN) because of wording: windows
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: as well as
is potentially ambiguous (VAGUE) because of wording: only
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 6.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/6 * 100% = 100%**

SREE's False Positives = **p - m = 6 – 6 = 0**

The user determined that reports is an ambiguous plural noun, and predefined is an ambiguous adjective. However, SREE doesn't recognise reports and predefined to be potentially ambiguous isntances because reports and predefined aren't defined as potential ambiguity indicators in the AIC.

The system shall have the ability to search the maintenance database

An authorised user shall have the ability to create a log entry.

In the 20 RStats that the human analyst determined to contain potential ambiguities in SREE's scope, SREE correctly identifies all of the potential ambiguities and gives for each an appropriate explanation of the type of the potential ambiguity as suggested by the guiding rules. Therefore, SREE has the 100% recall that it was designed with.

Seventeen of the potential ambiguities turned out to be false positives: namely, some (1x), and (3x), types (1x), series (2x), matches (2x), modules (1x), that (1x), notes (1x), will (2x), more (1x), practical (1x), and reports (1x). The false positives result from the confusion of the POS of a potentially ambiguous word. For example, SREE recognises matches as one of its pluralnouns, but in the RStat in which it appears, matches serves as as a singular verb. Since SREE finds matches in the *Plural* corpus, SREE reports matches to be a potentially ambiguous pluralnoun.

Table 12 gives a summary analysis from the Random Test Case. The term "Uncaught instance because no such indicator in the AIC" describes any token that is not caught by SREE as potentially ambiguous because it does not appear in the AIC. In this table, and in many others, "PA" means "potential ambiguities" and "TA" means "true ambiguities".

SREE Compilation Analysis					
Category	**Random Test Case**				
	PA	**Overall % of words**	**Overall % of PA**	**TA**	**False Positive**
Uncaught instance because no such indicator in the AIC	12	2.6	10.81	12	0
Continuance	3	0.65	2.70	2	1
Coordinator	15	3.25	13.51	12	3
Directive	0	0	0	0	0
Incompletes	2	0.44	1.80	2	0
Optional	1	0.22	0.9	1	0
Plural	25	5.41	22.52	18	7
Pronoun	10	2.16	9	9	1
Quantifier	5	1.08	4.5	4	1

Vague	25	5.41	22.52	23	2
Weak	13	2.81	11.71	11	2
Total	**111**			**94**	**17**

Table 12. Analysis on Random Test Case by SREE

As mentioned, SREE's overall recall is 100%. In addition, SREE's overall precision is 84.69%, which is the percentage of potential ambiguities that SREE finds that are truly ambiguous.

9.3.2 Tests of the New Adelaide Airport and MCSS RSs

The New Adelaide Airport RS contains 1036 words and 49 RStats. The MCSS RS contains 4673 words and 246 RStats. The MCSS RS contains a system description and RStats for 9 subsystems. The subsystems are (1) Switch Matrix Subsystem, (2) Switch Control Subsystem, (3) Timing Generator Subsystem, (4) Test and Monitoring Subsystem, (5) Growth, Reliability, Maintainability, and Availability, (6) Safety, (7) Facilities, (8) Technical Support, and (9) Training. The RStats in the MCSS RS include not only functional requirements but also non-functional requirements.

To evaluate the performance of ARM, TIGER, and SREE, each of the tools is to detect the same ambiguity indicators, which are and, or, all, and any. The author downloaded ARM, and added the indicators or and and/or to ARM's continuance corpus and the indicator all and any to ARM's weak corpus. What ARM's analysis found is shown in Table 13:

Indicator	Keyword Detected on New Adelaide Airport RS	Keyword Detected on MCSS RS
Imperative	shall (51x)	shall (234x), will (18x)
Continuance	or (39x), and (25x), : (5x)	or (195x), and (77x), and/or (2x), : (2x)
Directive	-	e.g. (1x), i.e. (1x), Figure (2x)
Option Phrases	-	may (1x)
Weak Phrase	all (1x), adequate (2x)	all (10x), any (26x), be able to (10x), be capable of (11x), capability to (4x)
Incomplete	-	as a minimum (1x)

Table 13. ARM's analyses New Adelaide Airport RS and MCSS RS

There is no specification in ARM's analysis report of how it found 39 occurrences of or and 25 occurrences of and. The computed value is not what the author expected and unfortunately ARM does not indicate how ARM computed the value. From SREE's analysis of the same RS, we know that there are 1 or and 23 ands.

TIGER's analysis finds 37 defective RStats in the New Adelaide Airport RS with:

- 5 multiple requirements in a line (defect type-1) from the occurrence of multiple shall in a RStat
- 28 possible multiple requirements with the use of coordinator and (defect type-2)
- 3 unverifiable words such as uses of keyword such as adequate and all (defect type-3)

A summary of the results obtained are as follows:

- *TIGER's Analyses of the New Adelaide Airport and MCSS RSs*

	New Adelaide Airport	MCSS
Records	83	246

Requirements	49	246
Ignore Poor Words	2	0
Number of Defective RStats	37	165
Defect type 1 - Multiple requirements in a line	5	4
Defect type 2 - Possible multiple requirements	28	93
Defect type 3 - Not verifiable word	3	45
Defect type 4 - Use of wrong word	0	23
Defect type 5 - User defined poor word	0	0

Table 14. Analyses New Adelaide Airport RS and MCSS RS by TIGER

TIGER’s type-1 defect is not inside SREE’s scope. The 28 type-2 defects in TIGER’s analysis include multiple occurrences of and in some RStats. If we do not count any multiple occurrences of and in one RStat, then TIGER detects 15 type-2 defects from the use of and (14x) and or (1x).

Tools	Indicator and	Indicator or	Indicator all
TIGER	14	1	1
ARM	25	39	1
SREE	23	1	3

Table 15. Analyses SREE, ARM, and TIGER New Adelaide Airport RS

In the MCSS RS, TIGER’s analysis finds165 defective RStats with:

- 4 multiple requirements in a line (defect type-1)
- 93 possible multiple requirements (defect type-2)
- 45 unverifiable words (defect type-3)

- 23 use of wrong word (defect type-4)

Tools	Indicator and	Indicator or	Indicator and/or	Indicator all	Indicator any
TIGER	61	15	0	11	24
ARM	77	195	2	10	26
SREE	78	14	2	19	28

Table 16. Analyses SREE, ARM, and TIGER MCSS RS

Comparing to TIGER's finding 93 type-2 defects from the uses of and and or, SREE finds 94 indicators of *Coordinator* with and (78x), or (14x), and and/or (2x). TIGER counts every and and or inside a RS, including multiple occurrences in a RStat. If we do not count multiple occurrences of and or or in one RStat, then TIGER finds 75 type-2 defects with and (61x) and or (14x). In addition, TIGER counts also 45 type-3 defects from all and any. If we do not count multiple occurrences of all or any in one RStat, then TIGER finds 40 type-3 defects with all (11x), any (24x), include (2x), appropriate (1x), sufficient (1x), and such as (1x). SREE finds 48 occurences of Quantifier with any (28x), all (19x), and much (1x) in the MCSS RS. Moreover, SREE treats multiple occurrences of and in a Rstat as one occurrence and reports that occurrence a potential ambiguity. SREE finds 63 Rstats out of 132 lines in the New Adelaide Airport RS as potentially ambiguous. The author has determined that these 63 Rstats are truly ambiguous Rstats, and that each of these 63 Rstats contains at least one ambiguity indicator from the AIC.

- *SREE's Analyses of the New Adelaide Airport and MCSS RSs*

SREE Compilation Analysis					
Category	**New Adelaide Airport RS**				
	PA	**Overall % of words**	**Overall % of PA**	**TA**	**False Positive**
Uncaught instance due to no such indicator in the AIC	21	2.04	11.79	21	0
Continuance	15	1.45	8.43	12	3
Coordinator	24	2.32	13.48	21	5
Directive	0	0	0	0	0
Incompletes	1	0.09	0.56	1	0
Optional	0	0	0	0	0
Plural	74	7.16	41.57	34	40
Pronoun	5	0.48	2.80	4	1
Quantifier	3	0.29	1.69	3	0
Vague	35	3.39	19.66	26	9
Weak	0	0	0	0	0
Total	**178**			**122**	**58**

Table 17. Analyses New Adelaide Airport RS by SREE

SREE Compilation Analysis					
Category	**MCSS RS**				
	PA	**Overall % of**	**Overall % of**	**TA**	**False Positive**

		words	PA		
Uncaught instance due to no such indicator in the AIC	42	1.33	5.81	42	0
Continuance	12	0.26	1.66	12	0
Coordinator	94	2.01	13	78	16
Directive	2	0.04	0.28	2	0
Incompletes	2	0.04	0.28	2	0
Optional	1	0.02	0.14	1	0
Plural	265	5.67	36.65	70	195
Pronoun	115	2.46	15.91	107	8
Quantifier	48	1.03	6.64	48	0
Vague	127	2.72	17.57	101	26
Weak	15	0.32	2.06	13	2
Total	**723**			**476**	**247**

Table 18. Analyses MCSS RS by SREE

The analysis of the New Adelaide Airport RS reports that there are 21 uncaught potential ambiguities because no indicators for them exist in SREE's AIC. These are peak (1x), major (1x), recently (1x), convenient (1x), domestic (3x), international (3x), aesthetics (1x), aesthetic (1x), specified (4x), loadings (2x), and world-class (3x). An indicator for each of these uncaught potential ambiguities should be added to the AIC so that SREE is able to identify these potential ambiguities in the future.

Nevertheless, SREE’s overall recall of potential ambiguities in SREE’s scope is 100%, and SREE’s overall precision is 67.78%.

SREE’s analysis of the MCSS RS leaves 42 potential ambiguities uncaught because no indicator for any of them is in the AIC. These are requirements (3x), subsystems (2x), full (1x), outputs (4x), definable (1x), default (2x), similar (2x), mnemonics (1x), diagnostics (1x), available (1x), predetermined (1x) predefined (1x),, electronics (1x), specified (1x), unique (2x), other (2x), identical (3x), valid (1x), in such a manner (1x), future (1x), additional (2x), uniquely (2x), custom (3x), plans (1x), appropriate (1x), and characteristics (1x). Once the user adds these indicators to SREE’s AIC, they will be caught and tagged as potential ambiguities in the future. SREE’s overall recall of potential ambiguities in SREE’s scope is 100%, and SREE’s overall precision is 65.84%. SREE reads 260 Rstats out of 471 lines in the MCSS RS as potentially ambiguous. The author has determined that these 260 potentially ambiguous Rstats are truly ambiguous, and each of the these 260 Rstats contains at least one ambiguity indicator from the AIC.

SREE’s analysis of the New Adelaide Airport RS and the MCSS RS show first many potential ambiguities from the *Plural* corpus and then many potential ambiguities from the *Vagueness* corpus. As mentioned elsewhere in this thesis, since SREE recognises only the potential ambiguities matching indicators in its AIC, the analysis is based on only the contents of the AIC. Hence, whenever a new indicator is added to the AIC, a new analysis has to be done.

The most disconcerting issue of these analyses is the number of false positives among the potential ambiguities from the *Plural* corpus in the AIC. As observed in

Tables 15 and 16, there were 195 false positives among the 265 potential ambiguities classified as from the *Plural* corpus in the MCSS RS, and 40 false positives among the 74 potential ambiguities classified as from the *Plural* corpus in the New Adelaide RS. The high numbers of false positives among the potential ambiguities from the *Plural* corpus raises concern whether the indicators of the *Plural* corpus should even be in the scope of SREE. The indicators in the *Plural* corpus are only a subset of the plural nouns. SREE's lexical analyser is unable to determine whether any token matching an indicator in the *Plural* corpus is indeed a plural noun. Therefore, SREE's precision may be lower than hoped, which is not too much less than 100%. SREE's human user is expected to determine whether any potential ambiguity is truly ambiguous. Perhaps, the use of a POS tagger or a parser can help increase precision. Normally, the accuracy of a POS tagger is less than 100%. Usually the difficulty is that the tagger identifies more than one possible POS for a token in its context. Perhaps when a tagger gives a unique POS to the token matching an indicator in the *Plural* corpus and that POS is not "noun", it is safe for SREE not to report the token as a potentially ambiguous plural noun. Experiments need to be done to test this idea.

9.4 Analysis Human Analysts

One programmer, one database administrator, and one Ph.D. candicate majoring in Knowledge Representation in RE (and therefore, not the author of this thesis), were asked to study and analyse the New Adelaide Airport RS and the MCSS RS. In the programmer's point of view, the New Adelaide RS looks like an airport operational specification rather than software system specification. There are a number of imprecise and incomplete RStats. Without domain knowledge and a complete reference tailored to the RS, it was difficult to understand the functional features in the system. Table 19 shows a summary of the potential ambiguities from the three analyses of the New Adelaide Airport RS. Table 20 shows a similar summary of the three analyses of the MCSS RS.

Ambiguity Type	**Instance**
Vague	major, adequate, specified, aesthetic, modular, world-class, peak, consistent, adequate, aesthetic, like, Land-Side, Banking, processing delay, Air-Side
Coordination	and
Continuance:	following
Directive	figure, appendix
Plural	guests, facilities
Acronym	GPO
Long RStats	1.2.4, 2.2.1
Imprecise RStats	1.2.3, 1.2.3.1, 1.2.5, 1.3.1, 2.3.1.1

Table 19. Summary of Ambiguity in New Adelaide Airport RS

Ambiguity Type	**Instance**
Vague	designated, specific, particular, by means of password protection scheme, redundant, sufficient, identical, predetermined, capability, manner, changeover, powered-down, master, of the kind, interrack, contiguous

Coordination	and, and/or
Pronoun	this
Plural	priviledges, controllers, modifications, changes, manuals
Acronym	LCT, NBG, BED
Quantifier	any, all
Under-reference	user, people
Long RStats	5.3.61, 5.3.63, 5.9.26
Imprecise RStats	5.10.4

Table 20. Summary of Ambiguity in MCSS RS

The analysis classifies an RStat as a Long RStat when the RStat appears to contain more than one requirement because it contains the coordinator and. An RStat is said to be imprecise when it is difficult to interpret the true intent of the requirement because the requirement doesn't describe precise information. SREE's *Vague* and *Plural* corpi do not contain some of the tokens that are called vague type or plural in Tables 15 and 16. Hence, in order for SREE to be able to recognise these tokens as potential ambiguities, they will have to be added to SREE's CIC.

To summarise, the discussion in Section 9.2 not only has explained how SREE processes an input RS and reports the detected potential ambiguities in that RS, but also has described the weaknesses of SREE. These weaknesses of SREE are the biggest concerns for future resolution:

- the fact that SREE has 0% recall for any potential ambiguity *not in* its scope even though SREE has 100% recall for the potential ambiguities *in* its scope. The user has to search for these potential ambiguities manually, but at least, the user *knows* what potential ambiguities he or she has to search for manually.

- the fact that SREE's scope includes only pluralnouns and not plural nouns. As explained in Section 8.3, pluralnouns are those tokens recognised as potentially ambiguous by SREE as a result of the *Plural* corpus, whereas plural nouns is the actual set of nouns that are plural. SREE has 100% recall of pluralnouns, but not of plural nouns. It may be better to report any read word that ends in s and define in the AIC a list of exceptional plural nouns that do not end in s. A complete enough list of exceptional plural nouns that do not end in s is probably smaller than the current list of 11,287 pluralnouns and is probably easier to make complete enough than a list of all plural nouns. Alternatively, the user is responsible to keep adding more plural nouns to the list of pluralnouns in the AIC until pluralnouns have converged all plural nouns, which is not feasible due to the richness of NL.
- the fact that the AIC is not complete, because the uses of SREE in the experiments have resulted in many suggestion of indicators to add to the AIC. However, it is not possible to have a complete AIC due to the richness of NL, as any newly defined word can be potentially ambiguous in any domain.
- the fact that the SREE's measured precision rate is lower than the goal of not too much less than 100%, particularly because of singular verbs that SREE recognises as pluralnouns and because SREE recognises plural nouns that are not in the subject position to be pluralnouns. Despite the low precision, the author believes that the use of SREE with its 100% recall of potential ambiguities in its scope is better for the user than to have to have to find these potential ambiguities manually in close

readings of the RS. However, if too many plural nouns are outside SREE's scope, then perhaps not even the 11,287 plural nouns in the *Plural* corpus should be in SREE's scope.

Nevertheless, SREE does help its user to find instances of the indicators in its AIC. The use of any requirements analysis tool, be it SREE, QuARS, or any other, does help to reduce the ambiguity of a RS. Furthermore, the tool helps detect some costly RS errors early, when it is cheaper to fix them.

10 Conclusion

The most significant benefit of this research is the insight gained about finding potential and actual ambiguities in NLRSs. This research work has derived a set of ambiguity avoiding guiding rules, and an experimental tool, SREE, to automate the detection of potential ambiguities in RSs. The ambiguity of a NLRS in NL can be reduced through the use of the guiding rules to find what should be avoided and to know what to write instead.

SREE is adaptable to different application domains and different kinds of NLRS. It accepts as input a text file, with a .txt extension, or a Word file, with a .doc extension. SREE searches its input for instances of indicators that are in its AIC, and it reports each of these as a potential ambiguity to be checked by its user for actual ambiguity. SREE has 100% recall by design for all potential ambiguities inside its scope, namely those which have indicators in its AIC. SREE's precision is less than 100%, but a human user can easily determine which of the reported potential ambiguities are false positives. In any case, it is easier for a human user to decide which of the reported list of potential ambiguities are false positives than to search for ambiguities in the full input document, especially since the list of potential ambiguities is shorter than the full input document.

The analysis of the SREE runs on the New Adelaide Airport RS shows that there were 58 false positives among the 180 potential ambiguities that SREE found in the RS, for a precision of 67.78%. The analysis of the SREE runs on the MCSS RS shows that there were 247 false positives among the 723 potential ambiguities that SREE found in the RS, for a precision of 65.84%.

The lack of uniformity and the hit-and-miss nature of the guiding rules are a bit disconcerting. However, these guiding rules cover the kinds of ambiguities that the author found in actual industrial RSs. Of course, the method by which the guiding rules were found makes it difficult to assess when enough rules have been found. Probably, there is no limit on the number of rules. However, we expect that at some point, the rate of addition of new rules will drop considerably, just because we will eventually begin not to find new kinds of ambiguities. Thus, this thesis work is complementary to all other research work cited in Chapter 4 that attempts to find systematic ways of detecting or avoiding ambiguities.

Another disconcerting property of the guiding rules is the difficulty of finding a pattern for each ambiguity. For any rule, there is no guarantee that every RStat meeting the pattern of the rule is an instance of the kind of ambiguity that is intended to be described by the rule. Conversely, there is no guarantee that the rule describes every instance of the kind of ambiguity that is intended to be described by the rule.

Probably the most valuable use of the guiding rules and of SREE is to help inspect a RS. The guiding rules and SREE identify questions that should be asked of the client of the RS. Any time a RStat is determined to be ambiguous, the requirements analyst must ask the client what she means by the ambiguous RStat. The nature of the ambiguity found in the RStat shows the questions that should be asked. A suggested resolution given with any guiding rule is only a suggestion; it may even be wrong! Only the client of the RS can say what an ambiguous RStat really means.

11 Future Direction

SREE, the experimental tool, has been used to analyse RSs from real industrial domains. The outcomes of these trials are promising because SREE proved to be effective in recognising potential ambiguities in its scope. The trials showed several weaknesses of SREE and opportunities for improvement, particularly in the guiding rules and in the AIC. The AIC certainly needs to be enhanced with more indicators of potential ambiguity, particularly in the *Vague* corpus, in the *Incomplete* corpus, and in the *Plural* corpus. Adding more indicators to the AIC helps to broaden SREE's scope, eventually leading to a reduction in the finding of uncaught ambiguities.

The high number of false positives among the potential ambiguities matching indicators in the *Plural* corpus raises concerns about the usefulness of the chosen method to deal with the plural noun ambiguity. Perhaps a larger list of plural nouns, including irregulars is needed in the AIC.

As discussed in Chapter 9, it may be better to have SREE's lexical analyser recognise

1. all words ending in s, es, ae, aux, and other common plural noun endings, and
2. all words in a new *Plural* corpus consisting of as many irregular plural nouns as possible

as potential plural nouns, causing higher imprecision. The issue is which is worse, to have to manually search for plural nouns not currently in SREE's scope or to have more imprecision?

Perhaps, the imprecision of the potential plural nouns found by the current or a modified SREE can be considerably reduced, as suggested in Chapter 9, by techniques with less than 100% recall but applied on the other side to reduce imprecision. For example, it might be safe not to report a lexically recognised potential plural noun if a parser together with a POS tagger concluded with certainty that the potential noun is a verb. It might be that if a parser together with a POS tagger gives only one possible POS tag to a word, that tagging can be regarded as certain. Only experimentation can validate the usefulness of this idea.

The author expects also to continue to examine more industrial RSs to find more guiding rules. So long there is a new software development, there will always be new RSs that introduce new kinds of ambiguities in the vocabulary used. Whenever there is a new kind of ambiguity, a new guiding rule should be written to deal with the new kind of ambiguity. The author believes that eventually, the rate of addition of new guiding rules will drop considerably, just because we will eventually begin not to find new kinds ambiguities, except in vocabulary.

SREE is still an experimental tool and more work needs to be done to turn the experimental tool into a production tool. The search strategy can certainly be optimised. Some of the ideas suggested above can be tried. It will be necessary also to test SREE on many more RSs. SREE needs to be tested in new product developments so that effects on the guiding rules can be monitored. Another possible improvement to SREE is to cluster the vocabulary; the clusters can be used to examine RStats for ommissions and conflicts [Baker and McCallum, 1998; Bellegarda et al., 1996].

Bibliography

[Adams and Kaye, 2006]

K.A. Adams and A.S. Kaye, Revisiting The Ambiguity of "And" and "Or" in Legal Drafting, 2006, St. John's Law Review 1167

[Ambriola and Gervasi, 2003]

V. Ambriola and V. Gervasi, The CIRCE approach to the systematic analysis of NL requirements, Technical Report: TR-03-05, 2003, Università Di Pisa

[Bach, 1998]

K. Bach, Ambiguity, 1998, In E. Craig and L. Floridi editors, Routledge Encyclopedia of Philosophy, London, UK, Routledge

[Baker and McCallum, 1998]

L.D. Baker, A.K. McCallum, "Distributional clustering of words for text classification", Proceedings of the 21st annual international ACM SIGIR, Melbourne, Australia, 1998, pp 96-103

[Barr, 1999]

V. Barr, Identifikation von Spezifikationsmustern im Echtzeitentwurf anhand der Fallstudie Antiblockiersystem, 1999, Diplomarbeit der Universitaet Oldenburg, Fachbereich Informatik

[Bellegarda et al., 1996]

J.R. Bellegarda, J.W. Butzberger, C. Yen-Lu, N.B. Coccaro, D. Naik, "A novel word clustering algorithm based on latent semantic analysis", In *IEEE Proceedings of the Acoustics, Speech, and Signal Processing*, 1996, vol 01, pp 172-1785

[Bennett, 2005]

B. Bennett, Modes of Concept Definition and Varieties of Vagueness, March 2005, Applied Ontology, 1(1), pp 17-26

[Berry and Kamsties, 2000]

Berry, D.M. and Kamsties, E. The Dangerous “All” in Specifications, Technical Report, July, 2000, University of Waterloo, Waterloo, ON, Canada

[Berry et. al., 2003]

D.M. Berry, E. Kamsties and M.M. Krieger, From Contract Drafting to Software Specification: Linguistic Sources of Ambiguity, A Handbook Version 1.0, University of Waterloo, Waterloo, ON, Canada

[Bradner, 1997]

S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, BCP 14, March 1997, Harvard University

[Brill and Mooney, 1997]

E. Brill and R. J. Mooney, An Overview of Empirical Natural Language Processing, 1997, AI Magazine 18(4): 13-24

[Bucchiarone et.al., 2005]

Bucchiarone, S. Gnesi, and P. Pierini, Quality analysis of NL requirements: An industrial case study, 2005, In Proceedings of the Thirteenth IEEE International Conference on Requirements Engineering (RE’05), pp. 390–394, 29 August–2 September 2005

[Carter and McCarthy, 2006]

R. Carter and M. McCarthy, Cambridge Grammar of English A Comprehensive Guide Spoken and Written English Grammar and Usage, © Cambridge University Press 2006

[Chandler, 1982]

M. Chandler. The logic of "unless", 1982, *Philosophical Studies*, 41(3):383–405, May 1982

[Chantree, 2005]

F. Chantree, B. Nuseibeh, A.D. Roeck, A. Willis, March 2005, "Nocuous Ambiguities in Requirements Specifications" Technical Report, Department of Computing, Faculty of Mathematics and Computing The Open University

[Chantree et al., 2006]

F. Chantree, A. Willis, A. Kilgarriff & A.D. Roeck, Detecting Dangerous Coordination Ambiguities Using Word Distribution, 2006, Recent Advances in Natural Language Processing, Springer Verlag

[Cohen, 2004]

S.M. Cohen, Introduction to Logic, Chapter 3: The Boolean Connectives, Copyright © 2004, University of Washington

http://faculty.washington.edu/smcohen/120/Chapter3.pdf

[Crystal, 2003]

D. Crystal, A Dictionary of Linguistics & Phonetics, Fifth Edition, 2003, Blackwell Publishing

[Denger et al., 2001]

C. Denger, D. Jőrg, E. Kamsties, QUASAR A Survey on Approaches for Writing Precise Natural Language Requirements, 2001, Fraunhofer IESE

[Denger, 2002]

C. Denger. High quality requirements specifications for embedded systems through authoring rules and language patterns. Technical Report M. Sc. Thesis, 2002, Fachbereich Informatik, Universit¨at Kaiserslautern

[Denger et al., 2003]

C. Denger, D.M. Berry, E. Kamsties, Higher Quality Requirements Specifications through Natural Language Patterns, 2003, Proceedings of the IEEE International Conference on Software – Science, Technology & Engineering (SwSTE'03)

[Ebenau and Strauss, 1994]

R.G. Ebenau and S.H. Strauss, Software Inspection Process, 1994, McGraw Hill, New York, NY

[Fabbrini et al., 2000]

F. Fabbrini, M. Fusani, G. Gnesi and G. Lami, Quality Evolution of Software Requirements Specifications, 2000, Proceedings Software and Internet Quality Week 2000 Conference, San Fransisco, CA

[Fabbrini et. al., 2002]

F. Fabbrini, M. Fusani, G. Gnesi and G. Lami, The Linguistic Approach to the Natural Language Requirements Quality: Benefit of the use of an Automatic Tool, 2002, Proceedings of the 26th Annual NASA Goddard Software Engineering Workshop (SEW'01)

[Fillmore, 1968]

C.J. Fillmore, The Case for Case, 1968, in Universals in Linguistic Theory

[Firesmith, 2003]

D. Firesmith, Specifying Good Requirements, 2003, Journal of Object Technology, 2(4):53–64, July-August 2003

[Fliedl et al., 2000]

G. Fliedl, C. Kop, W. Mayerthaler, H.C. Mayr, C. Winkler, Linguistic Aspects of Dynamics in Requirements Specifications & Linguistically Based Requirements Engineering, 2000, Data & Knowledge Engineering, Netherlands

[Fuchs and Schwitter, 1995]

N. E. Fuchs and R. Schwitter, Specifying logic programs in controlled natural language, 1995, In CLNLP'95, Workshop on Computational Logic for Natural language Processing

[Fuchs and Schwitter, 1996]

N.E. Fuchs and R. Schwitter, Controlled English for Requirements Specification, 1996, IEEE Computer Special Issue on Interactive Natural Language Processing

[Goro et al., 2004]

T. Goro, U. Minai, and S. Crain, Two Disjunctions for the Price of *Only* One, 2004, BUCLD 29: Proceedings of 29th annual Boston University Conference on Language Development

[Götz and Rupp, 1999]

R. Götz and C. Rupp, Regelwerk Natürlichsprachliche Methode, 1999, Sophist, Nürnberg, Germany

[Goetz R. and Rupp C., 2003]

R. Goetz and C. Rupp, Psychotherapy for System Requirements, 2003, Proceedings of the Second IEEE International Conference on Cognitive Informatics (ICCI'03), © 2003 IEEE

[Gause, 2000]

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

[Gilb and Graham, 1993]

T. Gilb and D. Graham, Software Inspection, 1993, Addison Wesley Wokingham, UK

[Hartmann, 2006]

D. Hartmann, Interview: Jim Johnson of the Standish Group, Delivering Value, Methodologies, 2006

[Hooks, 1994]

I. Hooks, Writing Good Requirements, Vol. 2, pp. 197-203, 1994, Proceedings of the Fourth International Symposium of the NCOSE 2, San Jose, CA

[IEEE, 1998]

IEEE Computer Society, IEEE Std. 830-1998, IEEE recommended practice for software requirements specification, 1998, Software Engineering Standards Committee of the IEEE Computer Society, USA

http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel4/5841/15571/00720574.pdf

[Jaszczolt, 2002]

K.M. Jaszczolt, Against Ambiguity and Underspecification: Evidence from Presupposition as Anaphora, Journal of Pragmatics 34, 829-849

[Juristo et al., 1997]

N. Juristo, A.M. Moreno and M. Lopez, How to Use Linguistic Instruments for OOA, 2000, *IEEE Software*, 17(3):80–89, May/Jun 1997

[Kamsties et. al., 2001]

E. Kamsties, D.M. Berry, and B. Paech, Detecting Ambiguities in Requirements Documents Using Inspections, Technical Report, June 2001University of Waterloo, Waterloo, ON, Canada

[Kasser, 2004]

J.E. Kasser, Tool to InGest and Elucidate Requirements PROfessional (TIGER Pro), 2004, Division of Information Technology Engineering and the Environment, Systems Engineering & Evaluation Centre, UniSA

http://www.seecforum.unisa.edu.au/SEECTools.html#tiger

[Kazmierczak, 2003]

E. Kazmierczak, Requirements Engineering, Course-work Masters, First Semester 2003, University of Melbourne

[Lami, 2005]

G. Lami, QuARS: A Tool for Analysing Requirements, September 2005, Carnegie Mellon University

[Lamsweerde, 2000a]

Lamsweerde A.V., *Building Formal Requirements Models for Reliable Software*, 2000, In The Future of Software Engineering, A. Finkelstein (ed.), ACM Press

[Lamsweerde, 2000b]

A.V. Lamsweerde, Requirements Engineering in the Year 00: A Research Perspective 2000, Keynote paper, 2000, Proc. ICSE 2000 - 22nd Intl. Conference on Software Engineering, IEEE Press

[Lamsweerde, 2000c]

A.V. Lamsweerde, Formal Specification: A Roadmap, 2000, Proc. ICSE 2000 - 22nd Intl. Conference on Software Engineering, Future of SE Track 147-159

[Lanman, 2002]

J. T. Lanman, Using Formal Methods in Requirements Engineering Version 1.0, 2002, Department of Computing and Mathematics, Embry-Riddle Aeronautical University

[Lappin and Leass, 1994]

S. Lappin and H.J. Leass, An algorithm for pronominal anaphora resolution, 1994, Computational Linguistics, 20(4), 535-561

[Lawler, 2005]

J. Lawler. Post subject: Re: difference between “each” and “every”, Sun October 23 2005. http://www.vocaboly.com/forums/ftopic9080.html.

[Luger, 2002]

G.F. Luger, Artificial Intelligence Structures and Strategies for Complex Problem Solving, 2002,© Pearson Education Limited

[Macias and Pulman, 1993]

B. Macias and S. Pulman. Natural language processing for requirements specifications, 1993, In F. Redmill and T. Anderson editors, Safety-critical systems: Current issues, techniques and standards, pp. 67–89, London, UK, Chapman & Hall

[Macaulay, 1996]

Macaulay L.A., *Requirements Engineering – Applied Computing*, 1996, © Springer – Verlag London Limited

[Martinez et al., 2004]

Martinez A., Pastor O., Estrada H., A pattern language to join early and late requirements, 2004, pp 51-64 Anais do WER04 - Workshop em Engenharia de Requisitos, Tandil, Argentina

[Meyer, 1985]

B. Meyer, On Formalism in Specifications, January 1985, IEEE Software Vol.2 No.1 6-26

[Miller et al., 2006]

G.A. Miller, C. Fellbaum, R. Tengi, P. Wakefield, R. Poddar, H. Langone, B. Haskell, WordNet 3.0 A lexical database for the English language, 2006, Cognitive Science Laboratory, Princeton University

[Noveck et. al., 2002]

I. A. Noveck, G. Chierchia, F. Chevaux, R. Guelminger, E. Sylvestre, Linguistic-pragmatic factors in interpreting disjunctions, 2002, Volume 8, Number 4, © Psychology Press Ltd

[Nuseibeh and Easterbrook, 2000]

B. Nuseibeh and S. Easterbrook, *Requirements Engineering: A Roadmap*, 2000, Proceedings of International Conference on Software Engineering (ICSE-2000), Limerick, Ireland, ACM Press.

[Nuseibeh and Easterbrook, 2001]

B. Nuseibeh, S. Easterbrook and Russo A., *Making Inconsistency Respectable in Software Development, Journal of Systems and Software*, 58(2):171-180, September 2001, Elsevier Science Publishers.

[Ohnishi, 1994]

Ohnishi A., *Customizable Software Requirements Languages*, 1994, Proceedings of the Eighth International Computer Software and Application Conference (COMPSAC), IEEE Computer Society, Los Alamitos, CA

[Robertson and Robertson, 1999]

S. Robertson and J. Robertson, Mastering the Requirements Process,1999, Addison-Wesley Professional,

[Rolland and Proix, 1992]

Rolland C. and Proix C., *A Natural Language Approach for Requirements Engineering*, 1992, pp. 257-277 in Proceedings of Conference on Advanced Informations Systems Engineering, CAiSE, Manchester UK

[Ruan et al., 2001]

D. Ruan, J. Kacprzyk, and M. Fedrizzi, Soft Computing for Risk Evaluation and Management: Application in Technology, Environment and Finance, 2001, © Physica-Verlag Heidelberg, Germany

[Rupp and Goetz]

C. Rupp and R. Goetz, Linguistic methods of requirements engineering (NLP), November 2000, In Proceedings of the European Software Process Improvement Conference (EuroSPI)

http://www.iscn.com/publications/#eurospi2000

[Russell, 1923]

B. Russell, Vagueness, 1923, Collected Papers, vol. 9, pp. 147-154, The Australasian Journal of Psychology and Philosophy: 84-92

http://cscs.umich.edu/~crshalizi/Russell/vagueness/

[Ryan, 1993]

K. Ryan, The Role of Natural Language in Requirements Engineering, 1993, Proceedings of the IEEE Int. Symposium on RE, pp. 240-242, San Diego California 4-6 Jan 1993

[Sawyer and Kotonya, 2001]

P. Sawyer and G. Kotonya, Chapter 2 Software Requirements, Guide to the Software Engineering Body of Knowledge Trial Version SWEBOK, A Project of the Software Engineering Coordinating Committee, 2001, © IEEE – Trial Version 1.00

[Schneider et. al., 1992]

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

[Schwertel, 2000]

U. Schwertel, Controlling plural ambiguities in Attempto Controlled English, 2000, In Proceedings of the Third Internaltional Workshop on Controlled Language Applications (CLAW), Seattle, WA, USA

[Standish, 2001]

Standish Group, 2001, Extreme Chaos

http://www.quarrygroup.com/wp-content/uploads/art-standishgroup-CHAOS0report.pdf

[Standish, 2003]

Standish Group, March 2003, Latest Standish Group CHAOS Report Shows Project Success Rates Have Improved by 50%, 2003, © The Standish Group International Inc.

http://www.standishgroup.com/press/article.php?id=2

[DOORS, 2008]

Telelogic DOORS®, Requirements Management for Advanced Systems and Software Development, © Copyright Telelogic AB 2008

http://www.telelogic.com/products/doors/doors/index.cfm

[Tjong, 2006a]

S. F. Tjong, Improving the quality of natural language requirements specifications through natural language requirements patterns, March 2006, Technical report, Faculty of Engineering and Computer Sciences, University of Nottingham Malaysia Campus

http://sepang.nottingham.edu.my/˜kcx4sfj/

[Tjong, 2006b]

S. F. Tjong, Elaborated natural language patterns for requirements specifications, August 2006, Technical report, Faculty of Engineering and Computer Sciences, University of Nottingham Malaysia Campus http://sepang.nottingham.edu.my/~kcx4sfj/

[Tjong, 2006c]

S. F. Tjong, Improving the quality of natural language requirements specifications through natural language requirements patterns, September 2006, In IEEE International Conference on Computer and Information Technology

[Tunstall, 1998]

S. Tunstall, The Interpretation of Quantifiers: Semantics and Processing. Thesis (Ph.D.), University of Massachusetts, Amherst

[Virrantaus, 2003]

K. Virrantaus, Analysis of the uncertainty and imprecision of the source data sets for a military terrain analysis application, 2003, Proceedings of 2nd International Symposium on Spatial Data Quality, Hong Kong

[Watt and Brown, 2000]

D. A. Watt and D. F. Brown, Programming Language Processors in Java, Compilers and Interpreters, 2000, © Pearson Education Limited

[Wilson et.al., 1997]

W. M. Wilson, L. H. Rosenberg, and L. E. Hyatt. Automated analysis of requirement specifications, 1997, In Proceedings of the Nineteenth International Conferences on Software Engineering (ICSE-97) pp. 161–171, New York, NY, USA, 17–23 May 1997, ACM Press

[Zowghi, 2007]

D. Zowghi, RE@UTS – Requirements Engineering Activities, Requirements Engineering @UTS Home Page, last accessed August 2007

http://research.it.uts.edu.au/re/re_uts_activities.html

Requirements Documents

[Bray, 2002]

Bray, I.K., An Introduction To Requirements Engineering, © Pearson Education Limited, 2002

- The yacht racing results (YRR) case study, p. 337-340
- The lift controller case study, p. 357-358

[BPS, 2005]

C. S. Eng, Batch poster system, Detailed Business Requirements. Technical report, EDS MySC, Malaysia, 2005

[CLS, 1999]

Cask Loader Software, General Requirements Document Draft, © Electric Power Research Institute, Inc., 3 Sept 1999

http://www.epri.com/eprisoftware/processguide/docs/srdexdoc.doc.

[DCS, 2002]

F.J. Nelbach, Software Requirements Document For the Data Cycle System (DCS) Of The SOFIA Project, Universities Space Research Association, 2002

http://www.astro.ucla.edu/~shuping/SOFIA/Documents/DCS_SRD_Rev1.pdf

[EVLA, 2003]

R. Moeser and P. Perley, Expanded Very Large Array (EVLA) operations interface, software requirements. Technical report, EVLA-SW-003 Revision: 2.5, 2003
http://www.aoc.nrao.edu/evla/techdocs/computer/workdocs/array-sw-rqmts.%pdf

[LAT, 2002]

R. Dubois, Gamma-ray Large Area Space Telescope (GLAST) Large area telescope (LAT) Science Analysis Software Level III Specification, Technical report, GE-0000X-DO, 2002
http://www.last.slac.stanford.edu/IntegrationTest/DataHandling/docs/LA%T-SS-00020-6.pdf.

[MRD, 1997]

T. U. Mead, Mission Requirements Document (MRD) Space Test Program (STP) Version 4.4, 19 June 1997, Tri-Service Experiments Mission 6 (TSX-6)
http://www.fas.org/spp/military/program/test/tsx6_mrd.htm

[PESA, 2001]

S. George, PESA high-level trigger selection software requirements. Technical report, Centre for Particle Physics at Royal Holloway University, 2001
http://www.pp.rhul.ac.uk/atlas/newsw/requirements/1.0.2/

[Stevenson et al., 2005]

M. Stevenson, M. Hartley, H. Iacovou, A. Tan, L. Phan, Software Requirements Specification for Sort Algorithm Demonstration Program, Version 2.4, 2 September 2005, SDPM

[CMS2]

CMS Manual System, Department of Health & Human Services (DHHS) Centers for Medicare & Medicaid Services (CMS), July 7, 2006
http://www.cms.hhs.gov/Transmittals/Downloads/R997CP.pdf

Books

[Anderman and Rogers, 2005]
G. Anderman and M. Rogers, In and Out of English: For Better, For Worse?, Copyright © 2005, Multilingual Matters Ltd.

[Gause and Weinberg, 1989]
D. C. Gause and D. M. Weinberg, Exploring Requirements: Quality Before Design, 1989, Dorset House Publishing Company Inc.

[Haspelmath, 2004]
M. Haspelmath, Coordinating Constructions, © 2004, John Benjamins B.V.

[Sommerville, 2001]
I. Sommerville, Software Engineering Sixth Edition, 2001 © Addison-Wesley

[Swan, 2005]
M. Swan, Practical English Usage New International Student's Edition, 2005, © Oxford University Press

Websites

Conjunction, as reviewed on 3rd July 2007, University of Ottawa, http://www.arts.uottawa.ca/writcent/hypergrammar/conjunct.html

Conjunctions, as reviewed on 3rd July 2007, Capital Community College Foundation, Hartford

http://grammar.ccc.commnet.edu/grammar/conjunctions.htm

Unless and Otherwise, as reviewed on 11 July 2007, Learning English, BBC.

http://www.bbc.co.uk/worldservice/learningenglish/grammar/learnit/learnitv272.shtml

Each and Every, as reviewed on 11 July 2007, Learning English, BBC.

http://www.bbc.co.uk/worldservice/learningenglish/grammar/learnit/learnitv110.shtml

Different uses of any, as reviewed on 11 July 2007, Learning English, BBC.

http://www.bbc.co.uk/worldservice/learningenglish/grammar/learnit/learnitv303.shtml

The definite article 'the', as reviewed on 11 July 2007, Learning English, BBC.

http://www.bbc.co.uk/worldservice/learningenglish/grammar/learnit/learnitv97.shtml

NASA SATC, ARM (Automated analysis of Requirements Specifications) Download, 2008 http://satc.gsfc.nasa.gov/tools/download/

Appendices

1. TIGER's Output for New Adelaide Airport

1 System Description

1.1 General Description The objective of the system is to provide world-class aviation services that are safe, convenient and efficient, and help to stimulate export growth, increase tourism, and promote Adelaide as abase for complementary capital investment.

1.2 Operational Requirements

1.2.1. The airport shall be a major hub for Departures from Australia to Asia.

1.2.2 The airport shall enhance Adelaide image as a world-class tourist destination.

1.2.3 The airport shall accommodate 90,000 movements in the first year of operation and increase to 180,000 five years later.

and defect type 2

1.2.3.1 The breakdown between domestic and international is 50:50.

1.2.4. The facilities shall handle the following peak profile: 7 international aircrafts with an average passenger loading of 200 and 5 cargo planes an hour and 10 domestic aircrafts per hour with an average passenger loading of 100 and 2 Helicopters per hour.

and defect type 2 3 times

1.2.5 The Largest aircraft is 747.

1.2.6 The Airport shall have facilities for: Carrying out maintenance for 10 aircraft at a time.

1.2.7 The Airport shall have facilities for: Refuelling 10 Aircraft at a time.

1.2.8 Aircraft Noise levels shall meet EPA specifications

1.2.9 The time from landing to arrival shall be less than 90 minutes for domestic and 120 minutes for international passengers.

and defect type 2

1.3 Operational Constraints

1.3.1 Airport cannot be located within a radius of 100km from the GPO.

1.3.2 The business community (potential investors in the project) like the current airport location and have voiced their opposition to building the new airport 100km north of the city.

2 Functional Requirements and Associated Performance Characteristics

2.1 System context and functional interface diagrams.

2.2 Land-Side Connections

2.2.1 The following parking and transport infrastructure shall be consistent with average and peak plane movements Hire car facilities and Taxi stands and Shuttle bus pick and drop off and Public bus connections and Cargo depots and There shall be provision for hotel accommodation

and defect type 2 8 times

Shall defect type 1

2.2.2 There shall be adequate parking for staff and visitors

Adequate defect type 3

and defect type 2

2.3 Terminal Operations

2.3.1 Buildings

2.3.1.1 All buildings shall be in line with the International Building Code (2000)

2.3.1.10 Office facilities, meeting facilities consistent with specified aircraft loadings

2.3.1.11 All buildings shall be in line with the International Building Code (2000)

2.3.1.2 Facilities shall accommodate elderly, disabled and wheelchair passengers and guests

and defect type 2 2 times

2.3.1.3 Buildings and interiors shall be designed to present Adelaide as a world-class destination. The aesthetics shall be judged by a panel assembled by the SA Government.

and defect type 2

Shall defect type 1

2.3.1.4 Design shall be consistent with specified aircraft loadings.

2.3.1.5 The airport shall have aesthetic departure and transit lounges.

and defect type 2

2.3.1.6 Passenger amenities - cafeteria, banking, post, communications (phone, mobile and internet) and frequent flyer lounges shall be provided.

and defect type 2 2 times

2.3.1.7 There shall be an arrival meeting area for international passengers.

2.3.1.8 The transit time between two departure gates shall not exceed 7 minutes.

2.3.1.9 The transit time between international and domestic terminals shall not exceed 10 minutes.

2.3.2 Ticketing Ticketing and Check-in facilities

2.3.2.1 Domestic: Maximum processing time shall be 15 minutes

2.3.2.2 International: Maximum processing time shall be 40 minutes

2.3.3 Security

2.3.3.1 Security measures as specified in Appendix A shall be put in place to screen in-going and outgoing passengers, baggage and cargo. These shall be in line with the recently revised International Security Standards adopted by the International Civil Aviation Organization (ICAO).

and	defect type 2	2 times
Shall	defect type 1	

2.3.3.2 Time taken to proceed through Security checkpoint shall be less than 10 minutes

2.3.4 Immigration and Customs

2.3.4.1 Customs facilities shall be in line with International Customs Standards adopted by the International Civil Aviation Organization (ICAO)

2.3.4.2 Immigration Facilities: Processing delay shall be no more than 15 minutes

2.3.4.3 Customs Facilities: Processing delay shall be no more than 15 minutes

2.4 Air-Side Operations

2.4.1 In-flight catering facilities shall be provided.

2.4.2 Air-bridges for all aircrafts with seating capacity greater than 70 shall be provided. Tarmac access for boarding/disembarking smaller aircraft shall be provided.

all	defect type 3
Shall	defect type 1

2.4.3 Aircraft maintenance facilities shall be provided for 10 aircraft. If facilities are offshore, there shall be adequate protection from sea environment.

Adequate	defect type 3

Shall defect type 1

2.4.4 Fuel storage and refuelling facilities shall be provided to cater for 10 aircraft at a time.

and defect type 2

2.5 Aircraft Handling

2.5.1 The design shall ensure that the availability (due to runway location and infrastructure design) of the airport for plane movements due to inclement weather is equal to or better than the current airport

and defect type 2

or defect type 2

2.6 Airside Connections

2.6.1 The design shall provide air traffic control, for the air traffic specified, in line with the standards adopted by the International Civil Aviation Organization (ICAO)

2.7 Baggage Handling

2.7.1 Baggage to be available for collection as follows:

2.7.2 Domestic: Shall be made available no more than 15 minutes after landing

2.7.3 International: Shall be made available no more than 25 minutes after landing

2.8 Freight Handling

2.8.1 Truck unloading facilities

2.8.2 Cargo handling facilities

2.8.3 Aircraft loading facilities

2.9 Utilities

2.9.1 The design of the following utilities shall be done on a site basis:

2.9.2 Power including UPS and lighting

2.9.3 Water, sewerage and storm water drainage

2.9.4 Communication systems.

3 Environmental Requirements

3.1 The noise level in built-up areas shall meet EPA specifications. These are contained in Annex APA.

3.2 The design shall pass an environmental impact study. The guidelines for preparing an impact study are contained in Annex EnvImpact.

3.3 Infrastructure shall be designed to handle a one in 100 year storm.

3.4 The design shall incorporate 10% renewable energy sources, in line with International Energy Conservation Code (2000)

4 Safety/Quarantine Requirements

4.1 An aircraft safety risk analysis of take-off and landing infrastructure (covering runway, apron, control tower and taxi design) shall demonstrate a probability of aircraft accident of less than 10-9.

and defect type 2 2 times

4.2 Airport shall provide on-site Fire and Emergency response in line with the International Fire Code (2000)

and defect type 2

4.3 The airport shall meet Australian Quarantine requirements.

5 Engineering/Construction

5.1 The construction of the facilities shall contain 50% local content

5.2 The facilities shall be designed for a 50-year life.

5.3 A modular design shall be adopted to facilitate growth.

6 Economic

6.1 The design shall have a 30-year pay back.

6.2 The design shall attract $10 million investment from local business.

2. SREE's output for New Adelaide Airport RS

This section describes SREE's analysis during a first-time run on the New Adelaide Airport RS. The output shows the kinds of potential ambiguities found in each RStat. The user of SREE validates SREE's analysis and discovers some unidentified ambiguties. SREE is unable to find the unidentified ambiguities because no indicators for them are defined in SREE's AIC. For SREE to recognise them as potentially ambiguous, they would have to be added to SREE's AIC.

```
New Adelaide Airport - System Specifications
is potentially ambiguous (PLURALNOUN) because of wording: specifications
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

specifications is not ambiguous in the RStat, but SREE recognises specifications to be potentially ambiguous because specifications is defined as a potential ambiguity indicator in the AIC.

```
1 System Description.
```

1.1 General Description The objective of the system is to provide world-class aviation services that are safe, convenient and efficient, and help to stimulate export growth, increase tourism, and promote Adelaide as abase for complementary capital investment.

is potentially ambiguous (PLURALNOUN) because of wording: services
is potentially ambiguous (VAGUE) because of wording: efficient
is potentially ambiguous (VAGUE) because of wording: general
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that that is not ambiguous, but world-class and convenient are ambiguous adjectives. However, SREE doesn't recognise world-class and convenient to be potentially ambiguous because world-class and convenient aren't defined as potential ambiguity indicators in the AIC.

1.2 Operational Requirements.

1.2.1. The airport shall be a major hub for Departures from Australia to Asia.

is potentially ambiguous (PLURALNOUN) because of wording: departures

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that major is an ambiguous adjective. However, SREE doesn't recognise major to be potentially ambiguous because major isn't defined as a potential ambiguity indicator in the AIC.

```
1.2.2 The airport shall enhance Adelaide image as a world-class tourist destination.
```

The user determined that world-class is an ambiguous adjective. However, SREE doesn't recognise world-class to be potentially ambiguous because world-class isn't defined as a potential ambiguity indicator in the AIC.

```
1.2.3 The airport shall accommodate 90,000 movements in the first year of operation and increase to 180,000 five years later.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: movements
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that movements is not ambiguous, but SREE reports movements as potentially ambiguous because movements is defined as a potential ambiguity indicator in the AIC.

```
1.2.3.1 The breakdown between domestic and international is 50:50.
```

```
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 0.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/2 * 100% = 0%**

SREE's False Positives = **p - m = 2 – 0 = 2**

Even though SREE recognises : and and as potential ambiguity indicators, : and and do not actually contribute to ambiguity in the 1.2.6 RStat. Hence this results in a false positive. On the other hand, Domestic and International are instances of potential ambiguity because it is not clear what Domestic and International refer to, such as to departures, arrivals, terminals, or even to something else. SREE does not report Domestic and International as instances of potential ambiguity because Domestic and International are not defined as potential ambiguity indicators in the AIC.

```
1.2.4. The facilities shall handle the following peak profile: 7 international aircrafts with an average passenger loading of 200 and 5 cargo planes an hour and 10 domestic aircrafts per hour with an average passenger loading of 100 and 2 Helicopters per hour.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: aircrafts
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (PLURALNOUN) because of wording: helicopters
is potentially ambiguous (PLURALNOUN) because of wording: planes
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: following
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 7, n = 7, m = 4.

SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/7 * 100% = 57%**

SREE's False Positives = **p - m = 7 – 4 = 3**

The user determined that peak is an ambiguous adjective, but aircrafts, helicopters, and planes are not ambiguous. However, SREE doesn't recognise peak to be potentially ambiguous because peak isn't defined as a potential ambiguity indicator in the AIC.

```
1.2.5 The Largest aircraft is 747.
```

```
1.2.6 The Airport shall have facilities for: Carrying out maintenance for 10 aircraft at a time.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (CONTINUANCE) because of wording: :
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

Even though SREE recognises : as a potential ambiguity indicator, : does not actually contribute to ambiguity in the 1.2.6 RStat. Hence this results in a false positive.

```
1.2.7 The Airport shall have facilities for: Refuelling 10 Aircraft at a time.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (CONTINUANCE) because of wording: :
```

Potential Ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

Even though SREE recognises : as a potential ambiguity indicator, : does not actually contribute to ambiguity in the 1.2.6 RStat. Hence this results in a false positive.

```
1.2.8 Aircraft Noise levels shall meet EPA specifications.

is potentially ambiguous (PLURALNOUN) because of wording: levels
is potentially ambiguous (PLURALNOUN) because of wording:
specifications
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

Even though SREE recognises specifications to be potentially ambiguous, however specifications does not contribute to ambiguity in the 1.2.8 RStat.

```
1.2.9 The time from landing to arrival shall be less than 90 minutes
for domestic and 120 minutes for international passengers.

is potentially ambiguous (PLURALNOUN) because of wording: passengers
is potentially ambiguous (VAGUE) because of wording: less
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

Even though SREE recognises less and passengers as potentially ambiguous instances, less and passengers do not actually contribute to ambiguity in the 1.2.9 RStat.

```
1.3   Operational Constraints.

is potentially ambiguous (PLURALNOUN) because of wording: constraints
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

constraints is not ambiguous in the above statement but SREE recognises constraints matches to the indicator of potential ambiguity in the AIC, hence SREE reports constraints to be potential ambiguity.

```
1.3.1 Airport cannot be located within a radius of 100km from the GPO.

1.3.2 The business community (potential investors in the project) like
the current airport location and have voiced their opposition to
building the new airport 100km north of the city.

is potentially ambiguous (PLURALNOUN) because of wording: investors
is potentially ambiguous (VAGUE) because of wording: (potential
investors in the project)
is potentially ambiguous (PRONOUN) because of wording: their
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

```
2 Functional Requirements and Associated Performance Characteristics.

is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 1, n = 1, m = 1. **SREE's Recall** = **p/n * 100% = 1/1 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 0/1 * 100% = 0%** **SREE's False Positives** = **p - m = 1 – 0 = 1**

The user determined that the above statement is not really an RStat. Though SREE recognises and as potentially ambiguous, and doesn't contribute to ambiguity in this statement.

```
2.1 System context and functional interface diagrams.

is potentially ambiguous (PLURALNOUN) because of wording: diagrams
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential Ambiguity with value of: p = 2, n = 2, m = 1. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/2 * 100% = 50%** **SREE's False Positives** = **p - m = 2 – 1 = 1**

The user determined that the above statement is not really an RStat. Though SREE recognises diagrams as potentially ambiguous, however diagrams does not contribute to ambiguity in this statement.

```
Figures 1 and 2 show the System context and functional interface
diagrams for the Airport.

is potentially ambiguous (PLURALNOUN) because of wording: figures
is potentially ambiguous (PLURALNOUN) because of wording: diagrams
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.
SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**
SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**
SREE's False Positives = **p - m = 3 – 2 = 1**

SREE recognises diagrams to be potentially ambiguous as defined in the AIC, but the user reads diagrams to be unambiguous in the statements.

```
2.2 Land-Side Connections.
is potentially ambiguous (PLURALNOUN) because of wording: connections
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

Even though the above statement is not an RStat, as long as SREE recognises the instance connections matches to the indicator of potential ambiguity in the AIC, SREE reports connections to be potential ambiguity.

```
2.2.1 The following parking and transport infrastructure shall be consistent with average and peak plane movements Hire car facilities and Taxi stands and Shuttle bus pick and drop off and Public bus connections and Cargo depots and There shall be provision for hotel accommodation.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: depots
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (PLURALNOUN) because of wording: movements
is potentially ambiguous (PLURALNOUN) because of wording: stands
is potentially ambiguous (CONTINUANCE) because of wording: following
```

```
is potentially ambiguous (VAGUE) because of wording: consistent
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (VAGUE) because of wording: there
```

Potential ambiguity with value of: p = 9, n = 9, m = 6.

SREE's Recall = **p/n * 100% = 9/9 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/9 * 100% = 67%**

SREE's False Positives = **p - m = 9 – 6 = 3**

The user determined that depots, stands, and connections are not potentially ambiguous instances though SREE recognises them as instances of potential ambiguity as defined in SREE's AIC.

```
2.2.2 There shall be adequate parking for staff and visitors.

is potentially ambiguous (PLURALNOUN) because of wording: visitors
is potentially ambiguous (VAGUE) because of wording: adequate
is potentially ambiguous (VAGUE) because of wording: there
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

```
2.3 Terminal Operations.

is potentially ambiguous (PLURALNOUN) because of wording: operations
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though the above statement is not an RStat, as long as SREE recognises the instance operations matches to the potential ambiguity indicator in the AIC, SREE reports operations to be potentially ambiguous.

```
2.3.1 Buildings.

is potentially ambiguous (PLURALNOUN) because of wording: buildings
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though the above statement is not an RStat, as long as SREE recognises the instance buildings matches to the indicator of potential ambiguity in the AIC, SREE reports buildings to be potentially ambiguous.

```
2.3.1.1 All buildings shall be in line with the International Building Code (2000).

is potentially ambiguous (PLURALNOUN) because of wording: buildings
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: (2000)
```

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

```
2.3.1.2 Facilities shall accommodate elderly, disabled and wheelchair passengers and guests.

is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (PLURALNOUN) because of wording: guests
```

```
is potentially ambiguous (PLURALNOUN) because of wording: passengers
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that guests and passengers are not potentially ambiguous instances despite their being plural nouns. SREE recognises guests and passengers as potential ambiguity indicators as indicated in its AIC.

```
2.3.1.3 Buildings and interiors shall be designed to present Adelaide as a world-class destination.  The aesthetics shall be judged by a panel assembled by the SA Government.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: buildings
is potentially ambiguous (PLURALNOUN) because of wording: interiors
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

The user determined that world-class and aesthetics are potentially ambiguous instances. However, SREE doesn't recognise world-class and aesthetics to be potentially ambiguous because world-class and aesthetics aren't defined as indicators of potential ambiguity in the AIC.

```
2.3.1.4 Design shall be consistent with specified aircraft loadings.
```

```
is potentially ambiguous (VAGUE) because of wording: consistent
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that specified and loadings are instances of potential ambiguity. However, SREE doesn't recognise specified and loadings to be potentially ambiguous because specified and loading aren't defined as potential ambiguity indicators in the AIC.

```
2.3.1.5 The airport shall have aesthetic departure and transit lounges.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: lounges
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that aesthetic is potentially ambiguous, but she reads lounges to be unambiguous. SREE doesn't recognise aesthetic to be a potential ambiguity due to aesthetic isn't defined as a potential ambiguity indicator in the AIC.

```
2.3.1.6 Passenger amenities - cafeteria, banking, post, communications
(phone, mobile and internet) and frequent flyer lounges shall be
provided.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: amenities
is potentially ambiguous (PLURALNOUN) because of wording:
communications
is potentially ambiguous (PLURALNOUN) because of wording: lounges
is potentially ambiguous (VAGUE) because of wording: (phone, mobile
and internet)
```

```
is potentially ambiguous (VAGUE) because of wording: provided
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 5.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/6 * 100% = 83%**

SREE's False Positives = **p - m = 6 – 5 = 1**

The user recognises lounges to be unambiguous despite its being plural. However, SREE recognises lounges as potentially ambiguous as lounges is defined as a potential ambiguity indicator in the AIC.

```
2.3.1.7 There shall be an arrival meeting area for international passengers.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: passengers
is potentially ambiguous (VAGUE) because of wording: there
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

SREE recognises passengers to be potentially ambiguous as defined in its AIC under Plural category that contains specific list of plural nouns. However, even though passengers is a plural noun, passengers is not a subject where the ambiguity mostly incurs, hence does not contribute to ambiguity in the 2.3.1.7 RSrat.

```
2.3.1.8 The transit time between two departure gates shall not exceed 7 minutes.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: gates
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though SREE detects gates to be a potentially ambiguous plural noun, however gates doesn't really contribute ambiguity to 2.3.1.8 RStat.

```
2.3.1.9 The transit time between international and domestic terminals
shall not exceed 10 minutes.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: terminals
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that terminals is not ambiguous even though SREE recognises terminals as potentially ambiguous because it appears as a potential ambiguity indicator in the AIC.

```
2.3.1.10 Office facilities, meeting facilities consistent with
specified aircraft loadings.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (VAGUE) because of wording: consistent
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that specified and loadings are instances of potential ambiguity. However, SREE doesn't recognise specified and loadings to be potentially ambiguous because specified and loadings aren't defined as potential ambiguity indicators in the AIC.

```
2.3.1.11 All buildings shall be in line with the International
Building Code (2000).

is potentially ambiguous (PLURALNOUN) because of wording: buildings
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: (2000)
```

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

```
2.3.2 Ticketing Ticketing and Check-in facilities.

is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 0.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

Even though 2.3.2 is not really an RStat, SREE recognises facilities to be potentially ambiguous because facilities is defined as a potential ambiguity indicator in the AIC.

```
2.3.2.1 Domestic: Maximum processing time shall be 15 minutes.

is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: maximum
```

PA with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

Even though SREE recognises maximum to be a potential ambiguity, maximum doesn't contribute to ambiguity in the 2.3.2.1 RStat. However domestic is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
2.3.2.2 International: Maximum processing time shall be 40 minutes.

is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: maximum
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

Even though SREE recognises maximum to be a potential ambiguity as defined in the AIC, maximum doesn't contribute to ambiguity in the 2.3.2.1 RStat. However international is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
2.3.3 Security.
```

2.3.3.1 Security measures as specified in Appendix A shall be put in place to screen in-going and outgoing passengers, baggage and cargo. These shall be in line with the recently revised International Security Standards adopted by the International Civil Aviation Organization (ICAO).

is potentially ambiguous (INCOMPLETES) because of wording: as specified
is potentially ambiguous (PLURALNOUN) because of wording: measures
is potentially ambiguous (PLURALNOUN) because of wording: passengers
is potentially ambiguous (PLURALNOUN) because of wording: standards
is potentially ambiguous (VAGUE) because of wording: (icao)
is potentially ambiguous (PRONOUN) because of wording: these
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 7, n = 7, m = 5.

SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/7 * 100% = 71%**

SREE's False Positives = **p - m = 7 – 5 = 2**

The user determined that recently and specified are instances of potential ambiguity, but passengers and standards are not instances of potential ambiguity in the RStat. SREE doesn't recognise recently and specified to be potentially ambiguous because recently and specified aren't defined as potential ambiguity indicators in the AIC.

2.3.3.2 Time taken to proceed through Security checkpoint shall be less than 10 minutes.

is potentially ambiguous (VAGUE) because of wording: less

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though SREE recognises less to be a potential ambiguity as defined in the AIC, less doesn't contribute to ambiguity in the 2.3.2.1 RStat.

```
2.3.4 Immigration and Customs.

is potentially ambiguous (PLURALNOUN) because of wording: customs
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 0.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/2 * 100% = 0%**

SREE's False Positives = **p - m = 2 – 0 = 2**

The 2.3.4 statement is not really an RStat. Even though SREE recognises and and customs to be potentially ambiguous instances as defined in the AIC, they do not contribute to ambiguity.

```
2.3.4.1 Customs facilities shall be in line with International Customs Standards adopted by the International Civil Aviation Organization (ICAO).

is potentially ambiguous (PLURALNOUN) because of wording: customs
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (PLURALNOUN) because of wording: standards
is potentially ambiguous (VAGUE) because of wording: (icao)
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

Even though SREE recognises standards and customs to be instances of potential ambiguity as defined in the AIC, however standards and customs do not contribute to ambiguity in the 2.3.4.1 RStat.

```
2.3.4.2 Immigration Facilities: Processing delay shall be no more than 15 minutes.
```

```
is potentially ambiguous (VAGUE) because of wording: facilities
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 66.7%**

SREE's False Positives = **p - m = 3 – 2 = 1**

Even though SREE recognises more to be potentially ambiguous as defined in the AIC, however more does not contribute to ambiguity in the 2.3.4.2 RStat.

```
2.3.4.3 Customs Facilities: Processing delay shall be no more than 15 minutes.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: customs
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

Even though SREE recognises more and customs to be potentially ambiguous instances as defined in the AIC, however more and customs do not contribute to ambiguity in the 2.3.4.3 RStat.

```
2.4   Air-Side Operations.
is potentially ambiguous (PLURALNOUN) because of wording: operations
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The 2.4 statement is not really an RStat. Despite that, SREE reports operations to be an potentially ambiguous because operations is defined as a potential ambiguity indicator in the AIC.

```
2.4.1 In-flight catering facilities shall be provided.
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user recognises facilities to be unambiguous although SREE recognises facilities to be potentiallty ambiguous because facilities is defined as a potential ambiguity indicator in the AIC.

2.4.2 Air-bridges for all aircrafts with seating capacity greater than 70 shall be provided. Tarmac access for boarding/disembarking smaller aircraft shall be provided.

```
is potentially ambiguous (PLURALNOUN) because of wording: aircrafts
is potentially ambiguous (PLURALNOUN) because of wording: bridges
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 5, n = 5, m = 5.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/5 * 100% = 100%**

SREE's False Positives = **p - m = 5 – 5 = 0**

2.4.3 Aircraft maintenance facilities shall be provided for 10 aircraft. If facilities are offshore, there shall be adequate protection from sea environment.

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (VAGUE) because of wording: adequate
is potentially ambiguous (VAGUE) because of wording: provided
is potentially ambiguous (VAGUE) because of wording: there
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

2.4.4 Fuel storage and refuelling facilities shall be provided to cater for 10 aircraft at a time.

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (VAGUE) because of wording: provided
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 66.7%**

SREE's False Positives = **p - m = 3 – 2 = 1**

Even though SREE recognises facilities to be potentially ambiguous as defined in the AIC, however facilities does not contribute to ambiguity in the 2.4.4 RStat.

```
2.5   Aircraft Handling.

2.5.1 The design shall ensure that the availability (due to runway location and infrastructure design) of the airport for plane movements due to inclement weather is equal to or better than the current airport.

is potentially ambiguous (PLURALNOUN) because of wording: movements
is potentially ambiguous (VAGUE) because of wording: (due to runway
location and infrastructure design)
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

Even though SREE recognises movements and and to be instances of potential ambiguity as defined in the AIC, however both movements and and do not contribute to ambiguity in the 2.5.1 RStat.

```
2.6   Airside Connections.

is potentially ambiguous (PLURALNOUN) because of wording: connections
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though 2.6 statement is not really an RStat,SREE recognises connections to be potentially ambiguous because connections is defined as a potential ambiguity indicator in the AIC.

```
2.6.1 The design shall provide air traffic control, for the air traffic specified, in line with the standards adopted by the International Civil Aviation Organization (ICAO).

is potentially ambiguous (PLURALNOUN) because of wording: standards
is potentially ambiguous (VAGUE) because of wording: (icao)
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that specified is an ambiguous adjective. However, SREE doesn't recognise specified to be potentially ambiguous because specified isn't defined as an indicator of potential ambiguity in the AIC.

```
2.7   Baggage Handling.

2.7.1 Baggage to be available for collection as follows:

is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: as follows
```

Potential ambiguity with value of: p = 2, n = 2, m = 2. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 2/2 * 100% = 100%** **SREE's False Positives** = **p - m = 2 – 2 = 0**

```
2.7.2 Domestic: Shall be made available no more than 15 minutes after
landing.

is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 2, n = 2, m = 1. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/2 * 100% = 50%** **SREE's False Positives** = **p - m = 2 – 1 = 1**

Even though SREE recognises more to be potentially ambiguous as defined in the AIC, more does not contribute to ambiguity in the 2.7.2 RStat. However domestic is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
2.7.3 International: Shall be made available no more than 25 minutes
after landing.

is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 2, n = 2, m = 2. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/2 * 100% = 50%** **SREE's False Positives** = **p - m = 2 – 1 = 1**

Even though SREE recognises more to be potentially ambiguous as defined in the AIC, more does not contribute to ambiguity in the 2.7.3 RStat. However international is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
2.8   Freight Handling.
2.8.1 Truck unloading facilities.
is potentially ambiguous (PLURALNOUN) because of wording: facilities
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user recognises facilities to be unambiguous despite that SREE recognises it as a potential ambiguity due to its definition as a potential ambiguity indicator in the AIC.

```
2.8.2 Cargo handling facilities.
is potentially ambiguous (PLURALNOUN) because of wording: facilities
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined facilities to be unambiguous despite that SREE recognises it as a potential ambiguity due to its definition as a potential ambiguity indicator in the AIC.

```
2.8.3 Aircraft loading facilities.
is potentially ambiguous (PLURALNOUN) because of wording: facilities
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined facilities to be unambiguous despite that SREE recognises it as a potential ambiguity due to its definition as a potential ambiguity indicator in the AIC.

```
2.9   Utilities.

is potentially ambiguous (PLURALNOUN) because of wording: utilities
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

Even though 2.9 statement is not really an RStat, SREE recognises utilities to be potentially ambiguous because utilities is defined as a potential ambiguity indicator in the AIC.

```
2.9.1 The design of the following utilities shall be done on a site basis:

is potentially ambiguous (PLURALNOUN) because of wording: utilities
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: following
```

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

`2.9.2 Power including UPS and lighting.`

`is potentially ambiguous (COORDINATOR) because of wording: and`

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 2.9.2 RStat.

`2.9.3 Water, sewerage and storm water drainage.`

`is potentially ambiguous (COORDINATOR) because of wording: and`

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

`2.9.4 Communication systems.`

`is potentially ambiguous (PLURALNOUN) because of wording: systems`

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined systems is not ambiguous in the RStat despite that that SREE recognises it as a potential ambiguity as indicated in SREE's AIC.

3 Environmental Requirements.

3.1 The noise level in built-up areas shall meet EPA specifications.

These are contained in Annex APA.

is potentially ambiguous (PLURALNOUN) because of wording: areas
is potentially ambiguous (PLURALNOUN) because of wording: specifications
is potentially ambiguous (PRONOUN) because of wording: these

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that specifications is not ambiguous despite that SREE recognises it as a potential ambiguity in SREE's AIC.

3.2 The design shall pass an environmental impact study. The guidelines for preparing an impact study are contained in Annex EnvImpact.

is potentially ambiguous (PLURALNOUN) because of wording: guidelines

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

3.3 Infrastructure shall be designed to handle a one in 100 year storm.

3.4 The design shall incorporate 10% renewable energy sources, in line with International Energy Conservation Code (2000).

is potentially ambiguous (PLURALNOUN) because of wording: sources
is potentially ambiguous (VAGUE) because of wording: (2000)

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
4     Safety/Quarantine Requirements.

is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
4.1 An aircraft safety risk analysis of take-off and landing
infrastructure (covering runway, apron, control tower and taxi design)
shall demonstrate a probability of aircraft accident of less than 10-9.

is potentially ambiguous (VAGUE) because of wording: (covering runway,
apron, control tower and taxi design)
is potentially ambiguous (VAGUE) because of wording: less
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

Even though SREE recognises less to be a potentially ambiguous instance as defined in the AIC, less does not contribute to ambiguity in the 4.1 RStat.

4.2 Airport shall provide on-site Fire and Emergency response in line with the International Fire Code (2000).

is potentially ambiguous (VAGUE) because of wording: (2000)
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

4.3 The airport shall meet Australian Quarantine requirements.

5 Engineering/Construction.

is potentially ambiguous (VAGUE) because of wording: /

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.1 The construction of the facilities shall contain 50% local content.

is potentially ambiguous (PLURALNOUN) because of wording: facilities

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.2 The facilities shall be designed for a 50-year life.

is potentially ambiguous (PLURALNOUN) because of wording: facilities

Potential Ambiguity with value of: p = 1, n = 1, m = 1. **SREE's Recall** = **p/n * 100% = 1/1 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/1 * 100% = 100%** **SREE's False Positives** = **p - m = 1 – 1 = 0**

```
5.3 A modular design shall be adopted to facilitate growth.
6 Economic
6.1 The design shall have a 30-year pay back.
6.2 The design shall attract $10 million investment from local business.
```

3. TIGER's output for MCSS

5.1 MCSS SYSTEM DESCRIPTION

missing "shall" defect type 4

5.10 TECHNICAL SUPPORT

missing "shall" defect type 4

5.10.1 A software users guide.

Missing "shall" defect type 4

5.9.9 Test plans and test procedures used in accepting the MCSS.

And defect type 2

missing "shall" defect type 4

5.10.11 The MCSS hardware and software documentation shall be at the same revision level as the hardware and software.

And defect type 2 2 times

5.10.12 All identical parts shall be at the same revision level.

5.10.13 At least 10 percent spares (minimum one item) for each replaceable hardware item (LRU) shall be provided.

5.10.14 At least 20 percent spares (minimum 2 items) for each replaceable Application Specific Integrated Circuit (ASIC) shall be provided.

5.10.15 All test rigs shall be delivered with the MCSS.

5.10.16 Test rig documentation shall be delivered with the MCSS.

5.10.17 Test rig documentation requirements shall be the same as those in paragraphs 5.10.1 through 5.10.11.

5.10.18 Custom test rig spare parts requirements shall be the same as those in paragraphs 5.10.12 and 5.10.13.

and defect type 2

5.9.9 MCSS technical manuals for the operation, maintenance, and testing of all the hardware components and systems of the switching system, described as follows:

all defect type 3

and defect type 2 2 times

missing "shall" defect type 4

5.9.9 The documentation needed to make use of the growth capabilities of the MCSS in terms of hardware requirements and software parameters.

And defect type 2

missing "shall" defect type 4

5.9.9 Training manuals for new personnel to be trained, both on hardware and software operations.

And defect type 2

missing “shall” defect type 4

5.9.9 Software source code, libraries, object modules, and custom software developed for the MCSS.

And defect type 2

missing “shall” defect type 4

5.9.9 Computer manufacturer reference and system programming manuals detailing machine instructions and programming considerations.

And defect type 2 2 times

missing “shall” defect type 4

5.9.9 Documentation of custom modifications and changes to purchased software.

And defect type 2

missing “shall” defect type 4

5.9.9 Detailed reference manuals describing all elements and operations of supplied software (e.g., language compilers, text editors, communications drivers, software tools, diagnostics, interface drivers).

All defect type 3

and defect type 2

missing “shall” defect type 4

5.9.9 Problem determination and debugging guides that shall include documentation of known problems and suspected system errors.

And defect type 2 2 times

includedefect type 3

5.11 TRAINING

missing “shall” defect type 4

5.9.9 The MCSS shall include a training plan that shall include a set of outlines for training courses.

Includedefect type 3 2 times

Shall defect type 1

5.11.2 The outline of each course shall include, but not be limited to, course objectives, topics addressed, prerequisite levels of technical skills (if any), and duration of the course (in hours and days).

And defect type 2 2 times

5.11.3 Each trainee shall be provided with appropriate written course material.

Appropriate defect type 3

5.11.4 A minimum of 4 contiguous hours of operations training shall be provided for a minimum of 4 operations personnel for each of 4 shifts.

5.11.5 A minimum of 40 contiguous hours of maintenance training for 8 persons shall be provided.

5.11.6 The location, time, and content of each course shall be subject to review and approval by the Government.

And defect type 2 2 times

5.11.7 Postponed courses shall be rescheduled with the mutual agreement of the Government and the MCSS supplier.

And defect type 2

5.2 SWITCH MATRIX SUBSYSTEM

missing “shall” defect type 4

5.2.1 The MCSS shall contain 255 full duplex ports.

5.2.10 This subsystem shall provide the capability to connect any one input to up to a minimum of 32 outputs simultaneously.

Any defect type 3

5.2.11 This subsystem shall not allow two or more inputs to be connected to the same output simultaneously.

Or defect type 2

5.2.12 This subsystem shall not allow data to appear on an output when that output is not connected to any input.

Any defect type 3

5.2.13 This subsystem shall place a mark state on every output when that output is not connected to any input.

Any defect type 3

5.9.9 The signal delay between any input port and its output port through the MCSS shall be less than 1 millisecond.

And defect type 2

any defect type 3

5.2.15 The delay on any signal path through the MCSS shall be within 10 percent of the delay on any other signal path.

Any defect type 3 2 times

5.2.16 The delay between input and output ports through the MCSS shall be independent of the data rate.

And defect type 2

5.2.17 The crosstalk between signals passing through the MCSS shall be less than -70 decibels (dB).

5.2.18 The added error to signals switched through the MCSS shall be less than 1 in 108 bits.

5.2.19 The frequency of each signal through the MCSS shall be between 0 and 6.312 MHz.

and defect type 2

5.2.2 The MCSS shall switch binary digital signals.

5.2.20 Each signal path through the MCSS shall be capable of passing data at 0 to 6.312 Mbps.

5.2.21 The MCSS shall be capable of passing data at up to 6.312 Mbps on all signal paths simultaneously.

All defect type 3

5.2.22 The MCSS interface to external equipment/data lines shall be Data Communication Equipment (DCE) using DB-15S connectors.

5.2.23 Each external data/timing port interface to the MCSS shall support both

5.2.24 The MCSS shall visually indicate that a specific port is configured as RS-422-A or RS-423-A.

or defect type 2

5.2.25 The MCSS shall contain five Nascom-supplied line driver chassis.

5.2.26 The MCSS shall interface with the Nascom-supplied line drivers.

5.2.27 The MCSS shall provide patch panel capability for each of the Nascom lines.

5.2.28 Thirty-two sets of bypass (patch) cables to enable the MCSS to be bypassed shall be provided.

5.2.28.1 This bypass capability shall be on both Nascom-to-MSOCC and MSOCC-to-MSOCC connections.

And defect type 2

5.2.3 The MCSS shall switch serial signals.

5.2.4 This subsystem shall switch data and timing in the same direction as signal pairs.

And defect type 2

5.9.9 The phase relationship between data and timing signals shall be maintained in each signal pair passing through the MCSS such that the variance shall not be greater than 5 percent.

And defect type 2

Shall defect type 1

5.2.6 The MCSS shall connect specific inputs to specific outputs.

5.2.7 The MCSS shall disconnect specific inputs from specific outputs.

5.2.8 This subsystem shall establish connections between 255 inputs and 255 outputs.

And defect type 2

5.9.9 This subsystem shall establish connections between any one input and any one of the 255 outputs.

And defect type 2

any defect type 3 2 times

5.3 SWITCH CONTROL SUBSYSTEM

missing “shall” defect type 4

5.3.1 This subsystem shall be the MCSS operator's control interface.

5.3.10 This subsystem shall only allow the renaming of a switch port mnemonic for a port that is not connected at the time.

5.3.11 This subsystem shall provide the capability to allow the control interface to select the active switch control unit.

5.3.12 This subsystem shall contain two identical LCT's.

5.3.13 The LCT display shall have a diagonal measurement of at least 19 inches.

5.3.14 The LCT display shall be capable of displaying at least 16 colors.

5.3.15 The LCT display shall have a resolution of at least 640 horizontal pixels and 480 vertical pixels.

And defect type 2

5.3.16 The information on the LCT display screen shall have colors and presentation characteristics for graphics and text usage similar to the DOCS switch configuration display screens.

And defect type 2 2 times

5.3.17 The LCT operator interface shall be functionally similar to the DOCS operator interface for switch-related functions.

5.3.18 Keyboard entries on the LCT shall be displayed on its screen within 0.5 seconds of keystroke entry.

5.3.19 The LCT shall have the capability to display all port mnemonics.

All defect type 3

5.3.2 This subsystem shall allow a user with designated privileges to establish and modify passwords for access to this subsystem.

And defect type 2

5.3.20 The LCT display of the switch configuration shall be identical to the actual switch configuration.

5.3.21 The LCT shall update the switch status display within 2 seconds of a change in the switch configuration.

5.3.22 The LCT shall make and break connections between sets of inputs and outputs as identified in a command file.

And defect type 2 2 times

5.3.23 This subsystem shall process control messages from any control interface.

Any defect type 3

5.3.24 This subsystem shall process control messages in the order in which they are received.

5.3.25 This subsystem shall acknowledge control messages only to the interface originating the control message.

5.3.26 This subsystem shall communicate with the control interface only in response to a control message.

5.3.27 This subsystem will identify at least 11 control message errors.

Will defect type 4

5.3.27.1 One of the 11 control message errors for this subsystem shall be 'Invalid American Standard Code for Information Interchange (ASCII) digit'.

5.3.27.10 One of the 11 control message errors for this subsystem shall be 'Unable to communicate to control interface RS-232-C port'.

5.3.27.11 One of the 11 control message errors for this subsystem shall be ‘Parity error on the control interface’.

5.3.27.2 One of the 11 control message errors for this subsystem shall be ‘Invalid latch or host number’.

Or defect type 2

5.3.27.3 One of the 11 control message errors for this subsystem shall be ‘Invalid latch setting’.

5.3.27.4 One of the 11 control message errors for this subsystem shall be ‘Invalid connection’.

5.3.27.5 One of the 11 control message errors for this subsystem shall be ‘Invalid message length’.

5.3.27.6 One of the 11 control message errors for this subsystem shall be ‘Invalid message type’.

5.3.27.7 One of the 11 control message errors for this subsystem shall be ‘Latch failure’.

5.3.27.8 One of the 11 control message errors for this subsystem shall be ‘Latch not installed’.

5.3.27.9 One of the 11 control message errors for this subsystem shall be ‘Latch already connected’.

5.3.28 This subsystem shall report any error status to the control interface originating the control message.

Any defect type 3

5.3.29 This subsystem shall not attempt to execute any control message containing an error.

Any defect type 3

5.3.3 This subsystem shall require the use of passwords to set up a session to access configuration tables and files.

And defect type 2

5.3.30 This subsystem shall respond to DOCS status requests once every 5 seconds, in accordance with the Interface Control Document (ICD) Between MSOCC DOCS and MSOCC Automated Switching Systems, CSC/TM-83/6105UDI, April 1989.

And defect type 2

5.3.31 This subsystem shall respond to commands within 10 seconds of power-up reset.

5.3.32 This subsystem shall execute and acknowledge commands within 0.5 seconds of receipt of a valid command.

And defect type 2

5.3.33 This subsystem shall not permit connection commands for a specific connection to disrupt any existing connections.

Any defect type 3

5.3.34 This subsystem shall not permit disconnection commands for a specific connection to disrupt any other existing connections.

Any defect type 3

5.3.35 This subsystem shall not set priorities for connections.

5.3.36 This subsystem shall respond to a status request within 0.5 seconds of receipt of the request.

5.3.37 This subsystem shall supply the status of the switch control units.

5.3.38 The MCSS shall provide status information for each port.

5.3.38.1 The status information for each port shall indicate the Port is functional (hardware exists).

5.3.38.2 The status information for each port shall indicate the Port is nonfunctional (hardware does not exist or failed to operate properly).

Or defect type 2

5.3.39 This subsystem shall supply status information indicating which outputs are connected to a specific inputs.

5.3.4 This subsystem shall require the use of passwords to set up a session to control, assign, or modify switch connections.

Or defect type 2

5.3.40 This subsystem shall supply status information indicating which inputs are connected to specific outputs.

5.3.41 This subsystem shall perform two kinds of diagnostics: background and troubleshooting.

And defect type 2

5.3.42 The LCT shall be able to display a menu of available diagnostic tests, with a brief description of their purpose.

5.3.43 A particular diagnostic test shall be activated by the LCT.

5.3.44 A particular diagnostic test shall be aborted by the LCT.

5.3.45 The LCT shall be able to display the current status of active diagnostic tests.

5.3.46 This subsystem shall accumulate all diagnostic test results.

All defect type 3

5.3.47 All accumulated diagnostic test results shall be viewable through the LCT.

5.3.48 This subsystem shall only execute troubleshooting diagnostics under LCT control.

5.3.49 The LCT shall limit access to troubleshooting diagnostic tests by means of a password protection scheme.

5.3.5 This subsystem shall identify each port with a unique mnemonic.

5.3.50 This subsystem shall require confirmation by the operator before altering the switch configuration when performing troubleshooting diagnostics.

5.3.51 The MCSS shall perform a background diagnostic self-test at least once per hour, or on command from the LCT, to determine the proper operation of all unused circuits.

All defect type 3

or defect type 2

5.3.52 The MCSS shall be able to execute background diagnostic tests without affecting the operational control of the MCSS from the DOCS.

5.3.53 The MCSS diagnostic tests shall have the capability to determine a failure down to the LRU level.

5.3.54 Background diagnostics tests shall not change any existing switch connections.

Any defect type 3

5.3.55 The MCSS shall reject control messages from any external control interface when performing troubleshooting diagnostics.

Any defect type 3

5.3.56 This subsystem shall display hardware error messages on the LCT upon detection.

5.3.57 The MCSS control and status interface shall be full duplex RS-232-C compliant for the four EIA signals.

And defect type 2

5.3.57.1 One of the four EIA signals shall be Transmitted data (BA)

5.3.57.2 One of the four EIA signals shall be Received data (BB)

5.3.57.3 One of the four EIA signals shall be Protective ground (AA)

5.3.57.4 One of the four EIA signals shall be Signal ground (AB).

5.3.58 The MCSS shall communicate with the DOCS at discrete selectable baud rates, including 9600 and 19200.

And defect type 2

5.3.59 The MCSS control and status interface data format shall be 1 start bit, 7 ASCII data bits, odd parity bit and 2 stop bits.

And defect type 2 2 times

5.3.6 Each port mnemonic shall be user definable.

5.3.60 The MCSS control and status interface shall be DTE using DB-25-S connectors.

And defect type 2

5.9.9 An MCSS power-up reset shall clear all hardware and software registers and memory, and shall initialize all outputs to a disconnected and operational state within 10 seconds.

All defect type 3 2 times

and defect type 2 4 times

Shall defect type 1

5.9.9 This subsystem shall perform a self-test on the MCSS upon power up. The self-test shall, as a minimum, verify the operational status of all controllers and perform a memory check on all RAM memory.

All defect type 3 2 times

and defect type 2

5.9.9 An MCSS soft reset shall reset and clear all controlling mechanisms used on the switch and bring them to a predetermined state.

All defect type 3

and defect type 2 2 times

5.3.64 An MCSS soft reset shall not affect data/timing signals being routed through the switch.

5.3.65 Actions initiated by an MCSS soft reset shall be completed within 10 seconds.

5.3.66 The LCT shall be capable of initiating a soft reset.

5.3.67 A front panel control shall be capable of initiating a soft reset.

5.3.68.1 The LCT shall generate RS-170-A video signals as separate red, green, and blue (RGB) signals

and defect type 2

5.3.68.2 The LCT shall generate RS-170-A video signals as negative synchronization pulses on the green signal.

5.3.7 This subsystem shall allow each port mnemonic to contain at least 8 characters.

5.3.8 This subsystem shall assign a standard default mnemonic for a port if a mnemonic has not been predefined for the port.

5.3.9 The assigned default mnemonic shall be in the form DTExxx or DCExxx, where xxx is the port number and identifies the port mnemonic assigned.

And defect type 2

or defect type 2

5.4 TIMING GENERATOR SUBSYSTEM

missing "shall" defect type 4

5.4.1 This subsystem shall contain identical primary and backup ST signal generators.

And defect type 2

5.4.2 This subsystem shall generate at least the following frequencies simultaneously:

5.4.2.1 One of the frequencies shall be 9.6 kHz +/- 1 percent

5.4.2.2 One of the frequencies shall be 19.2 kHz +/- 1 percent

5.4.2.3 One of the frequencies shall be 56.0 kHz +/- 1 percent

5.4.2.4 One of the frequencies shall be 224.0 kHz +/- 1 percent

5.4.2.5 One of the frequencies shall be 1.544 MHz +/- 1 percent

5.4.2.6 One of the frequencies shall be 2.048 MHz +/- 1 percent

5.4.2.7 One of the frequencies shall be 6.312 MHz +/- 1 percent

5.4.3 This subsystem shall be capable of accepting up to 10 external timing signals from Nascom for simultaneous distribution as ST.

5.4.4 This subsystem shall distribute ST signals to a minimum of 255 ports.

5.4.5 The MCSS ST voltage levels shall be RS-422-A compatible.

5.5 TEST AND MONITORING SUBSYSTEM

and defect type 2

missing "shall" defect type 4

5.5.1 This subsystem shall not affect the signals being monitored.

5.5.10 The BED shall indicate the presence of data and timing.

And defect type 2

5.5.11 The BED shall perform a 22-bit CRC on the 4800-bit Nascom blocks passing through the matrix switch subsystem.

5.5.12 The BED shall indicate the presence of CRC errors.

5.5.13 The BED shall display the number of bad blocks detected.

5.5.14 The BED display shall be able to be reset to zero.

5.5.15 The BED shall increment a bad block counter each time it identifies an error in a Nascom block.

5.5.16 This subsystem shall contain a rack-mounted oscilloscope.

5.5.17 The oscilloscope shall have a Y channel frequency response of at least 20 MHz.

5.5.18 The oscilloscope shall have at least 2 Y (vertical) channels.

5.5.19 The oscilloscope shall have floating inputs (i.e., nongrounded signal return lines).

5.5.2 This subsystem shall not affect any signals not being monitored and that are passing through the MCSS.

And defect type 2

any defect type 3

5.5.20 The Nascom line status shall be visible in the DOCS room for trouble- shooting purposes.

5.5.3 This subsystem shall contain at least 10 NBG’s.

5.5.4 The NBG shall be able to generate valid 4800-bit static Nascom blocks at the timing frequencies identified in paragraph 5.4.2.

5.5.5 The NBG shall be able to generate bad 4800-bit static Nascom blocks (with incorrect CRC) at the timing frequencies identified in paragraph 5.4.2.

5.5.6 The NBG output timing and signal voltage levels shall be capable of being set to RS-422-A or RS-423-A compatible.

And defect type 2

or defect type 2

5.5.7 The MCSS shall pass blocks generated by the NBG and corresponding timing signals as a signal pair through the switch.

And defect type 2

5.5.8 This subsystem shall contain at least 10 BED's.

5.5.9 This subsystem shall allow the operator to connect an oscilloscope and/or a BED to any input or output signal port in a monitoring capacity.

Any defect type 3

or defect type 2

5.6. GROWTH

missing "shall" defect type 4

5.6.1 The MCSS shall have a 50-percent growth capability in the number of input and output ports.

And defect type 2

5.6.2 The MCSS shall be expandable without requiring removal of any equipment from the MSOCC.

Any defect type 3

5.6.3 The MCSS shall be expandable without requiring down times greater than 30 minutes for more than 25 percent of the overall switching capacity at any one time.

any defect type 3

5.6.4 The MCSS shall be capable of being expanded to provide ST to any port requiring ST.

any defect type 3

5.9.┐ RELIABILITY, MAINTAINABILITY, AND AVAILABILITY

and defect type 2

missing "shall" defect type 4

5.7.1 Availability for the MCSS shall be at least 0.99998.

5.7.10 The MCSS shall be capable of operating on either one or both of its independent power supplies at any one time.

any defect type 3

or defect type 2

5.7.11 In the event of a failure of one of the power supplies, the MCSS shall continue to operate on the other power supply without affecting switch operation.

5.7.12 A changeover from one operational power supply to two operational power supplies shall not affect operation of the MCSS.

5.7.13 A single failure of any component in the MCSS shall not disrupt more than 25 percent of the overall switching capacity.

Any defect type 3

5.7.14 The removal of an LRU shall not require the MCSS to be powered down.

5.7.15 The removal of an LRU shall affect or disable no more than four signal pairs.

Or defect type 2

5.9.9 The replacement of an LRU while the MCSS is powered on shall not disrupt or impact any circuits other than those connected to the LRU being replaced.

Any defect type 3

or defect type 2

5.9.9 The MCSS shall have sufficient data generation and acquisition tools to troubleshoot, replace, and verify the proper operation of all LRU's of the MCSS.

All defect type 3

and defect type 2 2 times

Sufficient defect type 3

5.9.9 During the course of troubleshooting, when access to areas under investigation is obtained by sliding, rotating, or hinged parts, such parts shall be free to open, extend, or rotate full distance and remain in the open state without requiring additional support.

And defect type 2

or defect type 2 2 times

5.7.19 Clearance for maintenance access shall be provided.

5.7.2 The MTBF for the MCSS, except for the LCT, shall not be less than 10,000 hours (14 months).

5.7.20 MCSS equipment shall be installed in such a manner so as to permit the replacement of faulty LRU's without cutting, desoldering, unwrapping, or the use of other techniques requiring more disassembly than the removal of screws and connectors.

And defect type 2

or defect type 2

5.9.9 The MCSS shall be delivered with any unique maintenance tools and support devices required by the equipment.

And defect type 2

any defect type 3

5.7.22 The MCSS shall be delivered with a 1-year warranty for parts and labor in the MSOCC, and on-call service (onsite response time within 4 hours of the time of the call) during the prime shift (0800 to 1700 Eastern time).

and defect type 2 2 times

5.7.23 The period of the warrantee shall begin following acceptance of the MCSS in the MSOCC.

5.7.3 The MTBF for each LCT shall not be less than 3000 hours.

5.7.4 The MCSS shall have an MTTR of less than 30 minutes.

5.7.5 The MCSS shall have at least four external control paths.

5.7.6 The MCSS shall have redundant control paths to each of the two MSOCC DOCS (four paths).

5.7.7 The MCSS shall contain redundant switch control units.

5.7.8 The MCSS shall contain two independent, isolated power supplies.

5.7.9 Each power supply shall have a separate connection to the AC power lines.

5.8 SAFETY

missing "shall" defect type 4

5.9.9 The MCSS shall contain interlocks and/or other devices and methods to eliminate exposure to any voltage in excess of 30 V Root Mean Square (RMS).

And defect type 2

any defect type 3

5.9.9 The MCSS shall contain mechanical protection to prevent people from coming into contact with moving parts such as gears, fans, and belts.

And defect type 2

such as defect type 3

5.9.9 The MCSS shall have a master power switch that shall be able to disconnect all AC feeds to the MCSS.

All defect type 3

Shall defect type 1

5.9 FACILITIES

missing "shall" defect type 4

5.9.1 The MCSS shall be installed in room E239, Building 14, at Goddard Space Flight Center.

5.9.10 The MCSS shall be packaged in EIA 19-inch horizontal mounting width racks.

5.9.11 The MCSS shall occupy no more floor space than four racks, 30 inches deep, 24 inches wide, and 78 inches tall.

And defect type 2

5.9.12 The racks holding the MCSS shall be painted Federal color standard 595a-20372 using vinyl textured paint.

5.9.13 The MCSS racks shall be freestanding.

5.9.14 The MCSS racks shall have retractable casters.

5.9.15 The weight distribution of the MCSS equipment racks shall be limited to 250 pounds per square foot of floor space.

5.9.16 The weight of a single rack containing equipment shall not exceed 1000 pounds.

5.9.17 The MCSS racks shall be capable of being bolted together side by side in a straight line.

5.9.18 The MCSS shall be capable of being easily separated into single-rack units for shipping purposes only.

5.9.19 Each MCSS rack shall contain a bus bar connected to the MSOCC ground, as shown in Figure 5-2.

5.9.2 The MCSS shall only use main power at 120 VAC +/- 10 percent single-phase 60 Hz (57 to 63).

5.9.20 The grounding of equipment in a rack to the bus bar shall be in accordance with STDN SPEC-7.

5.9.21 Access to the equipment in the racks shall be from the front and back of the cabinet.

And defect type 2

5.9.22 All rack-mounted equipment shall be uniquely labeled and serialized.

And defect type 2

5.9.23 Every interrack cable within the MCSS shall be uniquely labeled.

5.9.24.1 The interrack cable label shall contain the connector source

5.9.24.2 The interrack cable label shall contain the connector destination

5.9.24.3 The interrack cable label shall contain the cable number.

5.9.25 The interrack cable label shall be placed on each end of the cable, approximately 3 inches from the connector.

5.9.26 All external cabling shall enter the rack from under the floor in an area beginning not less than 2 inches from the back surface of the rack and extending toward the front, but not exceeding 12 inches from the back surface of the rack.

And defect type 2

5.9.27 The MCSS shall be able to operate using ambient air cooling.

5.9.28 The MCSS shall be able to operate using under-floor plenum cooling air between 65 and 80 degrees Fahrenheit; relative humidity will be between 35 and 70 percent noncondensing.

And defect type 2 2 times

will defect type 4

5.9.29 The temperature of the air exhaust from the MCSS rack shall be within 10 degrees Fahrenheit of air being input.

5.9.3 The MCSS shall use two separate independent AC power cables.

5.9.30 The MCSS shall not generate more than 70 dBA of noise at a distance of 6 feet in any direction.

Any defect type 3

5.9.31 The operation of the MCSS shall not emit electromagnetic interference that will interfere with the operation of equipment already installed in the MSOCC.

Will defect type 4

5.9.4 Each MCSS AC power cable shall be terminated in a male twist-lock connector.

5.9.5 Each MCSS AC power cable shall be at least 10 feet long.

5.9.6 The MCSS shall utilize no more than 2 kilovolt-amperes (kVa) of power, based on 120-VAC, using a power factor of 0.66.

5.9.7 Power transients of a +/-10 percent change from the nominal voltage lasting for 2 seconds shall not interfere with MCSS operation during and immediately following the transient period.

And defect type 2

5.9.8 The MCSS shall not be damaged by short-duration, high-amplitude transients of as much as +/-50 percent change from nominal voltage for a period of 1 millisecond per line cycle.

5.9.9 The MCSS shall not be damaged by a sudden loss of power or prolonged transients of the kind mentioned in paragraphs 5.9.7 and 5.9.8 on the power supply line.

And defect type 2

or defect type 2

4. SREE's output for MCSS RS

This section describes SREE's analysis during first-time compilation on New Adelaide Airport RS. The output shows the kinds of potential ambiguities found in each RStat. The user of SREE validates SREE's analysis and discovers some of the unidentified ambiguities. SREE is unable to recognise the unidentified potentially ambiguous words because they are not defined in SREE's AIC.

```
SECTION 5
```

```
MCSS REQUIREMENTS

This section addresses MCSS requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that although the statement is not an RStat. SREE recognises addresses to be a potentially ambiguous indicator of the *Plural* corpus, but addresses is not an ambiguous plural noun because addresses is a singular verb and not a plural noun in the statement.

```
5.1   MCSS SYSTEM DESCRIPTION.

The MCSS will function as the front end to the MSOCC.  The MCSS is functionally composed of a number of individual subsystems, as shown in Figure 5-1.  These functional subsystems are as follows:

is potentially ambiguous (DIRECTIVE) because of wording: figure
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: as follows
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: these
```

Potential ambiguity with value of: p = 5, n = 5, m = 5.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/5 * 100% = 100%**

SREE's False Positives = **p - m = 5 – 5 = 0**

The user determined that subsystems is potentially ambiguous. However, SREE doesn't recognise subsystems to be potentially ambiguous because requirements isn't defined as an indicator of potential ambiguity in the AIC.

a. Switch matrix.

b. Switch control.

c. Timing generator.

d. Test and monitoring.

is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

Even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in RStat's list item.

The functional requirements are grouped by subsystems. General MCSS requirements associated with each subsystem are included in the section dealing with that subsystem to facilitate locating specific requirements.

is potentially ambiguous (VAGUE) because of wording: general
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: that

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

The user determined that requirements and subsystems are potentially ambiguous instances. However, SREE doesn't recognise requirements and subsystems to be potentially ambiguous because requirements and subsystems aren't defined as indicators of potential ambiguity in the AIC. Furthermore, the statement is not an RStat.

5.2 SWITCH MATRIX SUBSYSTEM

The MCSS will establish signal paths for digital communications between the inputs and outputs of 255 full-duplex ports. A signal path consists of both the data signal and the timing signal routed as a pair from input to output. Each port will contain a pair of inputs and outputs. The MCSS will be able to loopback any data pattern being sent from the SEND data lines back to the RECEIVE data lines of the same port.

```
is potentially ambiguous (PLURALNOUN) because of wording:
communications
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (PLURALNOUN) because of wording: lines
is potentially ambiguous (PLURALNOUN) because of wording: paths
is potentially ambiguous (PLURALNOUN) because of wording: ports
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 9, n = 9, m = 5.

SREE's Recall = **p/n * 100% = 9/9 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/9 * 100% = 56%**

SREE's False Positives = **p - m = 9 – 5 = 4**

The user determined that paths, communications, inputs, and lines, are not instances of potential ambiguity as identified by SREE. SREE recognises paths, communications, inputs, and lines as potentially ambiguous instances because they are defined as potential ambiguity indicators in the AIC.

```
This section addresses the signal path requirements of the MCSS as well as those of this subsystem.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (VAGUE) because of wording: as well as
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: those
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that although the statement is not an RStat, SREE recognises addresses as potentially ambiguous. However, addresses is not ambiguous because addresses is a singular verb instead of a plural noun..

```
5.2.1 The MCSS shall contain 255 full duplex ports.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: ports
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises ports to be potentially ambiguous as defined in the AIC, ports does not contribute to ambiguity in the 5.2.1 RStat.

```
5.2.2 The MCSS shall switch binary digital signals.

is potentially ambiguous (PLURALNOUN) because of wording: signals
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.2.2 RStat.

```
5.2.3 The MCSS shall switch serial signals.

is potentially ambiguous (PLURALNOUN) because of wording: signals
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.2.3 RStat.

```
5.2.4 This subsystem shall switch data and timing in the same direction as signal pairs.

is potentially ambiguous (PLURALNOUN) because of wording: pairs
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises pairs to be potentially ambiguous as defined in the AIC, pairs does not contribute to ambiguity in the 5.2.4 RStat.

```
5.2.5 The phase relationship between data and timing signals shall be maintained in each signal pair passing through the MCSS such that the variance shall not be greater than 5 percent.

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises that to be potentially ambiguous as defined in the AIC, that does not contribute to ambiguity in the 5.2.5 RStat.

```
5.2.6 The MCSS shall connect specific inputs to specific outputs.

is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (VAGUE) because of wording: specific
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that outputs is potentially ambiguous. However, SREE doesn't recognise outputs to be potentially ambiguous because outputs isn't defined as a potential ambiguity indicator in the AIC.

```
5.2.7 The MCSS shall disconnect specific inputs from specific outputs.

is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (VAGUE) because of wording: specific
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that outputs is potentially ambiguous. However, SREE doesn't recognise outputs to be potentially ambiguous because outputs isn't defined as an potential ambiguity indicator in the AIC.

```
5.2.8 This subsystem shall establish connections between 255 inputs
and 255 outputs.

is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises inputs and and to be potentially ambiguous instances as defined in the AIC, inputs and and do not contribute to ambiguity in the 5.2.8 RStat.

```
5.2.9 This subsystem shall establish connections between any one input
and any one of the 255 outputs.

is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 5.2.9 RStat.

```
5.2.10 This subsystem shall provide the capability to connect any one
input to up to a minimum of 32 outputs simultaneously.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: capability to
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.
SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**
SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**
SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises minimum to be potentially ambiguous as defined in the AIC, minimum does not contribute to ambiguity in the 5.2.2 RStat.

```
5.2.11 This subsystem shall not allow two or more inputs to be connected to the same output simultaneously.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (VAGUE) because of wording: more
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.
SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**
SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**
SREE's False Positives = **p - m = 4 – 4 = 0**

```
5.2.12 This subsystem shall not allow data to appear on an output when that output is not connected to any input.
```

```
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 3.
SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**
SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**
SREE's False Positives = **p - m = 3 – 3 = 0**

5.2.13 This subsystem shall place a mark state on every output when that output is not connected to any input.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (PRONOUN) because of wording: this

Potential Ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.2.14 The signal delay between any input port and its output port through the MCSS shall be less than 1 millisecond.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: less
is potentially ambiguous (PRONOUN) because of wording: its
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises less and and to be potentially ambiguous instances as defined in the AIC, less and and do not contribute to ambiguity in the 5.2.14 RStat.

5.2.15 The delay on any signal path through the MCSS shall be within 10 percent of the delay on any other signal path.

is potentially ambiguous (QUANTIFIER) because of wording: any

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that other is an ambiguous adjective. However, SREE doesn't recognise other to be potentially ambiguous because other isn't defined as a potential ambiguity indicator in the AIC.

```
5.2.16 The delay between input and output ports through the MCSS shall be independent of the data rate.

is potentially ambiguous (PLURALNOUN) because of wording: ports
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.2.17 The crosstalk between signals passing through the MCSS shall be less than -70 decibels (dB).

is potentially ambiguous (PLURALNOUN) because of wording: decibels
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (db)
is potentially ambiguous (VAGUE) because of wording: less
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 0**

The user determined that even though SREE recognises less and decibels to be potentially ambiguous instances as defined in the AIC, but less and decibels do not contribute to ambiguity in the 5.2.17 RStat.

```
5.2.18  The added error to signals switched through the MCSS shall be
less than 1 in 108 bits.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: bits
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: less
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33.3%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises less and bits to be potentially ambiguous as defined in the AIC, but less and bits do not contribute to ambiguity in the 5.2.18 RStat.

```
5.2.19  The frequency of each signal through the MCSS shall be between
0 and 6.312 MHz.
```

```
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 5.2.19 RStat.

5.2.20 Each signal path through the MCSS shall be capable of passing data at 0 to 6.312 Mbps.

is potentially ambiguous (VAGUE) because of wording: capable of

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.2.21 The MCSS shall be capable of passing data at up to 6.312 Mbps on all signal paths simultaneously.

is potentially ambiguous (PLURALNOUN) because of wording: paths
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: capable of

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.2.22 The MCSS interface to external equipment/data lines shall be Data Communication Equipment (DCE) using DB-15S connectors.

is potentially ambiguous (PLURALNOUN) because of wording: connectors
is potentially ambiguous (PLURALNOUN) because of wording: lines
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: (dce)

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises connectors to be potentially ambiguous as defined in the AIC, connectors does not contribute to ambiguity in the 5.2.22 RStat.

```
5.2.23 Each external data/timing port interface to the MCSS shall
support both RS-422-A and RS-423-A.

is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: both
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 5.2.23 RStat.

```
5.2.24 The MCSS shall visually indicate that a specific port is
configured as RS-422-A or RS-423-A.

is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises that to be potentially ambiguous as defined in the AIC, that does not contribute to ambiguity in the 5.2.24 RStat.

5.2.25 The MCSS shall contain five Nascom-supplied line driver chassis.

is potentially ambiguous (PLURALNOUN) because of wording: chassis

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises chassis to be potentially ambiguous as defined in the AIC, chassis does not contribute to ambiguity in the 5.2.25 RStat.

5.2.26 The MCSS shall interface with the Nascom-supplied line drivers.

is potentially ambiguous (PLURALNOUN) because of wording: drivers

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises drivers to be potentially ambiguous as defined in the AIC, drivers does not contribute to ambiguity in the 5.2.26 RStat.

```
5.2.27 The MCSS shall provide patch panel capability for each of the Nascom lines.
is potentially ambiguous (PLURALNOUN) because of wording: lines
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises lines to be potentially ambiguous as defined in the AIC, lines does not contribute to ambiguity in the 5.2.27 RStat.

```
5.2.28 Thirty-two sets of bypass (patch) cables to enable the MCSS to be bypassed shall be provided.
is potentially ambiguous (PLURALNOUN) because of wording: cables
is potentially ambiguous (PLURALNOUN) because of wording: sets
is potentially ambiguous (VAGUE) because of wording: (patch)
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises sets and cables to be potentially ambiguous instances as defined in the AIC, sets and cables do not contribute to ambiguity in the 5.2.28 RStat.

```
5.2.28.1 This bypass capability shall be on both Nascom-to-MSOCC and MSOCC-to-MSOCC connections.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises connections to be potentially ambiguous as defined in the AIC, connections does not contribute to ambiguity in the 5.2.28.1 RStat.

5.3 SWITCH CONTROL SUBSYSTEM

The switch control subsystem will be the interface between the operators and the switch matrix, timing generator, and the test and monitoring subsystems. This subsystem will contain the switch control units and two LCT's, which will be used in troubleshooting and as a control backup. One of the LCT's will be located in the DOCS room.

```
is potentially ambiguous (PLURALNOUN) because of wording: operators
is potentially ambiguous (PLURALNOUN) because of wording: units
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: which
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 4.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/6 * 100% = 67%**

SREE's False Positives = **p - m = 6 – 4 = 2**

The user determined that even though SREE recognises operators and units to be instances of potential ambiguity as defined in the AIC, however operators and units do not contribute to ambiguity in the 5.3 RStat.

```
This section addresses the command and status (control) requirements of the MCSS as well as those of this subsystem.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (VAGUE) because of wording: (control)
is potentially ambiguous (VAGUE) because of wording: as well as
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: those
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 5.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/6 * 100% = 83%**

SREE's False Positives = **p - m = 6 – 5 = 1**

The user determined that although the statement is not an RStat, the instance addresses is not ambiguous because addresses is a singular verb and not a plural noun.

```
5.3.1 This subsystem shall be the MCSS operator's control interface.
```

```
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.3.2       This subsystem shall allow a user with designated
privileges to establish and modify passwords for access to this
subsystem.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: passwords
is potentially ambiguous (PLURALNOUN) because of wording: privileges
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises passwords and privileges to be potentially ambiguous instances as defined in the AIC, passwords and privileges do not contribute to ambiguity in the 5.3.2 RStat. However, designated is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
5.3.3       This subsystem shall require the use of passwords to set
up a session to access configuration tables and files.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: files
is potentially ambiguous (PLURALNOUN) because of wording: passwords
is potentially ambiguous (PLURALNOUN) because of wording: tables
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 2.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/5 * 100% = 40%**

SREE's False Positives = **p - m = 5 – 2 = 3**

The user determined that even though SREE recognises passwords, tables, and files, to be potentially ambiguous instances as defined in the AIC, passwords, tables, and files, do not contribute to ambiguity in the 5.3.3 RStat.

```
5.3.4       This subsystem shall require the use of passwords to set
up a session to control, assign, or modify switch connections.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: passwords
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises connections and passwords to be potentially ambiguous instances as defined in the AIC, connections and passwords do not contribute to ambiguity in the 5.3.4 RStat.

```
5.3.5       This subsystem shall identify each port with a unique
mnemonic.
```

```
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that unique is an ambiguous adjective. However, SREE doesn't recognise unique to be potentially ambiguous because unique isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.6       Each port mnemonic shall be user definable.
```

The user determined that definable is a potentially ambiguous adjective. However, SREE doesn't recognise definable to be potentially ambiguous because definable isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.7       This subsystem shall allow each port mnemonic to contain
at least 8 characters.
is potentially ambiguous (PLURALNOUN) because of wording: characters
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises characters to be potentially ambiguous as defined in the AIC, characters does not contribute to ambiguity in the 5.3.7 RStat.

```
5.3.8       This subsystem shall assign a standard default mnemonic
for a port if a mnemonic has not been predefined for the port.
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that default and predefined are instances of potential ambiguity. However, SREE doesn't recognise default and predefined to be potentially ambiguous because default and predefined aren't defined as potential ambiguity indicators in the AIC.

```
5.3.9       The assigned default mnemonic shall be in the form DTExxx or DCExxx, where xxx is the port number and identifies the port mnemonic assigned.
```

```
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that default is potentially ambiguous, but SREE doesn't recognise default as a potential ambiguity indicator because default is not defined as a potential ambiguity indicator in the AIC.

```
5.3.10      This subsystem shall only allow the renaming of a switch port mnemonic for a port that is not connected at the time.
```

```
is potentially ambiguous (VAGUE) because of wording: only
is potentially ambiguous (PRONOUN) because of wording: that
```

```
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises that to be potentially ambiguous as defined in the AIC, that does not contribute to ambiguity in the 5.3.10 RStat.

```
5.3.11      This subsystem shall provide the capability to allow the
control interface to select the active switch control unit.

is potentially ambiguous (VAGUE) because of wording: capability to
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.3.12      This subsystem shall contain two identical LCT's.

is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that identical is an ambiguous adjective. However, SREE doesn't recognise identical to be potentially ambiguous because identical isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.13      The LCT display shall have a diagonal measurement of at
least 19 inches.
is potentially ambiguous (PLURALNOUN) because of wording: inches
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises inches to be potentially ambiguous as defined in the AIC, inches does not contribute to ambiguity in the 5.3.13 RStat.

```
5.3.14      The LCT display shall be capable of displaying at least 16
colors.
is potentially ambiguous (PLURALNOUN) because of wording: colors
is potentially ambiguous (VAGUE) because of wording: capable of
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises colors to be potentially ambiguous as defined in the AIC, colors does not contribute to ambiguity in the 5.3.14 RStat.

5.3.15 The LCT display shall have a resolution of at least 640 horizontal pixels and 480 vertical pixels.

is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 5.3.15 RStat.

5.3.16 The information on the LCT display screen shall have colors and presentation characteristics for graphics and text usage similar to the DOCS switch configuration display screens.

is potentially ambiguous (PLURALNOUN) because of wording: colors
is potentially ambiguous (PLURALNOUN) because of wording: graphics
is potentially ambiguous (PLURALNOUN) because of wording: screens
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that graphics and screens are not potentially ambiguous instances but characteristics and similar are potentially ambiguous instances. However, SREE doesn't recognise characteristics and similar to be potentially ambiguous because characteristics and similar aren't defined as potential ambiguity indicators in the AIC.

```
5.3.17      The LCT operator interface shall be functionally similar
to the DOCS operator interface for switch-related functions.
is potentially ambiguous (PLURALNOUN) because of wording: functions
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that functions is not potentially ambiguous but similar is a potentially ambiguous adjective. However, SREE doesn't recognise similar to be potentially ambiguous because similar isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.18      Keyboard entries on the LCT shall be displayed on its
screen within 0.5 seconds of keystroke entry.
is potentially ambiguous (PLURALNOUN) because of wording: entries
is potentially ambiguous (PRONOUN) because of wording: its
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.3.19      The LCT shall have the capability to display all port
mnemonics.
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: capability to
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that mnemonics is potentially ambiguous. However, SREE doesn't recognise mnemonics to be potentially ambiguous because mnemonics isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.20      The LCT display of the switch configuration shall be identical to the actual switch configuration.
```

The user determined that identical is an ambiguous adjective. However, SREE doesn't recognise identical to be potentially ambiguous because identical isn't defined as a potential ambiguity indicator in the AIC.

```
5.3.21      The LCT shall update the switch status display within 2 seconds of a change in the switch configuration.
5.3.22      The LCT shall make and break connections between sets of inputs and outputs as identified in a command file.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (PLURALNOUN) because of wording: sets
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 1.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/4 * 100% = 25%**

SREE's False Positives = **p - m = 4 – 1 = 3**

The user determined that even though SREE recognises connections, sets, and inputs, to be potentially ambiguous instances as defined in the AIC, connections, sets, and inputs, do not contribute to ambiguity in the 5.3.22 RStat.

```
5.3.23      This subsystem shall process control messages from any
control interface.

is potentially ambiguous (PLURALNOUN) because of wording: messages
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises messages to be potentially ambiguous as defined in the AIC, messages does not contribute to ambiguity in the 5.3.23 RStat.

```
5.3.24      This subsystem shall process control messages in the order
in which they are received.

is potentially ambiguous (PLURALNOUN) because of wording: messages
is potentially ambiguous (PRONOUN) because of wording: they
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: which
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises messages to be potentially ambiguous as defined in the AIC, messages does not contribute to ambiguity in the 5.3.24 RStat.

```
5.3.25      This subsystem shall acknowledge control messages only to
the interface originating the control message.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: messages
is potentially ambiguous (VAGUE) because of wording: only
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises messages and only to be potentially ambiguous instances as defined in the AIC, messages and only do not contribute to ambiguity in the 5.3.25 RStat.

```
5.3.26      This subsystem shall communicate with the control
interface only in response to a control message.
```

```
is potentially ambiguous (VAGUE) because of wording: only
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises only to be potentially ambiguous as defined in the AIC, only does not contribute to ambiguity in the 5.3.26 RStat.

`5.3.27 This subsystem will identify at least 11 control message errors.`

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises messages to be potentially ambiguous as defined in the AIC, messages does not contribute to ambiguity in the 5.3.27 RStat.

`5.3.27.1 One of the 11 control message errors for this subsystem shall be 'Invalid American Standard Code for Information Interchange (ASCII) digit'.`

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (VAGUE) because of wording: (ascii)
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.1 RStat.

```
5.3.27.2    One of the 11 control message errors for this subsystem
shall be 'Invalid latch or host number'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.2 RStat.

```
5.3.27.3    One of the 11 control message errors for this subsystem
shall be 'Invalid latch setting'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.3 RStat.

```
5.3.27.4    One of the 11 control message errors for this subsystem
shall be 'Invalid connection'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.4 RStat.

```
5.3.27.5    One of the 11 control message errors for this subsystem
shall be 'Invalid message length'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.5 RStat.

```
5.3.27.6    One of the 11 control message errors for this subsystem
shall be 'Invalid message type'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.6 RStat.

```
5.3.27.7    One of the 11 control message errors for this subsystem
shall be 'Latch failure'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.7 RStat.

```
5.3.27.8    One of the 11 control message errors for this subsystem
shall be 'Latch not installed'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.8 RStat.

```
5.3.27.9    One of the 11 control message errors for this subsystem
shall be 'Latch already connected'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.9 RStat.

```
5.3.27.10   One of the 11 control message errors for this subsystem
shall be 'Unable to communicate to control interface RS-232-C port'.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.10 RStat.

```
5.3.27.11   One of the 11 control message errors for this subsystem
shall be 'Parity error on the control interface'.

is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.3.27.11 RStat.

```
5.3.28      This subsystem shall report any error status to the
control interface originating the control message.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.3.29      This subsystem shall not attempt to execute any control
message containing an error.
```

```
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.3.30      This subsystem shall respond to DOCS status requests once
every 5 seconds, in accordance with the Interface Control Document
(ICD) Between MSOCC DOCS and MSOCC Automated Switching Systems,
CSC/TM-83/6105UDI, April 1989.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: systems
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: (icd)
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that even though SREE recognises systems and / to be potentially ambiguous instances as defined in the AIC, however both systems and / do not contribute to ambiguity in the 5.3.30 RStat.

```
5.3.31      This subsystem shall respond to commands within 10 seconds
of power-up reset.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: commands
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises commands to be potentially ambiguous as defined in the AIC, commands does not contribute to ambiguity in the 5.3.31 RStat.

```
5.3.32      This subsystem shall execute and acknowledge commands
within 0.5 seconds of receipt of a valid command.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: commands
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises commands to be potentially ambiguous as defined in the AIC, commands does not contribute to ambiguity in the 5.3.32 RStat.

```
5.3.33      This subsystem shall not permit connection commands for a
specific connection to disrupt any existing connections.

is potentially ambiguous (PLURALNOUN) because of wording: commands
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises commands to be potentially ambiguous as defined in the AIC, commands does not contribute to ambiguity in the 5.3.33 RStat.

```
5.3.34      This subsystem shall not permit disconnection commands for
a specific connection to disrupt any other existing connections.

is potentially ambiguous (PLURALNOUN) because of wording: commands
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises commands to be potentially ambiguous as defined in the AIC, commands does not contribute to ambiguity in the 5.3.27.3 RStat.

```
5.3.35      This subsystem shall not set priorities for connections.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: priorities
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises connections and priorities to be potentially ambiguous instances as defined in the AIC, connections and priorities do not contribute to ambiguity in the 5.3.35 RStat.

```
5.3.36      This subsystem shall respond to a status request within
0.5 seconds of receipt of the request.
```

```
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.3.37      This subsystem shall supply the status of the switch
control units.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: units
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.3.38 The MCSS shall provide status information for each port.

5.3.38.1 The status information for each port shall indicate the Port is functional (hardware exists).

is potentially ambiguous (VAGUE) because of wording: (hardware exists)

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.3.38.2 The status information for each port shall indicate the Port is nonfunctional (hardware does not exist or failed to operate properly).

is potentially ambiguous (VAGUE) because of wording: (hardware does not exist or failed to operate properly)
is potentially ambiguous (VAGUE) because of wording: exist
is potentially ambiguous (COORDINATOR) because of wording: or

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.3.39 This subsystem shall supply status information indicating which outputs are connected to a specific inputs.

```
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: which
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.
SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**
SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**
SREE's False Positives = **p - m = 4 – 4 = 0**

The user determined that outputs is potentially ambiguous. However, SREE doesn't recognise outputs to be potentially ambiguous because outputs isn't defined as a potential ambiguity indicator in the AIC.

5.3.40 This subsystem shall supply status information indicating which inputs are connected to specific outputs.

```
is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (VAGUE) because of wording: specific
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: which
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.
SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**
SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**
SREE's False Positives = **p - m = 4 – 4 = 0**

The user determined that outputs is potentially ambiguous. However, SREE doesn't recognise outputs to be potentially ambiguous because outputs isn't defined as indicator of potential ambiguity in the AIC.

`5.3.41 This subsystem shall perform two kinds of diagnostics: background and troubleshooting.`

```
is potentially ambiguous (PLURALNOUN) because of wording: kinds
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that diagnostics is potentially ambiguous, but kinds and and are not potentially ambiguous instances. However, SREE doesn't recognise diagnostics to be potentially ambiguous because diagnostics isn't defined as a potential ambiguity indicator in the AIC.

`5.3.42 The LCT shall be able to display a menu of available diagnostic tests, with a brief description of their purpose.`

```
is potentially ambiguous (PLURALNOUN) because of wording: tests
is potentially ambiguous (PRONOUN) because of wording: their
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that available is an ambiguous adjective, but tests is not an ambiguous plural noun. However, SREE doesn't recognise available to be potentially ambiguous because available isn't defined as a potential ambiguity indicator in the AIC.

`5.3.43 A particular diagnostic test shall be activated by the LCT.`

`is potentially ambiguous (VAGUE) because of wording: particular`

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

`5.3.44 A particular diagnostic test shall be aborted by the LCT.`

`is potentially ambiguous (VAGUE) because of wording: particular`

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

`5.3.45 The LCT shall be able to display the current status of active diagnostic tests.`

`is potentially ambiguous (PLURALNOUN) because of wording: tests`

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises tests to be potentially ambiguous as defined in the AIC, tests does not contribute to ambiguity in the 5.3.45 RStat.

`5.3.46 This subsystem shall accumulate all diagnostic test results.`

```
is potentially ambiguous (PLURALNOUN) because of wording: results
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises results to be potentially ambiguous as defined in the AIC, results does not contribute to ambiguity in the 5.3.46 RStat.

`5.3.47 All accumulated diagnostic test results shall be viewable through the LCT.`

```
is potentially ambiguous (PLURALNOUN) because of wording: results
is potentially ambiguous (QUANTIFIER) because of wording: all
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

`5.3.48 This subsystem shall only execute troubleshooting diagnostics under LCT control.`

```
is potentially ambiguous (VAGUE) because of wording: only
```

is potentially ambiguous (PRONOUN) because of wording: this

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.3.49 The LCT shall limit access to troubleshooting diagnostic tests by means of a password protection scheme.

is potentially ambiguous (PLURALNOUN) because of wording: means
is potentially ambiguous (PLURALNOUN) because of wording: tests

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises tests to be potentially ambiguous as defined in the AIC, tests does not contribute to ambiguity in the 5.3.49 RStat.

5.3.50 This subsystem shall require confirmation by the operator before altering the switch configuration when performing troubleshooting diagnostics.

is potentially ambiguous (PRONOUN) because of wording: this

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

`5.3.51 The MCSS shall perform a background diagnostic self-test at least once per hour, or on command from the LCT, to determine the proper operation of all unused circuits.`

```
is potentially ambiguous (PLURALNOUN) because of wording: circuits
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises circuits to be potentially ambiguous as defined in the AIC, circuits does not contribute to ambiguity in the 5.3.51 RStat.

`5.3.52 The MCSS shall be able to execute background diagnostic tests without affecting the operational control of the MCSS from the DOCS.`

```
is potentially ambiguous (PLURALNOUN) because of wording: tests
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises tests to be potentially ambiguous as defined in the AIC, tests does not contribute to ambiguity in the 5.3.52 RStat.

5.3.53 The MCSS diagnostic tests shall have the capability to determine a failure down to the LRU level.

is potentially ambiguous (PLURALNOUN) because of wording: tests
is potentially ambiguous (VAGUE) because of wording: capability to

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.3.54 Background diagnostics tests shall not change any existing switch connections.

is potentially ambiguous (PLURALNOUN) because of wording: connections
is potentially ambiguous (PLURALNOUN) because of wording: tests
is potentially ambiguous (QUANTIFIER) because of wording: any

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.3.55 The MCSS shall reject control messages from any external control interface when performing troubleshooting diagnostics.

is potentially ambiguous (PLURALNOUN) because of wording: messages
is potentially ambiguous (QUANTIFIER) because of wording: any

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises messages to be potentially ambiguous as defined in the AIC, messages does not contribute to ambiguity in the 5.3.55 RStat.

```
5.3.56      This subsystem shall display hardware error messages on
the LCT upon detection.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: messages
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises messages to be potentially ambiguous as defined in the AIC, messages does not contribute to ambiguity in the 5.3.56 RStat.

```
5.3.57      The MCSS control and status interface shall be full duplex
RS-232-C compliant for the four EIA signals.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.57 RStat.

```
5.3.57.1 One of the four EIA signals shall be Transmitted data (BA)

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (ba)
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.57.1 RStat.

```
5.3.57.2 One of the four EIA signals shall be Received data (BB)

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (bb)
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.57.2 RStat.

```
5.3.57.3 One of the four EIA signals shall be Protective ground (AA)
```

```
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (aa)
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.57.3 RStat.

```
5.3.57.4 One of the four EIA signals shall be Signal ground (AB).
```

```
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (ab)
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.57.4 RStat.

```
5.3.58 The MCSS shall communicate with the DOCS at discrete selectable baud rates, including 9600 and 19200.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: rates
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

```
5.3.59      The MCSS control and status interface data format shall be
1 start bit, 7 ASCII data bits, odd parity bit and 2 stop bits.
is potentially ambiguous (PLURALNOUN) because of wording: bits
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises bits to be potentially ambiguous as defined in the AIC, bits does not contribute to ambiguity in the 5.3.59 RStat.

```
5.3.60      The MCSS control and status interface shall be DTE using
DB-25-S connectors.
is potentially ambiguous (PLURALNOUN) because of wording: connectors
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises connectors to be potentially ambiguous as defined in the AIC, connectors does not contribute to ambiguity in the 5.3.60 RStat.

```
5.3.61      An MCSS power-up reset shall clear all hardware and
software registers and memory, and shall initialize all outputs to a
disconnected and operational state within 10 seconds.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: registers
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises registers to be potentially ambiguous as defined in the AIC, registers does not contribute to ambiguity in the 5.3.61 RStat.

```
5.3.62      This subsystem shall perform a self-test on the MCSS upon
power up.  The self-test shall, as a minimum, verify the operational
status of all controllers and perform a memory check on all RAM memory.
```

```
is potentially ambiguous (INCOMPLETES) because of wording: as a
minimum
is potentially ambiguous (PLURALNOUN) because of wording: controllers
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 4.
SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**
SREE's Precision = **m/p * 100% = 4/6 * 100% = 67%**
SREE's False Positives = **p - m = 6 – 4 = 2**

The user determined that even though SREE recognises minimum and controllers to be potentially ambiguous instances as defined in the AIC, minimum and controllers do not contribute to ambiguity, but as a minimum keyphrase should be identified as potentially ambiguous in the 5.3.62 RStat.

5.3.63 An MCSS soft reset shall reset and clear all controlling mechanisms used on the switch and bring them to a predetermined state.

```
is potentially ambiguous (PLURALNOUN) because of wording: mechanisms
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (PRONOUN) because of wording: them
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.
SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**
SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**
SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that predetermined is an ambiguous adjective, but mechanisms is not potentially ambiguous plural noun. SREE doesn't recognise predetermined to be potentially ambiguous because predetermined isn't defined as a potential ambiguity indicator in the AIC.

5.3.64 An MCSS soft reset shall not affect data/timing signals being routed through the switch.

```
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.64 RStat.

```
5.3.65      Actions initiated by an MCSS soft reset shall be completed
within 10 seconds.
is potentially ambiguous (PLURALNOUN) because of wording: actions
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**
SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.3.66      The LCT shall be capable of initiating a soft reset.
is potentially ambiguous (VAGUE) because of wording: capable of
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**
SREE's False Positives = **p - m = 1 – 1 = 0**

5.3.67 A front panel control shall be capable of initiating a soft reset.

is potentially ambiguous (VAGUE) because of wording: capable of

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.3.68.1 The LCT shall generate RS-170-A video signals as separate red, green, and blue (RGB) signals

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: (rgb)
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.68.1 RStat.

5.3.68.2 The LCT shall generate RS-170-A video signals as negative synchronization pulses on the green signal.

is potentially ambiguous (PLURALNOUN) because of wording: pulses
is potentially ambiguous (PLURALNOUN) because of wording: signals

Potential ambiguity with value of: p = 2, n = 2, m = 1.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**
SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.3.68.2 RStat.

```
5.4   TIMING GENERATOR SUBSYSTEM
The MSOCC Transition Plan requires all equipment installed in the future to generate their own timing signals.  In the meantime, the MCSS will provide timing signals for all equipment within the MSOCC that require this capability.
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (CONTINUANCE) because of wording: meantime
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (PRONOUN) because of wording: their
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 7, n = 7, m = 6.
SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**
SREE's Precision = **m/p * 100% = 6/7 * 100% = 86%**
SREE's False Positives = **p - m = 7 – 6 = 1**

The user determined that future is an ambiguous adjective, but signals is not an ambiguous plural noun. However, SREE doesn't recognise future to be potentially ambiguous because future isn't defined as a potential ambiguity indicator in the AIC.

The timing generator subsystem will consist of an equivalent to the current clock Buffer and a distribution capability for the timing signals it generates. The timing signal will be looped back at the data source in the MSOCC such that the timing signal accompanies the signal as an additional pair of wires in the same cable, to avoid any phase delay problems.

is potentially ambiguous (PLURALNOUN) because of wording: problems
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (PLURALNOUN) because of wording: wires
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: it
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 8, n = 8, m = 5.

SREE's Recall = **p/n * 100% = 8/8 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/8 * 100% = 63%**

SREE's False Positives = **p - m = 8 – 5 = 3**

The user determined that additional is an ambiguous adjective, but wires, signals, and that, are not instances of potential ambiguity. SREE doesn't recognise additional to be potentially ambiguous because additional isn't defined as a potential ambiguity indicator in the AIC.

When the MSOCC equipment requiring external timing is removed from the MSOCC, the timing generator subsystem will also be removed from the MCSS.

is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: also

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises also to be potentially ambiguous as defined in the AIC, also does not contribute to ambiguity in the RStat.

```
This section addresses the timing generator subsystem requirements of the MCSS, as well as those of this subsystem itself.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (VAGUE) because of wording: as well as
is potentially ambiguous (PRONOUN) because of wording: itself
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: those
```

The user determined that although the statement is not an RStat, SREE recognises addresses to be potentially ambiguous as defined in the AIC. addresses does not contribute to plural ambiguity because addresses is a singular verb and not a plural noun as caught.

```
5.4.1 This subsystem shall contain identical primary and backup ST signal generators.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: generators
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that identical is an ambiguous adjective, but generators is not an ambiguous plural noun. However, SREE doesn't recognise identical to be potentially ambiguous because identical isn't defined as an indicator of potential ambiguity in the AIC.

```
5.4.2 This subsystem shall generate at least the following frequencies
simultaneously:
```

```
is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: following
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

```
5.4.2.1     One of the frequencies shall be 9.6 kHz +/- 1 percent
```

```
is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.1 RStat.

```
5.4.2.2     One of the frequencies shall be 19.2 kHz +/- 1 percent
```

```
is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.2 RStat.

```
5.4.2.3     One of the frequencies shall be 56.0 kHz +/- 1 percent

is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.3 RStat.

```
5.4.2.4     One of the frequencies shall be 224.0 kHz +/- 1 percent

is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.4 RStat.

```
5.4.2.5     One of the frequencies shall be 1.544 MHz +/- 1 percent

is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.5 RStat.

```
5.4.2.6     One of the frequencies shall be 2.048 MHz +/- 1 percent

is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.6 RStat.

```
5.4.2.7     One of the frequencies shall be 6.312 MHz +/- 1 percent

is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises frequencies to be potentially ambiguous as defined in the AIC, frequencies does not contribute to ambiguity in the 5.4.2.7 RStat.

```
5.4.3 This subsystem shall be capable of accepting up to 10 external
timing signals from Nascom for simultaneous distribution as ST.

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: capable of
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.4.3 RStat.

```
5.4.4 This subsystem shall distribute ST signals to a minimum of 255 ports.

is potentially ambiguous (PLURALNOUN) because of wording: ports
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 4, n = 4, m = 1.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/4 * 100% = 25%**

SREE's False Positives = **p - m = 4 – 1 = 3**

The user determined that even though SREE recognises ports, signals, and minimum, to be potentially ambiguous as defined in the AIC, ports, signals, and minimum, do not contribute to ambiguity in the 5.4.4 RStat.

```
5.4.5  The MCSS ST voltage levels shall be RS-422-A compatible.

is potentially ambiguous (PLURALNOUN) because of wording: levels
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.5   TEST AND MONITORING SUBSYSTEM
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

This subsystem will contain BED's, redundant NBG's, and the electronics to couple both to the signal path under test. This subsystem also includes the patch panels used for manually troubleshooting and monitoring the Nascom lines. This subsystem will perform its functions without the use of patch panels.

```
is potentially ambiguous (PLURALNOUN) because of wording: functions
is potentially ambiguous (PLURALNOUN) because of wording: lines
is potentially ambiguous (PLURALNOUN) because of wording: panels
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: also
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (PRONOUN) because of wording: its
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 9, n = 9, m = 6.

SREE's Recall = **p/n * 100% = 9/9 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/9 * 100% = 67%**

SREE's False Positives = **p - m = 9 – 6 = 3**

The user determined that electronics is potentially ambiguous, but also, panels, and lines, do not contribute to ambiguity. SREE doesn't recognise electronics to be potentially ambiguous because electronics isn't defined as a potential ambiguity indicator in the AIC.

```
This section addresses the test and monitoring subsystem requirements of the MCSS as well as those of this subsystem.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (VAGUE) because of wording: as well as
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (PRONOUN) because of wording: those
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises addresses to be potentially ambiguous as defined in the AIC, addresses does not contribute to ambiguity in the statement.

```
5.5.1 This subsystem shall not affect the signals being monitored.

is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.5.1 RStat.

```
5.5.2 This subsystem shall not affect any signals not being monitored
and that are passing through the MCSS.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises signals to be potentially ambiguous as defined in the AIC, signals does not contribute to ambiguity in the 5.5.2 RStat.

```
5.5.3 This subsystem shall contain at least 10 NBG's.
is ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.5.4 The NBG shall be able to generate valid 4800-bit static Nascom
blocks at the timing frequencies identified in paragraph 5.4.2.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: blocks
is potentially ambiguous (PLURALNOUN) because of wording: frequencies
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises blocks to be potentially ambiguous as defined in the AIC, blocks does not contribute to ambiguity in the 5.5.4 RStat. However, valid is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

`5.5.5 The NBG shall be able to generate bad 4800-bit static Nascom blocks (with incorrect CRC) at the timing frequencies identified in paragraph 5.4.2.`

```
is potentially ambiguous (PLURALNOUN) because of wording: blocks
is potentially ambiguous (PLURALNOUN) because of wording: frequencies
is potentially ambiguous (VAGUE) because of wording: (with incorrect
crc)
is potentially ambiguous (VAGUE) because of wording: bad
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises blocks to be potentially ambiguous as defined in the AIC, blocks does not contribute to ambiguity in the 5.5.5 RStat.

```
5.5.6 The NBG output timing and signal voltage levels shall be capable
of being set to RS-422-A or RS-423-A compatible.

is potentially ambiguous (PLURALNOUN) because of wording: levels
is potentially ambiguous (VAGUE) because of wording: capable of
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

```
5.5.7 The MCSS shall pass blocks generated by the NBG and
corresponding timing signals as a signal pair through the switch.

is potentially ambiguous (PLURALNOUN) because of wording: blocks
is potentially ambiguous (PLURALNOUN) because of wording: signals
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises blocks and signals to be potentially ambiguous as defined in the AIC, blocks and signals do not contribute to ambiguity in the 5.5.7 RStat.

```
5.5.8 This subsystem shall contain at least 10 BED's.

is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.5.9 This subsystem shall allow the operator to connect an oscilloscope and/or a BED to any input or output signal port in a monitoring capacity.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
is potentially ambiguous (COORDINATOR) because of wording: and/or

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and should be considered as part of and/or and not as a single attribute.

5.5.10 The BED shall indicate the presence of data and timing.

is ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

`5.5.11 The BED shall perform a 22-bit CRC on the 4800-bit Nascom blocks passing through the matrix switch subsystem.`

`is ambiguous (PLURALNOUN) because of wording: blocks`

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises blocks to be potentially ambiguous as defined in the AIC, blocks does not contribute to ambiguity in the 5.5.11 RStat.

`5.5.12 The BED shall indicate the presence of CRC errors.`

`is potentially ambiguous (PLURALNOUN) because of wording: errors`

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises errors to be potentially ambiguous as defined in the AIC, errors does not contribute to ambiguity in the 5.5.12 RStat.

`5.5.13 The BED shall display the number of bad blocks detected.`

```
is potentially ambiguous (PLURALNOUN) because of wording: blocks
is potentially ambiguous (VAGUE) because of wording: bad
```

Potential ambiguity with value of: p = 2, n = 2, m = 1. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/2 * 100% = 50%** **SREE's False Positives** = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises blocks to be potentially ambiguous as defined in the AIC, blocks does not contribute to ambiguity in the 5.5.13 RStat.

```
5.5.14      The BED display shall be able to be reset to zero.
5.5.15      The BED shall increment a bad block counter each time it
identifies an error in a Nascom block.

is potentially ambiguous (VAGUE) because of wording: bad
is potentially ambiguous (PRONOUN) because of wording: it
```

Potential ambiguity with value of: p = 2, n = 2, m = 2. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 2/2 * 100% = 100%** **SREE's False Positives** = **p - m = 2 – 2 = 0**

```
5.5.16      This subsystem shall contain a rack-mounted oscilloscope.

is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 1, n = 1, m = 1. **SREE's Recall** = **p/n * 100% = 1/1 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/1 * 100% = 100%** **SREE's False Positives** = **p - m = 1 – 1 = 0**

5.5.17 The oscilloscope shall have a Y channel frequency response of at least 20 MHz.

5.5.18 The oscilloscope shall have at least 2 Y (vertical) channels.

is potentially ambiguous (PLURALNOUN) because of wording: channels
is potentially ambiguous (VAGUE) because of wording: (vertical)

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises channels to be potentially ambiguous as defined in the AIC, channels does not contribute to ambiguity in the 5.5.18 RStat.

5.5.19 The oscilloscope shall have floating inputs (i.e., nongrounded signal return lines).

is potentially ambiguous (PLURALNOUN) because of wording: inputs
is potentially ambiguous (PLURALNOUN) because of wording: lines
is potentially ambiguous (VAGUE) because of wording: (i.e., nongrounded signal return lines)

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises lines to be potentially ambiguous as defined in the AIC, lines does not contribute to ambiguity in the 5.5.19 RStat.

5.5.20 The Nascom line status shall be visible in the DOCS room for trouble-shooting purposes.

is potentially ambiguous (PLURALNOUN) because of wording: purposes

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises purposes to be potentially ambiguous as defined in the AIC, purposes does not contribute to ambiguity in the 5.5.20 RStat.

5.6. GROWTH

This section addresses growth requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PRONOUN) because of wording: this

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that addresses is not ambiguous because addresses is a singular verb instead of being a plural noun as detected by SREE.

```
5.6.1 The MCSS shall have a 50-percent growth capability in the number of input and output ports.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: ports
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises ports to be potentially ambiguous as defined in the AIC, ports does not contribute to ambiguity in the 5.6.1 RStat.

```
5.6.2 The MCSS shall be expandable without requiring removal of any equipment from the MSOCC.
```

```
is potentially ambiguous (QUANTIFIER) because of wording: any
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.6.3 The MCSS shall be expandable without requiring down times greater than 30 minutes for more than 25 percent of the overall switching capacity at any one time.
```

```
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**
SREE's False Positives = **p - m = 2 – 2 = 0**

5.6.4 The MCSS shall be capable of being expanded to provide ST to any port requiring ST.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: capable of

Potential ambiguity with value of: p = 2, n = 2, m = 2.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**
SREE's False Positives = **p - m = 2 – 2 = 0**

5.7 RELIABILITY, MAINTAINABILITY, AND AVAILABILITY

is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

Even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity.

Failure of the MCSS shall be defined as the inability of the system to route data and timing within the error rate limits specified in

```
paragraph 5.2.18.    Failure shall also include the inability to
control the switch.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: limits
is potentially ambiguous (VAGUE) because of wording: also
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that also and limits do not contribute to ambiguity but specified is an ambiguous adjective. However, SREE doesn't recognise specified to be potentially ambiguous because specified isn't defined as an indicator of potential ambiguity in the AIC.

```
The availability of any system is a function of UPTIME and DOWNTIME,
which may be expressed as:
```

```
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (WEAK) because of wording: may
is potentially ambiguous (PRONOUN) because of wording: which
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises which to be potentially ambiguous as defined in the AIC, which does not contribute to ambiguity in the statement.

```
    Availability =    (UPTIME)
```

`is potentially ambiguous (VAGUE) because of wording: (uptime)`

Potential ambiguity with value of: p = 2, n = 2, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises (uptime) to be potentially ambiguous as defined in the AIC, (uptime) does not contribute to ambiguity in the statement.

```
            (UPTIME + DOWNTIME)
```

`is potentially ambiguous (VAGUE) because of wording: (uptime + downtime)`

Potential ambiguity with value of: p = 2, n = 2, m = 2.
SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**
SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**
SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that even though SREE recognises (uptime+downtime) to be potentially ambiguous as defined in the AIC, (uptime + downtime) does not contribute to ambiguity in the statement.

`maintainability of a system is defined in terms of the mean time to repair the failed element and bring the system back into operation. All MCSS equipment provided will be installed in such a manner as to facilitate maintenance.`

```
is potentially ambiguous (INCOMPLETES) because of wording: is defined
is potentially ambiguous (PLURALNOUN) because of wording: terms
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: in terms of
is potentially ambiguous (VAGUE) because of wording: provided
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

Although SREE recognises terms and and to be potentially ambiguous as defined in the AIC, terms and and do not contribute to ambiguity in the statement.

```
This section addresses the reliability requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that although SREE recognises addresses to be potentially ambiguous as defined in the AIC, addresses does not contribute to plural ambiguity because addresses acts as a singular verb in the RStat.

```
5.7.1 Availability for the MCSS shall be at least 0.99998.

5.7.2 The MTBF for the MCSS, except for the LCT, shall not be less than 10,000 hours (14 months).

is potentially ambiguous (VAGUE) because of wording: (14 months)
```

`is potentially ambiguous (VAGUE) because of wording: less`

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 0**

The user determined that even though SREE recognises less to be potentially ambiguous as defined in the AIC, less does not contribute to ambiguity in the 5.7.2 RStat.

`5.7.3 The MTBF for each LCT shall not be less than 3000 hours.`

`is potentially ambiguous (VAGUE) because of wording: less`

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises less to be potentially ambiguous as defined in the AIC, less does not contribute to ambiguity in the 5.7.3 RStat.

`5.7.4 The MCSS shall have an MTTR of less than 30 minutes.`

`is potentially ambiguous (VAGUE) because of wording: less`

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises less to be potentially ambiguous as defined in the AIC, less does not contribute to ambiguity in the 5.7.4 RStat.

```
5.7.5 The MCSS shall have at least four external control paths.
is potentially ambiguous (PLURALNOUN) because of wording: paths
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises paths to be potentially ambiguous as defined in the AIC, paths does not contribute to ambiguity in the 5.7.5 RStat.

```
5.7.6 The MCSS shall have redundant control paths to each of the two
MSOCC DOCS (four paths).
is potentially ambiguous (PLURALNOUN) because of wording: paths
is potentially ambiguous (VAGUE) because of wording: (four paths)
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises paths to be potentially ambiguous as defined in the AIC, paths does not contribute to ambiguity in the 5.7.6 RStat.

`5.7.7 The MCSS shall contain redundant switch control units.`

`is potentially ambiguous (PLURALNOUN) because of wording: units`

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises units to be potentially ambiguous as defined in the AIC, units does not contribute to ambiguity in the 5.7.7 RStat.

`5.7.8 The MCSS shall contain two independent, isolated power supplies.`

`is potentially ambiguous (PLURALNOUN) because of wording: supplies`

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises supplies to be potentially ambiguous as defined in the AIC, supplies does not contribute to ambiguity in the 5.7.8 RStat.

`5.7.9 Each power supply shall have a separate connection to the AC power lines.`

`is potentially ambiguous (PLURALNOUN) because of wording: lines`

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 8%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises lines to be potentially ambiguous as defined in the AIC, lines does not contribute to ambiguity in the 5.7.9 RStat.

`5.7.10 The MCSS shall be capable of operating on either one or both of its independent power supplies at any one time.`

```
is potentially ambiguous (OPTIONAL) because of wording: either
is potentially ambiguous (PLURALNOUN) because of wording: supplies
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (VAGUE) because of wording: capable of
is potentially ambiguous (PRONOUN) because of wording: its
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 7, n = 7, m = 5.

SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/7 * 100% = 71%**

SREE's False Positives = **p - m = 7 – 5 = 2**

The user determined that even though SREE recognises both and supplies to be potentially ambiguous as defined in the AIC, both and supplies do not contribute to ambiguity in the 5.7.10 RStat.

`5.7.11 In the event of a failure of one of the power supplies, the MCSS shall continue to operate on the other power supply without affecting switch operation.`

```
is potentially ambiguous (PLURALNOUN) because of wording: supplies
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that other is an ambiguous adjective, but supplies is not an ambiguous plural noun. However, SREE doesn't recognise other to be potentially ambiguous because other isn't defined as a potential ambiguity indicator in the AIC.

```
5.7.12      A changeover from one operational power supply to two operational power supplies shall not affect operation of the MCSS.
is potentially ambiguous (PLURALNOUN) because of wording: supplies
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.
SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**
SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**
SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises supplies to be potentially ambiguous as defined in the AIC, supplies does not contribute to ambiguity in the 5.7.12 RStat

```
5.7.13      A single failure of any component in the MCSS shall not disrupt more than 25 percent of the overall switching capacity.
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises more to be potentially ambiguous as defined in the AIC, more does not contribute to ambiguity in the 5.6.1 RStat

`5.7.14 The removal of an LRU shall not require the MCSS to be powered down.`

`5.7.15 The removal of an LRU shall affect or disable no more than four signal pairs.`

```
is potentially ambiguous (PLURALNOUN) because of wording: pairs
is potentially ambiguous (VAGUE) because of wording: more
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises more to be potentially ambiguous as defined in the AIC, more does not contribute to ambiguity in the 5.7.15 RStat.

`5.7.16 The replacement of an LRU while the MCSS is powered on shall not disrupt or impact any circuits other than those connected to the LRU being replaced.`

```
is potentially ambiguous (PLURALNOUN) because of wording: circuits
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (PRONOUN) because of wording: those
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises circuits to be potentially ambiguous as defined in the AIC, circuits does not contribute to ambiguity in the 5.7.16 RStat.

5.7.17 The MCSS shall have sufficient data generation and acquisition tools to troubleshoot, replace, and verify the proper operation of all LRU's of the MCSS.

```
is potentially ambiguous (PLURALNOUN) because of wording: tools
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: sufficient
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 3.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/4 * 100% = 75%**

SREE's False Positives = **p - m = 4 – 3 = 1**

The user determined that even though SREE recognises tools to be potentially ambiguous as defined in the AIC, tools does not contribute to ambiguity in the 5.7.17 RStat.

5.7.18 During the course of troubleshooting, when access to areas under investigation is obtained by sliding, rotating, or hinged parts, such parts shall be free to open, extend, or rotate full distance and remain in the open state without requiring additional support.

is potentially ambiguous (PLURALNOUN) because of wording: areas
is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or

Potential ambiguity with value of: p = 4, n = 4, m = 4.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/4 * 100% = 100%**

SREE's False Positives = **p - m = 4 – 4 = 0**

The user determined that additional and full are potentially ambiguous. However, SREE doesn't recognise additional and full to be potentially ambiguous because additional and full aren't defined as potential ambiguity indicators in the AIC.

5.7.19 Clearance for maintenance access shall be provided.

is potentially ambiguous (VAGUE) because of wording: provided

Potential Ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.7.20 MCSS equipment shall be installed in such a manner so as to permit the replacement of faulty LRU's without cutting, desoldering, unwrapping, or the use of other techniques requiring more disassembly than the removal of screws and connectors.

```
is potentially ambiguous (PLURALNOUN) because of wording: connectors
is potentially ambiguous (PLURALNOUN) because of wording: screws
is potentially ambiguous (PLURALNOUN) because of wording: techniques
is potentially ambiguous (VAGUE) because of wording: more
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 6, n = 6, m = 4.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/6 * 100% = 67%**

SREE's False Positives = **p - m = 6 – 4 = 2**

The user determined that in such a manner is an ambiguous adjective, but connectors and screws are not potentially ambiguous instances. However, SREE doesn't recognise in such a manner to be potentially ambiguous because in such a manner isn't defined as a potential ambiguity indicator in the AIC.

5.7.21 The MCSS shall be delivered with any unique maintenance tools and support devices required by the equipment.

```
is potentially ambiguous (PLURALNOUN) because of wording: devices
is potentially ambiguous (PLURALNOUN) because of wording: tools
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that unique is an ambiguous adjective but devices and and are not potentially ambiguous instances. SREE doesn't recognise unique to be potentially ambiguous because unique isn't defined as an indicator of potential ambiguity in the AIC.

5.7.22 The MCSS shall be delivered with a 1-year warranty for parts and labor in the MSOCC, and on-call service (onsite response time within 4 hours of the time of the call) during the prime shift (0800 to 1700 Eastern time).

is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (VAGUE) because of wording: (onsite response time within 4 hours of the time of the call) during the prime shift (0800 to 1700 eastern time)
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.7.23 The period of the warrantee shall begin following acceptance of the MCSS in the MSOCC.

is potentially ambiguous (CONTINUANCE) because of wording: following

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.8 SAFETY

This section addresses safety requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PRONOUN) because of wording: this

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises addresses to be potentially ambiguous as defined in the AIC, addresses does not contribute to ambiguity in the statement.

`5.8.1 The MCSS shall contain interlocks and/or other devices and methods to eliminate exposure to any voltage in excess of 30 V Root Mean Square (RMS).`

```
is potentially ambiguous (PLURALNOUN) because of wording: devices
is potentially ambiguous (PLURALNOUN) because of wording: methods
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: (rms)
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: and/or
```

Potential ambiguity with value of: p = 7, n = 7, m = 6.

SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/7 * 100% = 86%**

SREE's False Positives = **p - m = 7 – 6 = 1**

The user determined that even though SREE recognises / to be potentially ambiguous as defined in the AIC, / does not contribute to ambiguity in the statement.

`5.8.2 The MCSS shall contain mechanical protection to prevent people from coming into contact with moving parts such as gears, fans, and belts.`

```
is potentially ambiguous (PLURALNOUN) because of wording: belts
is potentially ambiguous (PLURALNOUN) because of wording: fans
is potentially ambiguous (PLURALNOUN) because of wording: gears
is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (PLURALNOUN) because of wording: people
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 6.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/6 * 100% = 100%**

SREE's False Positives = **p - m = 6 – 6 = 0**

5.8.3 The MCSS shall have a master power switch that shall be able to disconnect all AC feeds to the MCSS.

```
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (PRONOUN) because of wording: that
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.9 FACILITIES

is potentially ambiguous (PLURALNOUN) because of wording: facilities

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

This section addresses facilities requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses

```
is potentially ambiguous (PLURALNOUN) because of wording: facilities
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises addresses to be potentially ambiguous as defined in the AIC, addresses does not contribute to ambiguity in the statement.

```
5.9.1 The MCSS shall be installed in room E239, Building 14, at Goddard Space Flight Center.

5.9.2 The MCSS shall only use main power at 120 VAC +/- 10 percent single-phase 60 Hz (57 to 63).

is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: (57 to 63)
is potentially ambiguous (VAGUE) because of wording: only
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that even though SREE recognises only to be potentially ambiguous as defined in the AIC, only does not contribute to ambiguity in the statement.

```
5.9.3 The MCSS shall use two separate independent AC power cables.

is potentially ambiguous (PLURALNOUN) because of wording: cables
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises cables to be potentially ambiguous as defined in the AIC, cables does not contribute to ambiguity in the statement.

```
5.9.4 Each MCSS AC power cable shall be terminated in a male twist-lock connector.
5.9.5 Each MCSS AC power cable shall be at least 10 feet long.
5.9.6 The MCSS shall utilize no more than 2 kilovolt-amperes (kVa) of power, based on 120-VAC, using a power factor of 0.66.
is potentially ambiguous (PLURALNOUN) because of wording: amperes
is potentially ambiguous (VAGUE) because of wording: (kva)
is potentially ambiguous (VAGUE) because of wording: more
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises amperes and more to be potentially ambiguous instances as defined in the AIC, amperes and more do not contribute to ambiguity in the 5.9.6 RStat.

```
5.9.7 Power transients of a +/-10 percent change from the nominal
voltage lasting for 2 seconds shall not interfere with MCSS operation
during and immediately following the transient period.

is potentially ambiguous (PLURALNOUN) because of wording: transients
is potentially ambiguous (CONTINUANCE) because of wording: following
is potentially ambiguous (VAGUE) because of wording: /
is potentially ambiguous (VAGUE) because of wording: transient
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 5.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 5/5 * 100% = 100%**

SREE's False Positives = **p - m = 5 – 5 = 0**

```
5.9.8 The MCSS shall not be damaged by short-duration, high-amplitude
transients of as much as +/-50 percent change from nominal voltage for
a period of 1 millisecond per line cycle.

is potentially ambiguous (PLURALNOUN) because of wording: transients
is potentially ambiguous (QUANTIFIER) because of wording: much
is potentially ambiguous (VAGUE) because of wording: /
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises much and transients to be potentially ambiguous instances as defined in the AIC, much and transients do not contribute to ambiguity in the 5.9.8 RStat.

5.9.9 The MCSS shall not be damaged by a sudden loss of power or prolonged transients of the kind mentioned in paragraphs 5.9.7 and 5.9.8 on the power supply line.

```
is potentially ambiguous (PLURALNOUN) because of wording: paragraphs
is potentially ambiguous (PLURALNOUN) because of wording: transients
is potentially ambiguous (COORDINATOR) because of wording: and
is potentially ambiguous (COORDINATOR) because of wording: or
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises paragraphs and transients to be potentially ambiguous instances as defined in the AIC, paragraphs and transients do not contribute to ambiguity in the 5.9.9 RStat.

5.9.10 The MCSS shall be packaged in EIA 19-inch horizontal mounting width racks.

```
is potentially ambiguous (PLURALNOUN) because of wording: racks
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that even though SREE recognises racks to be potentially ambiguous as defined in the AIC, racks does not contribute to ambiguity in the 5.9.10 RStat.

5.9.11 The MCSS shall occupy no more floor space than four racks, 30 inches deep, 24 inches wide, and 78 inches tall.

is potentially ambiguous (PLURALNOUN) because of wording: inches
is potentially ambiguous (PLURALNOUN) because of wording: racks
is potentially ambiguous (VAGUE) because of wording: more
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 4, n = 4, m = 1.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/4 * 100% = 25%**

SREE's False Positives = **p - m = 4 – 1 = 3**

The user determined that even though SREE recognises and, racks, and inches to be potentially ambiguous instances as defined in the AIC, and, racks, and inches do not contribute to ambiguity in the 5.9.11 RStat.

5.9.12 The racks holding the MCSS shall be painted Federal color standard 595a-20372 using vinyl textured paint.

is potentially ambiguous (PLURALNOUN) because of wording: racks

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.9.13 The MCSS racks shall be freestanding.

is potentially ambiguous (PLURALNOUN) because of wording: racks

Potential ambiguity with value of: p = 1, n = 1, m = 1. **SREE's Recall** = **p/n * 100% = 1/1 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/1 * 100% = 100%** **SREE's False Positives** = **p - m = 1 – 1 = 0**

```
5.9.14      The MCSS racks shall have retractable casters.

is potentially ambiguous (PLURALNOUN) because of wording: racks
```

Potential ambiguity with value of: p = 1, n = 1, m = 1. **SREE's Recall** = **p/n * 100% = 1/1 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 1/1 * 100% = 100%** **SREE's False Positives** = **p - m = 1 – 1 = 0**

```
5.9.15      The weight distribution of the MCSS equipment racks shall
be limited to 250 pounds per square foot of floor space.

is potentially ambiguous (PLURALNOUN) because of wording: pounds
is potentially ambiguous (PLURALNOUN) because of wording: racks
```

Potential ambiguity with value of: p = 2, n = 2, m = 0. **SREE's Recall** = **p/n * 100% = 2/2 * 100% = 100%** **SREE's Precision** = **m/p * 100% = 0/2 * 100% = 0%** **SREE's False Positives** = **p - m = 2 – 0 = 2**

The user determined that even though SREE recognises pounds and racks to be potentially ambiguous instances as defined in the AIC, pounds and racks do not contribute to ambiguity in the 5.9.15 RStat.

`5.9.16 The weight of a single rack containing equipment shall not exceed 1000 pounds.`

```
is potentially ambiguous (PLURALNOUN) because of wording: pounds
```

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises pounds to be potentially ambiguous as defined in the AIC, pounds does not contribute to ambiguity in the 5.9.16 RStat.

`5.9.17 The MCSS racks shall be capable of being bolted together side by side in a straight line.`

```
is potentially ambiguous (PLURALNOUN) because of wording: racks
is potentially ambiguous (VAGUE) because of wording: capable of
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

`5.9.18 The MCSS shall be capable of being easily separated into single-rack units for shipping purposes only.`

```
is potentially ambiguous (PLURALNOUN) because of wording: purposes
is potentially ambiguous (PLURALNOUN) because of wording: units
is potentially ambiguous (VAGUE) because of wording: capable of
is potentially ambiguous (VAGUE) because of wording: easily
is potentially ambiguous (VAGUE) because of wording: only
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that even though SREE recognises units and purposes to be instances of potential ambiguity as defined in the AIC, units and purposes do not contribute to ambiguity in the 5.9.18 RStat.

```
5.9.19      Each MCSS rack shall contain a bus bar connected to the
MSOCC ground, as shown in Figure 5-2.
is potentially ambiguous (DIRECTIVE) because of wording: figure
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.9.20      The grounding of equipment in a rack to the bus bar shall
be in accordance with STDN SPEC-7.
5.9.21      Access to the equipment in the racks shall be from the
front and back of the cabinet.
is potentially ambiguous (PLURALNOUN) because of wording: racks
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises racks to be potentially ambiguous as defined in the AIC, racks does not contribute to ambiguity in the 5.9.21 RStat.

```
5.9.22      All rack-mounted equipment shall be uniquely labeled and serialized.
```

```
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that uniquely is potentially ambiguous. However, SREE doesn't recognise uniquely to be potentially ambiguous due to uniquely isn't defined as a potential ambiguity indicator in the AIC.

```
5.9.23      Every interrack cable within the MCSS shall be uniquely labeled.
```

The user determined that uniquely is an ambiguous adjective. However, SREE doesn't recognise uniquely to be potentially ambiguous due to uniquely isn't defined as a potential ambiguity indicator in the AIC.

```
5.9.24.1    The interrack cable label shall contain the connector source
5.9.24.2    The interrack cable label shall contain the connector destination
5.9.24.3    The interrack cable label shall contain the cable number.
```

```
5.9.25      The interrack cable label shall be placed on each end of
the cable, approximately 3 inches from the connector.
is potentially ambiguous (PLURALNOUN) because of wording: inches
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises inches to be potentially ambiguous as defined in the AIC, inches does not contribute to ambiguity in the 5.9.25 RStat.

```
5.9.26      All external cabling shall enter the rack from under the
floor in an area beginning not less than 2 inches from the back
surface of the rack and extending toward the front, but not exceeding
12 inches from the back surface of the rack.
is potentially ambiguous (PLURALNOUN) because of wording: inches
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: but
is potentially ambiguous (VAGUE) because of wording: less
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that even though SREE recognises less and inches to be potentially ambiguous instances as defined in the AIC, less and inches do not contribute to ambiguity in the 5.9.26 RStat.

5.9.27 The MCSS shall be able to operate using ambient air cooling.

5.9.28 The MCSS shall be able to operate using under-floor plenum cooling air between 65 and 80 degrees Fahrenheit; relative humidity will be between 35 and 70 percent noncondensing.

is potentially ambiguous (PLURALNOUN) because of wording: degrees
is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (VAGUE) because of wording: ;
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises degrees and will to be potentially ambiguous instances as defined in the AIC, degrees and will do not contribute to ambiguity in the 5.9.28 RStat.

5.9.29 The temperature of the air exhaust from the MCSS rack shall be within 10 degrees Fahrenheit of air being input.

is potentially ambiguous (PLURALNOUN) because of wording: degrees

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises degrees to be potentially ambiguous as defined in the AIC, degrees does not contribute to ambiguity in the 5.9.29 RStat.

5.9.30 The MCSS shall not generate more than 70 dBA of noise at a distance of 6 feet in any direction.

is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: more

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that even though SREE recognises more to be potentially ambiguous as defined in the AIC, more does not contribute to ambiguity in the 5.9.30 RStat.

5.9.31 The operation of the MCSS shall not emit electromagnetic interference that will interfere with the operation of equipment already installed in the MSOCC.

is potentially ambiguous (WEAK) because of wording: will
is potentially ambiguous (PRONOUN) because of wording: that

Potential ambiguity with value of: p =2, n = 2, m = 0.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/2 * 100% = 0%**

SREE's False Positives = **p - m = 2 – 0 = 2**

The user determined that even though SREE recognises will and that to be potentially ambiguous instances as defined in the AIC, will and that does not contribute to ambiguity in the 5.9.31 RStat.

```
5.10  TECHNICAL SUPPORT

This section addresses technical support requirements for documentation and spare parts.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (PRONOUN) because of wording: this
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that addresses is not ambiguous because addresses acts as a singular verb rather than as plural noun.

```
The MCSS shall be delivered with manuals for operation, maintenance, and testing the system.  The manuals shall include, but not be limited to, those described in the following paragraphs.

is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (PLURALNOUN) because of wording: paragraphs
is potentially ambiguous (CONTINUANCE) because of wording: following
is potentially ambiguous (VAGUE) because of wording: but
is potentially ambiguous (PRONOUN) because of wording: those
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 6, n = 6, m = 6.

SREE's Recall = **p/n * 100% = 6/6 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/6 * 100% = 100%**

SREE's False Positives = **p - m = 6 – 6 = 0**

5.10.1 A software users guide.

is potentially ambiguous (PLURALNOUN) because of wording: users

Potential ambiguity with value of: p = 1, n = 1, m = 0.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/1 * 100% = 0%**

SREE's False Positives = **p - m = 1 – 0 = 1**

The user determined that even though SREE recognises users to be potentially ambiguous as defined in the AIC, users does not contribute to ambiguity in the 5.10.1 RStat.

5.10.2 MCSS technical manuals for the operation, maintenance, and testing of all the hardware components and systems of the switching system, described as follows:

is potentially ambiguous (PLURALNOUN) because of wording: components
is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (PLURALNOUN) because of wording: systems
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (CONTINUANCE) because of wording: :
is potentially ambiguous (CONTINUANCE) because of wording: as follows
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 7, n = 7, m = 7.

SREE's Recall = **p/n * 100% = 7/7 * 100% = 100%**

SREE's Precision = **m/p * 100% = 7/7 * 100% = 100%**

SREE's False Positives = **p - m = 7 – 7 = 0**

a. Manufacturer system hardware manuals describing system architecture, Central Processing Unit (CPU), memory, and peripheral devices.

is potentially ambiguous (PLURALNOUN) because of wording:

```
devices
is potentially ambiguous (PLURALNOUN) because of wording:
manuals
is potentially ambiguous (VAGUE) because of wording: (cpu)
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises and and devices to be potentially ambiguous instances as defined in the AIC, and and devices does not contribute to ambiguity in the 5.10.2.a RStat.

```
b.    Interface manuals describing electrical and mechanical aspects of system interfaces, such as peripheral devices.

is potentially ambiguous (PLURALNOUN) because of wording:
aspects
is potentially ambiguous (PLURALNOUN) because of wording:
devices
is potentially ambiguous (PLURALNOUN) because of wording:
interfaces
is potentially ambiguous (PLURALNOUN) because of wording:
manuals
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 2.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/5 * 100% = 40%**

SREE's False Positives = **p - m = 5 – 2 = 3**

The user determined that even though SREE recognises aspects, devices, and interfaces to be potentially ambiguous instances as defined in the AIC, aspects, devices, and interfaces do not contribute to ambiguity in the 5.10.2.b RStat.

c. Diagnostic procedure manuals needed to identify a replaceable hardware component failure.

is potentially ambiguous (PLURALNOUN) because of wording: manuals

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

d. Manuals containing a list of replacement part numbers, installation and removal procedures, schematic diagrams, net lists, Integrated Circuit (IC) data sheets, and basic operating procedures.

is potentially ambiguous (PLURALNOUN) because of wording: diagrams
is potentially ambiguous (PLURALNOUN) because of wording: lists
is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (PLURALNOUN) because of wording: numbers
is potentially ambiguous (PLURALNOUN) because of wording: procedures
is potentially ambiguous (PLURALNOUN) because of wording: sheets
is potentially ambiguous (VAGUE) because of wording: (ic)
is potentially ambiguous (COORDINATOR) because of wording: and

Potential ambiguity with value of: p = 8, n = 8, m = 3.

SREE's Recall = **p/n * 100% = 8/8 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/8 * 100% = 38%**

SREE's False Positives = **p - m = 8 – 3 = 5**

The user determined that even though SREE recognises numbers, diagrams, lists, sheets, and procedures to be potentially ambiguous instances as defined in the AIC, they do not contribute to ambiguity in the 5.10.2.d RStat.

5.10.3 The documentation needed to make use of the growth capabilities of the MCSS in terms of hardware requirements and software parameters.

```
is potentially ambiguous (PLURALNOUN) because of wording: capabilities
is potentially ambiguous (PLURALNOUN) because of wording: parameters
is potentially ambiguous (PLURALNOUN) because of wording: terms
is potentially ambiguous (VAGUE) because of wording: in terms of
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 2.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/5 * 100% = 40%**

SREE's False Positives = **p - m = 5 – 2 = 3**

The user determined that but parameters, terms, and and are not ambiguous. terms shouldn't be taken as a single instance rather than as the whole keyphrase in terms of.

5.10.4 Training manuals for new personnel to be trained, both on hardware and software operations.

```
is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (PLURALNOUN) because of wording: operations
is potentially ambiguous (PLURALNOUN) because of wording: personnel
is potentially ambiguous (VAGUE) because of wording: both
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 4.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 4/5 * 100% = 80%**

SREE's False Positives = **p - m = 5 – 4 = 1**

The user determined that even though SREE recognises operations to be potentially ambiguous as defined in the AIC, operations does not contribute to ambiguity in the 5.10.4 RStat.

```
5.10.5      Software source code, libraries, object modules, and custom software developed for the MCSS.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: libraries
is potentially ambiguous (PLURALNOUN) because of wording: modules
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 2.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/3 * 100% = 67%**

SREE's False Positives = **p - m = 3 – 2 = 1**

The user determined that custom is an ambiguous adjective, but modules is not an ambiguous plural noun. However, SREE doesn't recognise custom to be potentially ambiguous because custom isn't defined as a potential ambiguity indicator in the AIC.

```
5.10.6      Computer manufacturer reference and system programming manuals detailing machine instructions and programming considerations.
```

```
is potentially ambiguous (PLURALNOUN) because of wording:
considerations
is potentially ambiguous (PLURALNOUN) because of wording: instructions
is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that even though SREE recognises considerations and instructions to be potentially ambiguous instances as defined in the AIC, considerations and instructions do not contribute to ambiguity in the 5.10.6 RStat.

```
5.10.7      Documentation of custom modifications and changes to
purchased software.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: changes
is potentially ambiguous (PLURALNOUN) because of wording:
modifications
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that custom is an ambiguous adjective, but changes and modifications are not potentially ambiguous instances. However, SREE doesn't recognise custom to be potentially ambiguous because custom isn't defined as a potential ambiguity indicator in the AIC.

```
5.10.8      Detailed reference manuals describing all elements and
operations of supplied software (e.g., language compilers, text
editors, communications drivers, software tools, diagnostics,
interface drivers).
```

```
is potentially ambiguous (PLURALNOUN) because of wording:
communications
is potentially ambiguous (PLURALNOUN) because of wording: compilers
is potentially ambiguous (PLURALNOUN) because of wording: drivers
is potentially ambiguous (PLURALNOUN) because of wording: editors
is potentially ambiguous (PLURALNOUN) because of wording: elements
is potentially ambiguous (PLURALNOUN) because of wording: manuals
is potentially ambiguous (PLURALNOUN) because of wording: operations
is potentially ambiguous (PLURALNOUN) because of wording: tools
is potentially ambiguous (QUANTIFIER) because of wording: all
is potentially ambiguous (VAGUE) because of wording: (e.g., language
```

```
compilers, text editors, communications drivers, software tools,
diagnostics, interface drivers)
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 11, n = 11, m = 6.

SREE's Recall = **p/n * 100% = 11/11 * 100% = 100%**

SREE's Precision = **m/p * 100% = 6/11 * 100% = 55%**

SREE's False Positives = **p - m = 11 – 6 = 5**

The user determined that even though SREE recognises compilers, editors, communications, drivers, and tools to be potentially ambiguous instances as defined in the AIC, they do not contribute to ambiguity in the 5.10.8 RStat.

```
5.10.9    Problem determination and debugging guides that shall
include documentation of known problems and suspected system errors.

is potentially ambiguous (PLURALNOUN) because of wording: errors
is potentially ambiguous (PLURALNOUN) because of wording: guides
is potentially ambiguous (PLURALNOUN) because of wording: problems
is potentially ambiguous (PRONOUN) because of wording: that
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that even though SREE recognises errors and that to be potential ambiguities as defined in the AIC, errors and that do not contribute to ambiguity in the 5.10.9 RStat. However, suspected is potentially ambiguous and should be defined as a potential ambiguity indicator in the AIC.

```
5.10.10    Test plans and test procedures used in accepting the MCSS.
```

```
is potentially ambiguous (PLURALNOUN) because of wording: procedures
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that plans is potentially ambiguous. However, SREE doesn't recognise plans to be potentially ambiguous because plans isn't defined as an indicator of potential ambiguity in the AIC.

5.10.11 The MCSS hardware and software documentation shall be at the same revision level as the hardware and software.

```
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

5.10.12 All identical parts shall be at the same revision level.

```
is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (QUANTIFIER) because of wording: all
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.10.13 At least 10 percent spares (minimum one item) for each replaceable hardware item (LRU) shall be provided.

is potentially ambiguous (PLURALNOUN) because of wording: spares
is potentially ambiguous (VAGUE) because of wording: (minimum one item) for each replaceable hardware item (lru)
is potentially ambiguous (VAGUE) because of wording: provided

Potential ambiguity with value of: p = 3, n = 3, m = 3.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/3 * 100% = 100%**

SREE's False Positives = **p - m = 3 – 3 = 0**

5.10.14 At least 20 percent spares (minimum 2 items) for each replaceable Application Specific Integrated Circuit (ASIC) shall be provided.

is potentially ambiguous (PLURALNOUN) because of wording: items
is potentially ambiguous (PLURALNOUN) because of wording: spares
is potentially ambiguous (VAGUE) because of wording: (minimum 2 items) for each replaceable application specific integrated circuit (asic)
is potentially ambiguous (VAGUE) because of wording: provided
is potentially ambiguous (VAGUE) because of wording: specific

Potential ambiguity with value of: p = 5, n = 5, m = 3.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/5 * 100% = 60%**

SREE's False Positives = **p - m = 5 – 3 = 2**

The user determined that even though SREE recognises specific and items to be potentially ambiguous instances as defined in the AIC, specific and items do not contribute to ambiguity in the 5.10.14 RStat.

5.10.15 All test rigs shall be delivered with the MCSS.

is potentially ambiguous (PLURALNOUN) because of wording: rigs

```
is potentially ambiguous (QUANTIFIER) because of wording: all
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

5.10.16 Test rig documentation shall be delivered with the MCSS.

5.10.17 Test rig documentation requirements shall be the same as those in paragraphs 5.10.1 through 5.10.11.

```
is potentially ambiguous (PLURALNOUN) because of wording: paragraphs
is potentially ambiguous (PRONOUN) because of wording: those
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that requirements is potentially ambiguous, but paragraphs is not an ambiguous plural noun. However, SREE doesn't recognise requirements to be potentially ambiguous because requirements isn't defined as a potential ambiguity indicator in the AIC.

5.10.18 Custom test rig spare parts requirements shall be the same as those in paragraphs 5.10.12 and 5.10.13.

```
is potentially ambiguous (PLURALNOUN) because of wording: paragraphs
is potentially ambiguous (PLURALNOUN) because of wording: parts
is potentially ambiguous (PRONOUN) because of wording: those
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 4, n = 4, m = 2.

SREE's Recall = **p/n * 100% = 4/4 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/4 * 100% = 50%**

SREE's False Positives = **p - m = 4 – 2 = 2**

The user determined that requirements is potentially ambiguous, but paragraphs and parts are not instances of potential ambiguity. SREE doesn't recognise requirements to be potentially ambiguous because requirements isn't defined as a potential ambiguity indicator in the AIC.

```
5.11  TRAINING

This section addresses training requirements.

is potentially ambiguous (PLURALNOUN) because of wording: addresses
is potentially ambiguous (PRONOUN) because of wording: this
```

Potential ambiguity with value of: p = 2, n = 2, m = 1.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/2 * 100% = 50%**

SREE's False Positives = **p - m = 2 – 1 = 1**

The user determined that addresses is not ambiguous because addresses acts as a singular verb rather than a plural noun.

```
5.11.1      The MCSS shall include a training plan that shall include
a set of outlines for training courses.

is potentially ambiguous (PLURALNOUN) because of wording: courses
is potentially ambiguous (PLURALNOUN) because of wording: outlines
is potentially ambiguous (PRONOUN) because of wording: that
```

Potential ambiguity with value of: p = 3, n = 3, m = 0.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 0/3 * 100% = 0%**

SREE's False Positives = **p - m = 3 – 0 = 3**

The user determined that even though SREE recognises that, outlines, and sets to be potentially ambiguous instances as defined in the AIC, they do not contribute to ambiguity in the 5.11.1 RStat.

```
5.11.2    The outline of each course  shall include, but not be limited to, course objectives, topics addressed, prerequisite levels of technical skills (if any), and duration of the course (in hours and days).
```

```
is potentially ambiguous (PLURALNOUN) because of wording: levels
is potentially ambiguous (PLURALNOUN) because of wording: objectives
is potentially ambiguous (PLURALNOUN) because of wording: skills
is potentially ambiguous (PLURALNOUN) because of wording: topics
is potentially ambiguous (QUANTIFIER) because of wording: any
is potentially ambiguous (VAGUE) because of wording: (if any), and
duration of the course (in hours and days)
is potentially ambiguous (VAGUE) because of wording: but
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 8, n = 8, m = 3.

SREE's Recall = **p/n * 100% = 8/8 * 100% = 100%**

SREE's Precision = **m/p * 100% = 3/8 * 100% = 38%**

SREE's False Positives = **p - m = 8 – 3 = 5**

The user determined that even though SREE recognises levels, objectives, skills, and topics, to be potentially ambiguous instances as defined in the AIC, they do not contribute to ambiguity in the 5.11.2 RStat.

```
5.11.3      Each trainee shall be provided with appropriate written
course material.
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

The user determined that appropriate is potentially ambiguous. However, SREE doesn't recognise appropriate to be potentially ambiguous because appropriate isn't defined as a potential ambiguity indicator in the AIC.

```
5.11.4      A minimum of 4 contiguous hours of operations training
shall be provided for a minimum of 4 operations personnel for each of
4 shifts.
is potentially ambiguous (PLURALNOUN) because of wording: operations
is potentially ambiguous (PLURALNOUN) because of wording: shifts
is potentially ambiguous (PLURALNOUN) because of wording: personnel
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 5, n = 5, m = 1.

SREE's Recall = **p/n * 100% = 5/5 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/5 * 100% = 20%**

SREE's False Positives = **p - m = 5 – 1 = 4**

The user determined that even though SREE recognises shifts, operations, personnel, and minimum to be potentially ambiguous instances as defined in the AIC, shifts, operations, personnel, and minimum do not contribute to ambiguity in the 5.11.4 RStat.

```
5.11.5      A minimum of 40 contiguous hours of maintenance training for 8 persons shall be provided.

is potentially ambiguous (PLURALNOUN) because of wording: persons
is potentially ambiguous (VAGUE) because of wording: minimum
is potentially ambiguous (VAGUE) because of wording: provided
```

Potential ambiguity with value of: p = 3, n = 3, m = 1.

SREE's Recall = **p/n * 100% = 3/3 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/3 * 100% = 33%**

SREE's False Positives = **p - m = 3 – 1 = 2**

The user determined that even though SREE recognises persons and minimum to be potentially ambiguous instances as defined in the AIC, persons and minimum do not contribute to ambiguity in the 5.11.4 RStat.

```
5.11.6      The location, time, and content of each course shall be subject to review and approval by the Government.

is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 1, n = 1, m = 1.

SREE's Recall = **p/n * 100% = 1/1 * 100% = 100%**

SREE's Precision = **m/p * 100% = 1/1 * 100% = 100%**

SREE's False Positives = **p - m = 1 – 1 = 0**

```
5.11.7      Postponed courses shall be rescheduled with the mutual agreement of the Government and the MCSS supplier.

is potentially ambiguous (PLURALNOUN) because of wording: courses
is potentially ambiguous (COORDINATOR) because of wording: and
```

Potential ambiguity with value of: p = 2, n = 2, m = 2.

SREE's Recall = **p/n * 100% = 2/2 * 100% = 100%**

SREE's Precision = **m/p * 100% = 2/2 * 100% = 100%**

SREE's False Positives = **p - m = 2 – 2 = 0**

The user determined that even though SREE recognises and to be potentially ambiguous as defined in the AIC, and does not contribute to ambiguity in the 5.11.7 RStat.